

**PENGEMBANGAN APLIKASI PEMESANAN JASA INSTRUKTUR  
OLAHRAGA KOTA MALANG BERBASIS WEB**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Tiara Rossa Diassananda

NIM: 155150201111199



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2019**

## PENGESAHAN

PENGEMBANGAN APLIKASI PEMESANAN JASA INSTRUKTUR OLAHRAGA  
KOTA MALANG BERBASIS WEB

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :  
Tiara Rossa Diassananda  
NIM: 155150201111199

Skripsi ini telah diuji dan dinyatakan lulus pada  
22 Juli 2019

Telah diperiksa dan disetujui oleh:

Pembimbing I

Pembimbing II

  
Bayu Priyambadha, S.Kom, M.Kom  
NIP: 19820909 200812 1 004

  
Arief Andy Soebroto, ST., M.Kom  
NIP : 197204251999031002

Mengetahui  
Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D  
NIP: 19710518 200312 1 001



## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 27 July 2019



Tiara Rossa Diassananda

NIM: 155150201111199



## PRAKATA

Puji syukur kepada Tuhan Yang Maha Esa atas karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “PENGEMBANGAN APLIKASI PEMESANAN JASA INSTRUKTUR OLAHRAGA KOTA MALANG BERBASIS WEB”. Penulis mengucapkan terima kasih kepada semua pihak yang sangat membantu penulis dalam menyelesaikan skripsi ini. Ucapan terima kasih ditujukan kepada yang terhormat:

1. Kedua orangtua yaitu Bapak Budi Setiawan dan Ibu Cucik Asmindiati, adik saya satu-satunya Gagah Hanugra Diassananda serta seluruh keluarga besar penulis atas segala doa, nasihat, dukungan baik moril maupun materil yang begitu besar terhadap kelancaran dalam menyelesaikan skripsi ini.
2. Bapak Bayu Priyambadha, S. Kom, M. Kom selaku Dosen Pembimbing I dan Bapak Arief Andy Soebroto, ST., M.Kom selaku Dosen Pembimbing II yang telah sabar dalam memberikan bimbingan, ilmu, arahan, bantuan, dan saran terhadap pelaksanaan dan penulisan skripsi penulis.
3. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D. selaku ketua Jurusan Teknik Informatika.
4. Bapak Agus Wahyu Widodo, S.T, M.Cs selaku ketua Program Studi Teknik Informatika.
5. Bapak Wayan Firdaus Mahmudy, S. Si, M.T., Ph.D. selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
6. Ibu Fitri Utaminngrum, Dr. Eng., S.T, M.T selaku fasilitator seminar hasil skripsi saya yang sudah memberikan masukan untuk penelitian skripsi saya.
7. Bapak Denny Sagita, S.Kom, M.Kom dan Bapak Achmad Arwan, S.Kom, M.Kom selaku dosen penguji skripsi saya yang sudah menjadikan penelitian skripsi saya lebih baik.
8. Seluruh dosen dan karyawan Fakultas Ilmu Komputer Universitas Brawijaya yang telah mendidik dan memberikan ilmu kepada penulis selama menempuh masa studi di Fakultas Ilmu Komputer.
9. Teman-teman bimbingan Bapak Bayu Priyambadha, S. Kom, M. Kom dan Bapak Arief Andy Soebroto, ST., M.Kom yang telah membantu penulis mengerjakan skripsi.
10. Teman istimewa saya selama tujuh tahun yang sudah memberikan dukungan moril maupun materil, semangat serta doa untuk penulis menyelesaikan pengerjaan skripsi ini.
11. Teman-teman Pony Family dan teman-teman istimewa penulis lainnya yang sudah membantu penulis menyelesaikan pengerjaan skripsi ini.
12. Semua pihak yang tidak dapat disebutkan satu persatu, terima kasih banyak atas bantuan, semangat serta doa yang diberikan kepada penulis.

Penulis menyadari bahwa skripsi ini masih banyak kekurangan. Oleh karena itu penulis akan dengan senang hati menerima kritikan dan saran yang membangun dalam menyempurnakan segala kekurangan penelitian ini. Akhir kata penulis berharap agar skripsi ini dapat bermanfaat dan menambah wawasan bagi pembaca maupun penulis sendiri.

Malang, 21 July 2019

Penulis

[tyaradiassananda@gmail.com](mailto:tyaradiassananda@gmail.com)



## ABSTRAK

**Tiara Rossa Diassananda ,Pengembangan Aplikasi Pemesanan Jasa Instruktur Olahraga Kota Malang**

**Pembimbing: Bayu Priyambadha, S.Kom, M.Kom dan Arief Andy Soebroto, S.T, M.Kom**

Olahraga adalah salah satu kegiatan untuk meningkatkan kualitas hidup dan kesehatan jasmani maupun rohani. Pada era sekarang dalam melakukan kegiatan olahraga diberikan berbagai kemudahan diantaranya : dengan meniru tutorial diyoutube ataupun menyewa jasa instruktur olahraga. Dari hasil wawancara 14 instruktur olahraga Kota Malang menyebutkan bahwa kendala dalam pemasaran jasa mereka adalah penyebaran informasinya yang masih terbatas serta persaingan dan perkembangan yang pesat dalam dunia instruktur olahraga. Sedangkan dari hasil survey responden di Kota Malang yang terbagai menjadi 3 kalangan yaitu umum, pelajar dan mahasiswa terdiri dari laki-laki dan perempuan 83,8% dari mereka menyukai olahraga seperti zumba, aerobic ,yoga, gym, renang, muaythai dan 60,6% dari mereka membutuhkan jasa instruktur olahraga dalam melakukan kegiatan olahraga tersebut tetapi 63,4% dari mereka menemui masalah dalam pencarian instruktur olahraga diantaranya sulitnya mencari profil instruktur, kurangnya refrensi, sulit mencari instruktur yang dapat menyesuaikan dengan jadwal kita dan kurangnya informasi paket harga instruktur per sesi latihan. Solusi yang dikembangkan adalah mengembangkan aplikasi pemesanan jasa instruktur olaharaga berbasis web dengan metode pengembangan perangkat lunak waterfall. Aplikasi yang dibangun menggunakan basis web dengan tujuan mempermudah untuk diakses oleh banyak platform dikomputer. Dalam proses rekayasa kebutuhan diperoleh 54 kebutuhan fungsionalitas dengan 4 aktor didalamnya dan 1 kebutuhan non fungsional. Hasil pengujian blackbox testing diperoleh 100% valid terhadap fungsionalitas aplikasi. Hasil pengujian whitebox testing diperoleh tingkat probabilitas rendah terhadap eror. Dan pada pengujian non fungsional compability menunjukkan aplikasi tidak menemui eror terhadap browser yang diuji. Berdasarkan hasil tersebut dapat disimpulkan bahwa aplikasi ini layak menjadi solusi dari permasalahan yang terjadi.

**Kata kunci :** web, waterfall, aplikasi pemesanan jasa, instruktur olahraga

## ABSTRACT

Tiara Rossa Diassananda ,Pengembangan Aplikasi Pemesanan Jasa Instruktur Olahraga Kota Malang

**Pembimbing:** Bayu Priyambadha, S.Kom, M.Kom dan Arief Andy Soebroto, S.T, M.Kom

*Sport is one of the activities to improve the quality of life and physical and spiritual health. In the present era in doing sport activities can be given various facilities including: imitating tutorials on youtube or hiring a sports instructor. From the results of interviews with 14 sports instructors in the City of Malang, the obstacles to marketing their services were the limited information dissemination and rapid competition and development in the world of sports instructors. And then the survey result of respondents in the city of Malang which is divided into 3 circles are public, students and coeds of men and women, 83.8% of them like zumba, aerobic, yoga, gym, swimming, muaythai and 60.6% of them need a sports instructor in conducting the sport activities but 63.4% of them encountered problems in the search for sports instructors are difficult to find a profile instructor, lack of reference, difficult to find an instructor who can adapt to our schedule and lack of instructor pricing information per training session. The developed solution is to develop a web-based sports instructor service application with waterfall software development cycle. Applications built using a web-based basis make it easy to access many platforms on the computer. In the process of engineering needs obtained 54 your functionality with 4 actors inside and 1 non functional needs. Test results blackbox testing obtained 100% valid against the functionality of the application. Test results of whitebox testing were obtained low probability levels against the error. And test result non functional testing compability indicates the application does not encounter errors against the browser tested. Based on these results it can be concluded that this application deserves to be the solution of the problems that occur.*

**Keywords:** web, waterfall, service order application, sports Instructor

## DAFTAR ISI

PENGESAHAN .....	<b>Error! Bookmark not defined.</b>
PERNYATAAN ORISINALITAS .....	<b>Error! Bookmark not defined.</b>
PRAKATA.....	iii
ABSTRAK .....	v
<i>ABSTRACT</i> .....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR .....	xx
DAFTAR LAMPIRAN .....	xxiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat.....	4
1.4.1 Bagi Pencari Jasa Instruktur Olahraga.....	4
1.4.2 Bagi Instruktur Olahraga .....	4
1.5 Batasan Masalah.....	4
1.6 Sistematika Pembahasan .....	4
BAB 2 LANDASAN KEPUSTAKAAN .....	6
2.1 Kajian Pustaka .....	6
2.2 Definisi Instruktur Olahraga.....	8
2.2.1 Pengertian Instruktur Olahraga .....	8
2.2.2 Manfaat Instruktur Olahraga .....	8
2.3 Pengertian Aplikasi Pencarian Jasa .....	9
2.4 Pengertian <i>Website</i> .....	9
2.5 Rekayasa Perangkat Lunak .....	9
2.6 Pengembangan Perangkat Lunak .....	9
2.7 <i>Software Development Live Cycle</i> .....	10
2.8 Metode Pengembangan <i>Waterfall</i> .....	10

2.9 Pendekatan Berorientasi Objek .....	11
2.10 Pemodelan <i>Unified Modelling Language</i> (UML).....	12
2.11 Bahasa Pemrograman PHP.....	15
2.12 HTML ( <i>Hyper Text Markup Language</i> ).....	15
2.13 CSS ( <i>Cascading Style Sheet</i> ) .....	15
2.14 <i>Bootstrap</i> .....	15
2.15 <i>Javascript</i> .....	16
2.16 <i>JQuery</i> .....	16
2.17 <i>Framework Codeigniter</i> .....	16
2.18 <i>Database MySql</i> .....	16
2.20 <i>Email Gateway</i> .....	17
2.21 Teori Pengujian .....	17
2.21.1 <i>Black-box Testing</i> .....	17
2.21.2 <i>White-box Testing</i> .....	17
<b>BAB 3 METODOLOGI</b> .....	<b>18</b>
3.1 Studi Kepustakaan .....	18
3.2 Pengumpulan Data .....	19
3.3 Rekayasa Kebutuhan.....	20
3.4 Perancangan .....	20
3.5 Implementasi .....	20
3.6 Pengujian .....	21
3.7 Kesimpulan dan Saran .....	21
<b>BAB 4 REKAYASA KEBUTUHAN</b> .....	<b>22</b>
4.1 Deskripsi Umum Sistem .....	22
4.2 Proses Rekayasa Kebutuhan .....	23
4.2.1 Elisitasi Kebutuhan .....	23
4.2.2 Spesifikasi Kebutuhan .....	27
4.2.3 Pemodelan Diagram <i>Use case</i> .....	36
4.2.4 <i>Use Case Scenario</i> .....	37
<b>BAB 5 PERANCANGAN DAN IMPLEMENTASI</b> .....	<b>66</b>
5.1 Perancangan.....	66
5.1.1 Perancangan <i>Sequence Diagram</i> .....	66

5.1.2 Perancangan <i>Class Diagram</i> .....	69
5.1.3 Perancangan <i>Algoritma</i> .....	70
5.1.4 Perancangan <i>Interface</i> .....	73
5.2 <i>Implementasi Sistem</i> .....	115
5.2.1 <i>Spesifikasi Sistem</i> .....	115
5.2.2 <i>Implementasi Source Code</i> .....	116
5.2.3 <i>Implementasi Interface</i> .....	126
BAB 6 PENGUJIAN .....	145
6.1 Pengujian <i>Whitebox Testing</i> .....	145
6.1.1 Pengujian <i>Whitebox</i> Kelas <i>Order Controller</i> untuk <i>method</i> <i>konfirmasi_pembayaran()</i> .....	145
6.1.2 Pengujian <i>Whitebox</i> Kelas <i>Members Controller</i> untuk <i>method</i> <i>add_member()</i> .....	148
6.1.3 Pengujian <i>Whitebox</i> Kelas <i>Events Controller</i> untuk <i>method</i> <i>add_event()</i> .....	150
6.1.4 Pengujian <i>Whitebox</i> Kelas <i>Order Controller</i> untuk <i>method</i> <i>confirm_instruktur()</i> .....	152
6.1.5 Pengujian <i>Whitebox</i> Kelas <i>Order Controller</i> untuk <i>method</i> <i>pesen_reguler()</i> .....	155
6.1.6 Hasil Analisis Pengujian <i>Whitebox Testing</i> .....	157
6.2 Pengujian <i>Blackbox Testing</i> .....	157
6.2.1 Pengujian <i>Blackbox</i> Fungsi <i>Register Member</i> .....	158
6.2.2 Pengujian <i>Blackbox</i> Fungsi <i>Register Instruktur Olahraga</i> .....	159
6.2.3 Pengujian <i>Blackbox</i> Fungsi <i>Login</i> .....	161
6.2.4 Pengujian <i>Blackbox</i> Fungsi <i>Mencari Instruktur Olahraga</i> .....	162
6.2.5 Pengujian <i>Blackbox</i> Fungsi <i>Mencari Kelas Event</i> .....	163
6.2.6 Pengujian <i>Blackbox Fungsi</i> <i>Melihat Detail Profil Instruktur Olahraga</i> .....	163
6.2.7 Pengujian <i>Blackbox</i> Fungsi <i>Melakukan Pemesanan Kelas Event</i> .....	164
6.2.8 Pengujian <i>Blackbox</i> Fungsi <i>Melihat Kode Pembayaran</i> .....	165
6.2.9 Pengujian <i>Blackbox</i> Fungsi <i>Melihat Profil Member</i> .....	165
6.2.10 Pengujian <i>Blackbox</i> Fungsi <i>Edit Profil Member</i> .....	166
6.2.11 Pengujian <i>Blackbox</i> Fungsi <i>Memberi Rating Instruktur</i> .....	167



6.2.12 Pengujian <i>Blackbox</i> Fungsi Konfirmasi Pembayaran .....	167
6.2.13 Pengujian <i>Blackbox</i> Fungsi Melakukan Pembatalan Pemesanan .....	168
6.2.14 Pengujian <i>Blackbox</i> Fungsi Hapus Akun <i>Member</i> .....	169
6.2.15 Pengujian <i>Blackbox</i> Fungsi Melihat Jadwal Latihan <i>Member</i> .....	169
6.2.16 Pengujian <i>Blackbox</i> Fungsi Melihat Profil Instruktur Olahraga .....	170
6.2.17 Pengujian <i>Blackbox</i> Fungsi Edit Profil Instruktur Olahraga .....	170
6.2.18 Pengujian <i>Blackbox</i> Fungsi Melakukan Pembuatan Kelas Reguler .....	171
6.2.19 Pengujian <i>Blackbox</i> Fungsi Melakukan Pembuatan Kelas Event .....	172
6.2.20 Pengujian <i>Blackbox</i> Fungsi Melihat Jadwal Latihan Instruktur Olahraga.....	173
6.2.21 Pengujian <i>Blackbox</i> Fungsi Melakukan Pembuatan Kelas Paket .....	174
6.2.22 Pengujian <i>Blackbox</i> Fungsi Hapus Akun Instruktur Olahraga .....	175
6.2.23 Pengujian <i>Blackbox</i> Fungsi Melihat Daftar Instruktur Terverifikasi .....	175
6.2.24 Pengujian <i>Blackbox</i> Fungsi Melihat Detail Profil Pendaftaran Instruktur .....	176
6.2.25 Pengujian <i>Blackbox</i> Fungsi Melakukan Validasi Pendaftaran Instruktur .....	176
6.2.26 Pengujian <i>Blackbox</i> Fungsi Menolak Pendaftaran Instruktur .....	177
6.2.27 Pengujian <i>Blackbox</i> Fungsi Melihat Daftar Pembayaran <i>Member</i> .....	177
6.2.28 Pengujian <i>Blackbox</i> Fungsi Melakukan Validasi Pembayaran .....	178
6.2.29 Pengujian <i>Blackbox</i> Fungsi Membatalkan Pembayaran .....	178
6.2.30 Pengujian <i>Blackbox</i> Fungsi Melihat Daftar Member .....	178
6.2.31 Pengujian <i>Blackbox</i> Fungsi Melihat Daftar Kelas Event .....	179
6.2.32 Pengujian <i>Blackbox</i> Fungsi Melakukan Validasi Kelas Event.....	179
6.2.33 Pengujian <i>Blackbox</i> Fungsi Menolak Kelas Event .....	180



6.2.34 Pengujian <i>Blackbox</i> Fungsi Logout .....	180
6.2.35 Pengujian <i>Blackbox</i> Fungsi Melihat Daftar Instruktur Belum Terverifikasi .....	180
6.2.36 Pengujian <i>Blackbox</i> Fungsi Menambahkan Instruktur.....	181
6.2.37 Pengujian <i>Blackbox</i> Fungsi Menambahkan <i>Member</i> .....	182
6.2.38 Pengujian <i>Blackbox</i> Fungsi Hapus <i>Member</i> .....	184
6.2.39 Pengujian <i>Blackbox</i> Fungsi Hapus Instruktur .....	184
6.2.40 Pengujian <i>Blackbox</i> Fungsi Melihat <i>Dashboard</i> .....	184
6.2.41 Pengujian <i>Blackbox</i> Fungsi Edit Instruktur .....	185
6.2.42 Pengujian <i>Blackbox</i> Fungsi Edit <i>Member</i> .....	186
6.2.43 Pengujian <i>Blackbox</i> Fungsi Edit Foto .....	187
6.2.44 Pengujian <i>Blackbox</i> Fungsi Mencari Kelas Reguler.....	188
6.2.45 Pengujian <i>Blackbox</i> Fungsi Mencari Kelas Paket.....	189
6.2.46 Pengujian <i>Blackbox</i> Fungsi Mencari Berdasarkan Kriteria .....	189
6.2.47 Pengujian <i>Blackbox</i> Fungsi Melihat Daftar Pembayaran .....	189
6.2.48 Pengujian <i>Blackbox</i> Fungsi Melihat Detail Pemesanan .....	190
6.2.49 Pengujian <i>Blackbox</i> Fungsi Melihat Bukti Pembayaran .....	190
6.2.50 Pengujian <i>Blackbox</i> Fungsi Melihat Detail Pemesanan <i>Member</i> .....	191
6.2.51 Pengujian <i>Blackbox</i> Fungsi Melihat Detail Pembayaran .....	191
6.2.52 Pengujian <i>Blackbox</i> Fungsi Melihat Detail Pemesan .....	192
6.2.53 Pengujian <i>Blackbox</i> Fungsi Melakukan Pemesanan Kelas Reguler.....	192
6.2.54 Pengujian <i>Blackbox</i> Fungsi Melakukan Pemesanan Kelas Paket .....	194
6.1.55 Hasil Analisis Pengujian <i>Blackbox</i> Testing .....	197
6.3 Pengujian Non Fungsional .....	197
6.3.1 Hasil Analisis Pengujian Non Fungsional .....	197
BAB 7 PENUTUP.....	198
7.1 Kesimpulan.....	198
7.2 Saran .....	198
DAFTAR PUSTAKA.....	199



## DAFTAR TABEL

<i>Tabel 4.1 Identifikasi Aktor .....</i>	<i>27</i>
<i>Tabel 4.2 Penjelasan Aturan Penomoran Kebutuhan.....</i>	<i>29</i>
<i>Tabel 4.3 Daftar Kebutuhan Fungsional Guest .....</i>	<i>29</i>
<i>Tabel 4.4 Daftar Kebutuhan Fungsional Member.....</i>	<i>30</i>
<i>Tabel 4.5 Daftar Kebutuhan Fungsional Instruktur Olahraga .....</i>	<i>32</i>
<i>Tabel 4.6 Daftar Kebutuhan Fungsional Admin .....</i>	<i>34</i>
<i>Tabel 4.7 Daftar Kebutuhan Non Fungsional .....</i>	<i>36</i>
<i>Tabel 4.8 Use Case Scenario Guset Register sebagai Member .....</i>	<i>38</i>
<i>Tabel 4.9 Use Case Scenario Register Guest sebagai Instruktur Olahraga.....</i>	<i>38</i>
<i>Tabel 4.10 Use Case Scenario Login .....</i>	<i>39</i>
<i>Tabel 4.11 Use Case Scenario Mencari Instruktur Olahraga .....</i>	<i>40</i>
<i>Tabel 4.12 Use Case Scenario Mencari Kelas Event .....</i>	<i>40</i>
<i>Tabel 4.13 Use Case Scenario Melihat Detail Profil Instrukur Olahraga .....</i>	<i>41</i>
<i>Tabel 4.14 Use Case Scenario Melakukan Pemesanan Kelas Event .....</i>	<i>41</i>
<i>Tabel 4.15 Use Case Scenario Melihat Kode Pembayaran .....</i>	<i>42</i>
<i>Tabel 4.16 Use Case Scenario Melihat Profil Member .....</i>	<i>43</i>
<i>Tabel 4.17 Use Case Scenario Edit Profil Member .....</i>	<i>43</i>
<i>Tabel 4.18 Use Case Scenario Memberi Rating Instruktur .....</i>	<i>43</i>
<i>Tabel 4.19 Use Case Scenario Melakukan Konfirmasi Pembayaran .....</i>	<i>44</i>
<i>Tabel 4.20 Use Case Scenario Melakukan Pembatalan Pemesanan.....</i>	<i>44</i>
<i>Tabel 4.21 Use Case Scenario Hapus Akun Member .....</i>	<i>45</i>
<i>Tabel 4.22 Use Case Scenario Melihat Jadwal Latihan Member .....</i>	<i>45</i>
<i>Tabel 4.23 Use Case Scenario Melihat Profil Instruktur Olahraga .....</i>	<i>46</i>
<i>Tabel 4.24 Use Case Scenario Edit Profil Instruktur Olahraga .....</i>	<i>46</i>
<i>Tabel 4.25 Use Case Scenario Melakukan Pembuatan Kelas Reguler .....</i>	<i>47</i>
<i>Tabel 4.26 Use Case Scenario Melakukan Pembuatan Kelas Event .....</i>	<i>47</i>
<i>Tabel 4.27 Use Case Scenario Melihat Jadwal Latihan Instruktur .....</i>	<i>48</i>
<i>Tabel 4.28 Use Case Scenario Melakukan Pembuatan Kelas Paket.....</i>	<i>48</i>
<i>Tabel 4.29 Use Case Scenario Hapus Akun Instruktur .....</i>	<i>49</i>
<i>Tabel 4.30 Use Case Scenario Melihat Daftar Instruktur Terverifikasi.....</i>	<i>49</i>
<i>Tabel 4.31 Use Case Scenario Melihat Detail Profil Pendaftaran Instruktur .....</i>	<i>50</i>



<i>Tabel 4.32 Use Case Scenario Melakukan Validasi Pendaftaran Instruktur .....</i>	<i>50</i>
<i>Tabel 4.33 Use Case Scenario Menolak Pendaftaran Instruktur.....</i>	<i>51</i>
<i>Tabel 4.34 Use Case Scenario Melihat Daftar Pembayaran Member .....</i>	<i>51</i>
<i>Tabel 4.35 Use Case Scenario Melakukan Validasi Pembayaran.....</i>	<i>52</i>
<i>Tabel 4.36 Use Case Scenario Membatalkan Pembayaran .....</i>	<i>52</i>
<i>Tabel 4.37 Use Case Scenario Melihat Daftar Member .....</i>	<i>52</i>
<i>Tabel 4.38 Use Case Scenario Melihat Daftar Kelas Event .....</i>	<i>53</i>
<i>Tabel 4.39 Use Case Scenario Melakukan Validasi Kelas Event.....</i>	<i>53</i>
<i>Tabel 4.40 Use Case Scenario Menolak Kelas Event .....</i>	<i>54</i>
<i>Tabel 4.41 Use Case Scenario Logout.....</i>	<i>54</i>
<i>Tabel 4.42 Use Case Scenario Melihat Daftar Instruktur Belum Terverifikasi .....</i>	<i>54</i>
<i>Tabel 4.43 Use Case Scenario Menambahkan Instruktur.....</i>	<i>55</i>
<i>Tabel 4.44 Use Case Scenario Menambahkan Member.....</i>	<i>56</i>
<i>Tabel 4.45 Use Case Scenario Hapus Member.....</i>	<i>56</i>
<i>Tabel 4.46 Use Case Scenario Hapus Instruktur.....</i>	<i>57</i>
<i>Tabel 4.47 Use Case Scenario Melihat Dashboard.....</i>	<i>57</i>
<i>Tabel 4.48 Use Case Scenario Edit Instruktur .....</i>	<i>58</i>
<i>Tabel 4.49 Use Case Scenario Edit Member .....</i>	<i>58</i>
<i>Tabel 4.50 Use Case Scenario Edit Foto.....</i>	<i>59</i>
<i>Tabel 4.51 Use Case Scenario Mencari Kelas Reguler .....</i>	<i>59</i>
<i>Tabel 4.52 Use Case Scenario Mencari Kelas Paket .....</i>	<i>60</i>
<i>Tabel 4.53 Use Case Scenario Mencari Berdasarkan Kriteria .....</i>	<i>60</i>
<i>Tabel 4.54 Use Case Scenario Melihat Daftar Pembayaran .....</i>	<i>61</i>
<i>Tabel 4.55 Use Case Scenario Melihat Detail Pemesanan.....</i>	<i>61</i>
<i>Tabel 4.56 Use Case Scenario Melihat Bukti Pembayaran .....</i>	<i>61</i>
<i>Tabel 4.57 Use Case Scenario Melihat Detail Pemesanan Member.....</i>	<i>62</i>
<i>Tabel 4.58 Use Case Scenario Melihat Detail Pembayaran .....</i>	<i>62</i>
<i>Tabel 4.59 Use Case Scenario Melihat Detail Pemesan .....</i>	<i>63</i>
<i>Tabel 4.60 Use Case Scenario Melakukan Pemesanan Kelas Paket.....</i>	<i>63</i>
<i>Tabel 5.1 Algoritma Konfirmasi Pembayaran dibawah ini.....</i>	<i>70</i>
<i>Tabel 5.2 Algoritma Konfirmasi Pembayaran .....</i>	<i>70</i>



Tabel 5.3 <i>Algoritma</i> Mengkonfirmasi .....	70
Tabel 5.4 <i>Algoritma</i> Mengkonfirmasi Pendaftaran Instruktur .....	70
Tabel 5.5 <i>Algoritma</i> Melakukan Pemesanan Reguler pada dibawah ini. ....	71
Tabel 5.6 <i>Algoritma</i> Melakukan Pemesanan Reguler .....	71
Tabel 5.7 <i>Algoritma</i> Menambahkan .....	71
Tabel 5.8 <i>Algoritma</i> Menambahkan <i>Member</i> .....	72
Tabel 5.9 <i>Algoritma</i> Membuat Kelas .....	72
Tabel 5.10 <i>Algoritma</i> Membuat Kelas Event .....	72
Tabel 5.11 Keterangan Perancangan <i>Interface</i> Halaman Daftar Pendaftar Instruktur. .....	73
Tabel 5.12 Keterangan Perancangan <i>Interface</i> Halaman Daftar Pendaftar Instruktur .....	74
Tabel 5.13 Keterangan Perancangan <i>Interface</i> Halaman Membuat <i>Member</i> . ....	75
Tabel 5.14 Keterangan Perancangan <i>Interface</i> Halaman Membuat <i>Member</i> .....	75
Tabel 5.15 Keterangan Perancangan <i>Interface</i> Halaman Membuat Instruktur. ....	76
Tabel 5.16 Keterangan Perancangan <i>Interface</i> Halaman Membuat Instruktur .....	76
Tabel 5.17 Keterangan Perancangan <i>Interface</i> Halaman Buat Kelas Event.....	77
Tabel 5.18 Keterangan Perancangan <i>Interface</i> Halaman Buat Kelas Event.....	78
Tabel 5.19 Keterangan Perancangan <i>Interface</i> Halaman Pemesanan Kelas Event. ...	79
Tabel 5.20 Keterangan Perancangan <i>Interface</i> Halaman Pemesanan Kelas Event ....	79
Tabel 5.21 Keterangan Perancangan <i>Interface</i> Halaman Lihat Event. ....	80
Tabel 5.22 Keterangan Perancangan <i>Interface</i> Halaman Lihat Event .....	80
Tabel 5.23 Keterangan Perancangan <i>Interface</i> Halaman Lihat Instruktur Terverifikasi. .....	81
Tabel 5.24 Keterangan Perancangan <i>Interface</i> Halaman Lihat Instruktur Terverifikasi .....	81
Tabel 5.25 Keterangan Perancangan <i>Interface</i> Halaman Lihat <i>Member</i> . ....	82
Tabel 5.26 Keterangan Perancangan <i>Interface</i> Halaman Lihat <i>Member</i> .....	82
Tabel 5.27 Keterangan Perancangan <i>Interface</i> Halaman <i>Register Member</i> . ....	84
Tabel 5.28 Keterangan Perancangan <i>Interface</i> Halaman <i>Register Member</i> .....	84
Tabel 5.29 Keterangan Perancangan <i>Interface</i> Halaman <i>Register</i> Instruktur.....	85
Tabel 5.30 Keterangan Perancangan <i>Interface</i> Halaman <i>Register</i> Instruktur.....	85
Tabel 5.31 Keterangan Perancangan <i>Interface</i> Halaman <i>Login</i> . ....	86



Tabel 5.32 Keterangan Perancangan <i>Interface</i> Halaman <i>Login</i> .....	86
Tabel 5.33 Keterangan Perancangan <i>Interface</i> Halaman Utama/Pencarian.....	87
Tabel 5.34 Keterangan Perancangan <i>Interface</i> Halaman Utama/Pencarian .....	87
Tabel 5.35 Keterangan Perancangan <i>Interface</i> Halaman Edit Profil Instruktur. ....	88
Tabel 5.36 Keterangan Perancangan <i>Interface</i> Halaman Edit Profil Instruktur .....	88
Tabel 5.37 Keterangan Perancangan <i>Interface</i> Halaman Edit Profil <i>Member</i> . ....	89
Tabel 5.38 Keterangan Perancangan <i>Interface</i> Halaman Edit Profil <i>Member</i> .....	90
Tabel 5.39 Keterangan Perancangan <i>Interface</i> Halaman Membuat Instruktur. ....	91
Tabel 5.40 Keterangan Perancangan <i>Interface</i> Halaman Membuat Instruktur .....	91
Tabel 5.41 Keterangan Perancangan <i>Interface</i> Halaman Edit <i>Member</i> . ....	92
Tabel 5.42 Keterangan Perancangan <i>Interface</i> Halaman Edit <i>Member</i> .....	93
Tabel 5.43 Keterangan Perancangan <i>Interface</i> Halaman <i>Login Admin</i> . ....	93
Tabel 5.44 Keterangan Perancangan <i>Interface</i> Halaman <i>Login Admin</i> .....	94
Tabel 5.45 Keterangan Perancangan <i>Interface</i> Halaman Lihat Profil Instruktur.....	94
Tabel 5.46 Keterangan Perancangan <i>Interface</i> Halaman Lihat Profil Instruktur.....	94
Tabel 5.47 Keterangan Perancangan <i>Interface</i> Halaman Lihat Profil <i>Member</i> . ....	95
Tabel 5.48 Keterangan Perancangan <i>Interface</i> Halaman Lihat Profil <i>Member</i> .....	95
Tabel 5.49 Keterangan Perancangan <i>Interface</i> Halaman Lihat Detail Profil Instruktur. .....	96
Tabel 5.50 Keterangan Perancangan <i>Interface</i> Halaman Lihat Detail Profil Instruktur .....	97
Tabel 5.51 Keterangan Perancangan <i>Interface</i> Halaman Edit Foto.....	98
Tabel 5.52 Keterangan Perancangan <i>Interface</i> Halaman Edit Foto.....	98
Tabel 5.53 Keterangan Perancangan <i>Interface</i> Buat Kelas Reguler.....	98
Tabel 5.54 Keterangan Perancangan <i>Interface</i> Buat Kelas Reguler .....	99
Tabel 5.55 Keterangan Perancangan <i>Interface</i> Buat Kelas Paket. ....	100
Tabel 5.56 Keterangan Perancangan <i>Interface</i> Buat Kelas Paket .....	100
Tabel 5.57 Keterangan Perancangan <i>Interface</i> Halaman Lihat Daftar Pembayaran. 101	
Tabel 5.58 Keterangan Perancangan <i>Interface</i> Halaman Lihat Daftar Pembayaran 101	
Tabel 5.59 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan. ....	102
Tabel 5.60 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan .....	102
Tabel 5.61 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan. ....	103
Tabel 5.62 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan .....	103



Tabel 5.63 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan.....	103
Tabel 5.64 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan.....	104
Tabel 5.65 Keterangan Perancangan <i>Interface</i> Konfirmasi Pembayaran. ....	104
Tabel 5.66 Keterangan Perancangan <i>Interface</i> Konfirmasi Pembayaran .....	105
Tabel 5.67 Keterangan Perancangan <i>Interface</i> Lihat Daftar Pembayaran <i>Member</i> . ..	106
Tabel 5.68 Keterangan Perancangan <i>Interface</i> Lihat Daftar Pembayaran <i>Member</i> . ..	106
Tabel 5.69 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan <i>Member</i> . .	107
Tabel 5.70 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesanan <i>Member</i> ...	107
Tabel 5.71 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesan.....	108
Tabel 5.72 Keterangan Perancangan <i>Interface</i> Lihat Detail Pemesan.....	108
Tabel 5.73 Keterangan Perancangan <i>Interface</i> Lihat Detail Pembayaran. ....	109
Tabel 5.74 Keterangan Perancangan <i>Interface</i> Lihat Detail Pembayaran .....	109
Tabel 5.75 Keterangan Perancangan <i>Interface</i> Halaman Pemesanan Kelas Paket. .	109
Tabel 5.76 Keterangan Perancangan <i>Interface</i> Halaman Pemesanan Kelas Paket ..	110
Tabel 5.77 Keterangan Perancangan <i>Interface</i> Halaman Pemesanan Kelas Reguler. .....	111
Tabel 5.78 Keterangan Perancangan <i>Interface</i> Halaman Pemesanan Kelas Reguler	111
Tabel 5.79 Keterangan Perancangan <i>Interface</i> Halaman Lihat Jadwal <i>Member</i> .....	112
Tabel 5.80 Keterangan Perancangan <i>Interface</i> Halaman Lihat Jadwal <i>Member</i> .....	112
Tabel 5.81 Keterangan Perancangan <i>Interface</i> Halaman Lihat Jadwal Instruktur Olahraga.....	113
Tabel 5.82 Keterangan Perancangan <i>Interface</i> Halaman Lihat Jadwal Instruktur Olahraga.....	113
Tabel 5.83 Keterangan Perancangan <i>Interface</i> Halaman <i>Dashboard Admin</i> .....	114
Tabel 5.84 Keterangan Perancangan <i>Interface</i> Halaman <i>Dashboard Admin</i> .....	114
Tabel 5.85 <i>Spesifikasi</i> Perangkat Keras .....	115
Tabel 5.86 <i>Spesifikasi</i> Perangkat Lunak.....	116
Tabel 5.87 <i>Implementasi Source Code</i> Konfirmasi Pembayaran dibawah ini. ....	116
Tabel 5.88 <i>Implementasi Source Code</i> Konfirmasi Pembayaran.....	116
Tabel 5.89 <i>Implementasi Source Code</i> Mengkonfirmasi Pendaftaran Instruktur dibawah ini. ....	117
Tabel 5.90 <i>Implementasi Source Code</i> Mengkonfirmasi Pendaftaran Instruktur ...	118
Tabel 5.91 <i>Implementasi Source Code</i> Melakukan Pemesanan Reguler dibawah ini. .....	119



Tabel 5.92 Implementasi Source Code Melakukan Pemesanan Reguler .....	120
Tabel 5.93 Implementasi Source Code Membuat Member dibawah ini.....	121
Tabel 5.94 Implementasi Source Code Membuat Member .....	121
Tabel 5.95 Implementasi Source Code Membuat Jadwal Paket Event dibawah ini.	123
Tabel 5.96 Implementasi Source Code Membuat Jadwal Paket Event .....	124
Tabel 6.1 Pseudocode Pengujian Whitebox Method konfirmasi_pembayaran..	146
Tabel 6.2 Pseudocode Pengujian Whitebox Method add_member() .....	148
Tabel 6.3 Pseudocode Pengujian Whitebox Method add_event().....	151
Tabel 6.4 Pseudocode Pengujian Whitebox Method confirm_instruktur() .....	153
Tabel 6.5 Pseudocode Pengujian Whitebox Method pesan_reguler ().....	155
Tabel 6.6 Pengujian Blackbox Fungsi Register Member .....	158
Tabel 6.7 Pengujian Blackbox Fungsi Register Member .....	158
Tabel 6.8 Pengujian Blackbox Fungsi Register Member .....	159
Tabel 6.9 Pengujian Blackbox Fungsi Register Instruktur Olahraga.....	159
Tabel 6.10 Pengujian Blackbox Fungsi Register Instruktur Olahraga.....	160
Tabel 6.11 Pengujian Blackbox Fungsi Register Instruktur Olahraga.....	160
Tabel 6.12 Pengujian Blackbox Fungsi Login .....	161
Tabel 6.13 Pengujian Blackbox Fungsi Login .....	161
Tabel 6.14 Pengujian Blackbox Fungsi Login .....	162
Tabel 6.15 Pengujian Blackbox Fungsi Mencari Instruktur Olahraga .....	162
Tabel 6.16 Pengujian Blackbox Fungsi Mencari Instruktur Olahraga .....	162
Tabel 6.17 Pengujian Blackbox Fungsi Mencari Kelas Event .....	163
Tabel 6.18 Pengujian Blackbox Fungsi Melihat Detail Profil Instruktur Olahraga ...	163
Tabel 6.19 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Event .....	164
Tabel 6.20 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Event .....	164
Tabel 6.21 Pengujian Blackbox Fungsi Meihat Kode Pembayaran .....	165
Tabel 6.22 Pengujian Blackbox Fungsi Melihat Profil Member.....	165
Tabel 6.23 Pengujian Blackbox Fungsi Edit Profil Member .....	166
Tabel 6.24 Pengujian Blackbox Fungsi Edit Profil Member .....	166
Tabel 6.25 Pengujian Blackbox Fungsi Memberi Rating Instruktur .....	167
Tabel 6.26 Pengujian Blackbox Fungsi Konfirmasi Pembayaran.....	167
Tabel 6.27 Pengujian Blackbox Fungsi Konfirmasi Pembayaran .....	168



<i>Tabel 6.28 Pengujian Blackbox Melakukan Pembatalan Pemesanan.....</i>	<i>168</i>
<i>Tabel 6.29 Pengujian Blackbox Fungsi Hapus Akun Member .....</i>	<i>169</i>
<i>Tabel 6.30 Pengujian Blackbox Fungsi Melihat Jadwal Latihan Member.....</i>	<i>169</i>
<i>Tabel 6.31 Pengujian Blackbox Fungsi Melihat Profil Instruktur Olahraga .....</i>	<i>170</i>
<i>Tabel 6.32 Pengujian Blackbox Fungsi Edit Profil Instruktur Olahraga .....</i>	<i>170</i>
<i>Tabel 6.33 Pengujian Blackbox Fungsi Edit Profil Instruktur Olahraga .....</i>	<i>171</i>
<i>Tabel 6.34 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Reguler .....</i>	<i>171</i>
<i>Tabel 6.35 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Reguler .....</i>	<i>172</i>
<i>Tabel 6.36 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Event .....</i>	<i>172</i>
<i>Tabel 6.37 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Event .....</i>	<i>173</i>
<i>Tabel 6.38 Pengujian Blackbox Fungsi Melihat Jadwal Latihan Instruktur Olahraga .....</i>	<i>174</i>
<i>Tabel 6.39 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Paket .....</i>	<i>174</i>
<i>Tabel 6.40 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Paket .....</i>	<i>174</i>
<i>Tabel 6.41 Pengujian Blackbox Fungsi Hapus Akun Instruktur Olahraga.....</i>	<i>175</i>
<i>Tabel 6.42 Pengujian Blackbox Fungsi Melihat Daftar Instruktur Terverifikasi .....</i>	<i>175</i>
<i>Tabel 6.43 Pengujian Blackbox Fungsi Melihat Detail Profil Pendaftaran Instruktur .....</i>	<i>176</i>
<i>Tabel 6.44 Pengujian Blackbox Fungsi Melakukan Validasi Pendaftaran Instruktur</i>	<i>176</i>
<i>Tabel 6.45 Pengujian Blackbox Fungsi Menolak Pendaftaran Instruktur .....</i>	<i>177</i>
<i>Tabel 6.46 Pengujian Blackbox Fungsi Melihat Daftar Pembayaran Member .....</i>	<i>177</i>
<i>Tabel 6.47 Pengujian Blackbox Fungsi Melakukan Validasi Pembayaran .....</i>	<i>178</i>
<i>Tabel 6.48 Pengujian Blackbox Fungsi Membatalkan Pembayaran .....</i>	<i>178</i>
<i>Tabel 6.49 Pengujian Blackbox Fungsi Melihat Daftar Member.....</i>	<i>178</i>
<i>Tabel 6.50 Pengujian Blackbox Fungsi Melihat Daftar Kelas Event .....</i>	<i>179</i>
<i>Tabel 6.51 Pengujian Blackbox Fungsi Melakukan Validasi Kelas Event .....</i>	<i>179</i>
<i>Tabel 6.52 Pengujian Blackbox Fungsi Menolak Kelas Event .....</i>	<i>180</i>
<i>Tabel 6.53 Pengujian Blackbox Fungsi Logout.....</i>	<i>180</i>
<i>Tabel 6.54 Pengujian Blackbox Fungsi Melihat Daftar Instruktur Belum Terverifikasi .....</i>	<i>180</i>
<i>Tabel 6.55 Pengujian Blackbox Fungsi Menambahkan Instruktur .....</i>	<i>181</i>
<i>Tabel 6.56 Pengujian Blackbox Fungsi Menambahkan Instruktur .....</i>	<i>181</i>
<i>Tabel 6.57 Pengujian Blackbox Fungsi Menambahkan Instruktur .....</i>	<i>182</i>



<i>Tabel 6.58 Pengujian Blackbox Fungsi Menambahkan Member</i> .....	182
<i>Tabel 6.59 Pengujian Blackbox Fungsi Menambahkan Member</i> .....	183
<i>Tabel 6.60 Pengujian Blackbox Fungsi Menambahkan Member</i> .....	183
<i>Tabel 6.61 Pengujian Blackbox Fungsi Hapus Member</i> .....	184
<i>Tabel 6.62 Pengujian Blackbox Fungsi Hapus Instruktur</i> .....	184
<i>Tabel 6.63 Pengujian Blackbox Fungsi Melihat Dashboard</i> .....	184
<i>Tabel 6.64 Pengujian Blackbox Fungsi Edit Instruktur</i> .....	185
<i>Tabel 6.65 Pengujian Blackbox Fungsi Edit Instruktur</i> .....	185
<i>Tabel 6.66 Pengujian Blackbox Fungsi Edit Member</i> .....	186
<i>Tabel 6.67 Pengujian Blackbox Fungsi Edit Member</i> .....	187
<i>Tabel 6.68 Pengujian Blackbox Fungsi Edit Foto</i> .....	187
<i>Tabel 6.69 Pengujian Blackbox Fungsi Edit Foto</i> .....	188
<i>Tabel 6.70 Pengujian Blackbox Fungsi Mencari Kelas Reguler</i> .....	188
<i>Tabel 6.71 Pengujian Blackbox Fungsi Mencari Kelas Paket</i> .....	189
<i>Tabel 6.72 Pengujian Blackbox Fungsi Mencari Berdasarkan Kriteria</i> .....	189
<i>Tabel 6.73 Pengujian Blackbox Fungsi Melihat Daftar Pembayaran</i> .....	189
<i>Tabel 6.74 Pengujian Blackbox Fungsi Melihat Detail Pemesanan</i> .....	190
<i>Tabel 6.75 Pengujian Blackbox Fungsi Melihat Bukti Pembayaran</i> .....	190
<i>Tabel 6.76 Pengujian Blackbox Fungsi Melihat Detail Pemesanan Member</i> .....	191
<i>Tabel 6.77 Pengujian Blackbox Fungsi Melihat Detail Pembayaran</i> .....	191
<i>Tabel 6.78 Pengujian Blackbox Fungsi Melihat Detail Pemesanan Member</i> .....	192
<i>Tabel 6.79 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Reguler</i> .....	192
<i>Tabel 6.80 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Reguler</i> .....	193
<i>Tabel 6.81 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Reguler</i> .....	193
<i>Tabel 6.82 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Paket</i> .....	194
<i>Tabel 6.83 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Paket</i> .....	195
<i>Tabel 6.84 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Paket</i> .....	195
<i>Tabel 6.85 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Paket</i> .....	196

## DAFTAR GAMBAR

<i>Gambar 2.1 Waterfall Model (Sommerville,2011)</i> .....	11
<i>Gambar 2.2 Contoh Class Diagram</i> .....	13
<i>Gambar 2.3 Association</i> .....	13
<i>Gambar 2.4 Composition</i> .....	14
<i>Gambar 2.5 Agregation</i> .....	14
<i>Gambar 3.1 Diagram Alir Metodologi Penelitian</i> .....	18
<i>Gambar 4.1 Struktur Diagram Rekayasa Kebutuhan</i> .....	22
<i>Gambar 4.2 Standar Operational Procedure(SOP) As Is Pemesanan Jasa Instruktur Olahraga Kota Malang</i> .....	24
<i>Gambar 4.3 Standar Operational Procedure(SOP) To Be Pemesanan Jasa Instruktur Olahraga Kota Malang</i> .....	26
<i>Gambar 4.4 Aturan Penomoran Kebutuhan</i> .....	29
<i>Gambar 4.5 Use Case Diagram</i> .....	37
<i>Gambar 5.1 Struktur Diagram Perancangan dan Implementasi</i> .....	66
<i>Gambar 5.2 Sequence Diagram Menambahkan Member</i> .....	67
<i>Gambar 5.3 Sequence Diagram Membuat Kelas Event</i> .....	68
<i>Gambar 5.4 Sequence Diagram Melakukan Pemesanan Kelas Reguler</i> .....	69
<i>Gambar 5.5 Class Diagram</i> .....	69
<i>Gambar 5.6 Perancangan Interface Halaman Lihat Daftar Pendaftar Instruktur</i> .....	73
<i>Gambar 5.7 Perancangan Interface Halaman Membuat Member</i> .....	75
<i>Gambar 5.8 Perancangan Interface Halaman Membuat Instruktur</i> .....	76
<i>Gambar 5.9 Perancangan Interface Halaman Buat Kelas Event</i> .....	78
<i>Gambar 5.10 Perancangan Interface Halaman Pemesanan Kelas Event</i> .....	79
<i>Gambar 5.11 Perancangan Interface Halaman Lihat Event</i> .....	80
<i>Gambar 5.12 Perancangan Interface Halaman Lihat Instruktur Terverifikasi</i> .....	81
<i>Gambar 5.13 Perancangan Interface Halaman Lihat Member</i> .....	82
<i>Gambar 5.14 Perancangan Interface Halaman Register Member</i> .....	84
<i>Gambar 5.15 Perancangan Interface Halaman Register Instruktur</i> .....	85
<i>Gambar 5.16 Perancangan Interface Halaman Login</i> .....	86
<i>Gambar 5.17 Perancangan Interface Halaman Utama/Pencarian</i> .....	87
<i>Gambar 5.18 Perancangan Interface Halaman Edit Profil Instruktur</i> .....	88

Gambar 5.19 Perancangan Interface Halaman Edit Profil Member .....	90
Gambar 5.20 Perancangan Interface Halaman Edit Instruktur .....	91
Gambar 5.22 Perancangan Interface Halaman Login Admin	<b>Error! Bookmark not defined.</b>
Gambar 5.23 Perancangan Interface Halaman Lihat Profil Instruktur .....	94
Gambar 5.24 Perancangan Interface Halaman Lihat Profil Member .....	95
Gambar 5.25 Perancangan Interface Halaman Lihat Detail Profil Instruktur .....	<b>Error! Bookmark not defined.</b>
Gambar 5.26 Perancangan Interface Halaman Edit Foto.....	98
Gambar 5.27 Perancangan Interface Halaman Buat Kelas Reguler .....	99
Gambar 5.28 Perancangan Interface Halaman Buat Kelas Paket .....	100
Gambar 5.29 Perancangan Interface Halaman Lihat Daftar Pembayaran .....	101
Gambar 5.30 Perancangan Interface Lihat Detail Pemesanan.....	102
Gambar 5.31 Perancangan Interface Lihat Bukti Pembayaran .....	103
Gambar 5.32 Perancangan Interface Cek Kode Pembayaran	<b>Error! Bookmark not defined.</b>
Gambar 5.33 Perancangan Interface Konfirmasi Pembayaran .....	105
Gambar 5.34 Perancangan Interface Lihat Daftar Pembayaran Member .....	106
Gambar 5.35 Perancangan Interface Lihat Detail Pemesanan Member.....	107
Gambar 5.36 Perancangan Interface Lihat Detail Pemesan.....	108
Gambar 5.37 Perancangan Interface Lihat Detail Pembayaran .....	109
Gambar 5.38 Perancangan Interface Halaman Pemesanan Kelas Paket.....	110
Gambar 5.39 Perancangan Interface Halaman Pemesanan Kelas Reguler.....	111
Gambar 5.40 Perancangan Interface Halaman Lihat Jadwal Member .....	112
Gambar 5.41 Perancangan Interface Halaman Lihat Jadwal Instruktur Olahraga..	113
Gambar 5.42 Perancangan Interface Halaman Dashboard Admin .....	114
Gambar 5.43 Implementasi Interface Halaman Login Member dan Instruktur .	126
Gambar 5.45 Implementasi Interface Halaman Register Instruktur .....	127
Gambar 5.46 Implementasi Interface Halaman Utama	<b>Error! Bookmark not defined.</b>
Gambar 5.47 Implementasi Interface Halaman Pemesanan Kelas Event .....	128
Gambar 5.48 Implementasi Interface Halaman Ganti Foto	<b>Error! Bookmark not defined.</b>



Gambar 5.49 Implementasi Interface Halaman Lihat Daftar Instruktur Terverifikasi.....	129
Gambar 5.50 Implementasi Interface Halaman Membuat Instruktur.....	<b>Error! Bookmark not defined.</b>
Gambar 5.51 Implementasi Interface Halaman Edit Instruktur .....	130
Gambar 5.53 Implementasi Interface Halaman Lihat Detail Pendaftar Instruktur.....	131
Gambar 5.54 Implementasi Interface Halaman Lihat Daftar Member .....	<b>Error! Bookmark not defined.</b>
Gambar 5.55 Implementasi Interface Halaman Edit Member .....	132
Gambar 5.56 Implementasi Interface Halaman Lihat Daftar Event .....	<b>Error! Bookmark not defined.</b>
Gambar 5.57 Implementasi Interface Halaman Buat Event.....	<b>Error! Bookmark not defined.</b>
Gambar 5.58 Implementasi Interface Halaman Lihat Profil Instruktur .....	<b>Error! Bookmark not defined.</b>
Gambar 5.59 Implementasi Interface Halaman Lihat Profil Member .....	134
Gambar 5.60 Implementasi Interface Halaman Edit Profil Instruktur.....	<b>Error! Bookmark not defined.</b>
Gambar 5.61 Implementasi Interface Halaman Edit Profil Member .....	135
Gambar 5.62 Implementasi Interface Halaman Lihat Detail Profil Instruktur.....	<b>Error! Bookmark not defined.</b>
Gambar 5.63 Implementasi Interface Halaman Login Admin .....	136
Gambar 5.64 Implementasi Interface Halaman Buat Kelas Reguler .....	<b>Error! Bookmark not defined.</b>
Gambar 5.65 Implementasi Interface Halaman Buat Kelas Paket.....	<b>Error! Bookmark not defined.</b>
Gambar 5.66 Implementasi Interface Lihat Detail Pembayaran .....	137
Gambar 5.67 Implementasi Interface Lihat Detail Pemesanan Member .....	138
Gambar 5.68 Implementasi Interface Lihat Detail Pemesanan.....	<b>Error! Bookmark not defined.</b>
Gambar 5.69 Implementasi Interface Lihat Daftar Pembayaran Member .....	139
Gambar 5.70 Implementasi Interface Lihat Daftar Pembayaran.....	<b>Error! Bookmark not defined.</b>
Gambar 5.71 Implementasi Interface Lihat Detail Pemesanan.....	140
Gambar 5.72 Implementasi Interface Lihat Bukti Pembayaran.....	<b>Error! Bookmark not defined.</b>

*Gambar 5.73 Implementasi Interface Konfirmasi Pembayaran ..... 141*

*Gambar 5.74 Implementasi Interface Cek Kode Pembayaran***Error! Bookmark not defined.**

*Gambar 5.75 Implementasi Interface Halaman Pemesanan Kelas Paket ..... 142*

*Gambar 5.76 Implementasi Interface Halaman Pemesanan Kelas Reguler ....***Error! Bookmark not defined.**

*Gambar 5.77 Implementasi Interface Halaman Lihat Jadwal Member..... 143*

*Gambar 5.78 Implementasi Interface Halaman Lihat Jadwal Instruktur Olahraga .....* **Error! Bookmark not defined.**

*Gambar 5.79 Implementasi Interface Halaman Dashboard Admin ..... 144*

*Gambar 6.1 Struktur Diagram Pengujian.....145*

*Gambar 6.2 Flowgraph method konfirmasi\_pembayaran ..... 146*

*Gambar 6.3 Flowgraph method add\_member ..... 149*

*Gambar 6.4 Flowgraph method add\_event ..... 151*

*Gambar 6.5 Flowgraph method confirm\_instruktur ..... 153*

*Gambar 6.6 Flowgraph method pesan\_reguler..... 156*

## DAFTAR LAMPIRAN

Lampiran 1 Surat Telah Melakukan Wawancara ..... 201

Lampiran 2 Hasil Wawancara..... 202

Lampiran 3 Hasil Kuisisioner Online ..... 252





## BAB 1 PENDAHULUAN

### 1.1 Latar Belakang

Pada era sekarang ini olahraga menjadi sesuatu hal yang sangat populer, bisa dilihat dari banyaknya macam olahraga serta video tutorial cara berolahraga dengan benar. Namun seringkali masyarakat tidak puas dengan hasil yang didapat karena hanya dengan mengikuti tutorialnya saja belum tentu dapat mempraktikkannya dengan benar, sehingga banyak masyarakat yang sekarang ini menggunakan jasa instruktur olahraga untuk menunjang kegiatan olahraganya sehingga mendapat hasil yang maksimal dan sesuai keinginan.

Dilansir dari *Everydayhealth.com* petunjuk untuk memilih instruktur olahraga yang tepat adalah dengan menentukan tujuan yang ingin dicapai dengan cara berkonsultasi dengan instruktur, mencari referensi instruktur yang tepat, menentukan jadwal dan tempat latihan serta mengecek keahlian dan sertifikasi instruktur tersebut. Selain itu menjadi instruktur olahraga merupakan pekerjaan sampingan yang menguntungkan dikarenakan banyaknya ajang lomba yang ditawarkan, serta banyaknya masyarakat yang membutuhkan keahlian mereka. Maka tak jarang jika sekarang instruktur olahraga tidak melihat jenis kelamin dalam berbagai macam jenis olahraganya, namun melihat keprofesionalan dalam melatih. Menggunakan jasa instruktur olahraga akan membuat pelanggan termotivasi lebih giat untuk melakukan segala rangkaian program yang diinginkan dan juga membantu memantau hasil selama berlatih (Chocky, 2011). Berdasarkan hasil wawancara terhadap 14 instruktur olahraga di Kota Malang, disimpulkan bahwa permasalahan yang mereka hadapi dalam menjalankan pemasaran jasanya antara lain:

Penyebaran informasinya masih sebatas media social seperti *whatsapp, instagram, facebook* atau masih dari mulut ke mulut sehingga dirasa kurang efektif dalam proses pemasaran jasanya. Hal ini terjadi karena masyarakat yang ingin melakukan pemesanan jasa harus menghubungi instruktur lewat media social yang dimiliki sehingga terkadang instruktur merasa penyampaian informasi tentang jasa yang ditawarkan kurang efektif. Yang mana instruktur harus membalas pesan pada social media satu persatu untuk menjelaskan jasa yang ditawarkan, belum lagi jika pengguna jasa hanya ingin sekedar bertanya namun tidak serius untuk melakukan pemesanan.

Persaingan ketat dan perkembangan yang pesat dalam dunia pemasaran jasa instruktur olahraga, yang mana banyaknya instruktur pendatang baru yang lebih eksis di dunia social media membuat sebagian instruktur yang sudah berpengalaman namun belum banyak dikenal masyarakat akan kalah dalam segi pemasaran jasanya dibanding dengan instruktur yang masih fresh namun sudah banyak dikenal masyarakat.

Hal ini dikarenakan biasanya para instruktur senior masih jarang memiliki media social yang populer dimasyarakat untuk mengeksplor kemampuan dibandingkan instruktur yang masih fresh namun memiliki wadah pada media social untuk mengeksplor kemampuan. Sehingga mengakibatkan masyarakat kurang familiar terhadap instruktur yang justru sudah memiliki pengalaman dalam jasanya. Berdasarkan penyebaran kuisisioner yang dilakukan pada penelitian ini, terhadap 70 responden di Kota Malang yang terdiri dari 31% laki-laki dan 69% perempuan, rentang umur 15-60 tahun dengan kriteria pekerjaan 67,6% mahasiswa, 15,5% pelajar dan 16,9% umum menyebutkan bahwa 83,8% dari mereka menyukai olahraga tertentu seperti *zumba, aerobic, yoga, muaythai, gym*, renang. Dan 60,6% dari mereka membutuhkan instruktur olahraga dalam melakukan kegiatan olahraga tersebut, namun sebesar 63,4% dari mereka menemui permasalahan dalam mencari dan melakukan pemesanan jasa terhadap instruktur olahraga tersebut, masalah yang sering terjadi diantara lain:

Sulitnya mencari profil instruktur, penyebab permasalahan ini dikarenakan sebagian penduduk Kota Malang hanya mengetahui jasa instruktur olahraga berdasarkan mulut ke mulut tanpa mengetahui profil sebenarnya tentang instruktur. Hal ini dapat memicu adanya kekecewaan para pengguna jasa terhadap hasil dari jenis layanan jasa, menggunakan jasa instruktur olahraga yang kurang berpengalaman dan bersertifikasi.

Kurangnya referensi, penyebab dari permasalahan ini adalah rata-rata penduduk Kota Malang yang kebanyakan merupakan warga pendatang sehingga rata-rata dari mereka masing masing asing terhadap pelayanan jasa instruktur olahraga di Kota Malang, serta belum adanya list daftar penyedia jasa instruktur olahraga tersebut. Banyak dari pengguna jasa yang belum memiliki referensi datang langsung pada tempat olahraga untuk mencari informasi tersebut namun seringkali setelah masuk pada kelas latihan, mereka merasa kurang nyaman dengan kelas latihan tersebut karena pesertanya terlalu banyak atau pun canggung terhadap peserta lain yang belum dikenal.

Sulit mencari instruktur yang dapat menyesuaikan dengan jadwal kita, permasalahan ini sering muncul karena biasanya para instruktur olahraga melakukan kerjasama dengan tempat olahraga untuk melakukan pemasaran jasanya, maka dari itu biasanya jadwal terkait pelayanan jasanya sudah dibuat oleh tempat olahraga tersebut yang terkadang jadwalnya tidak sesuai dengan jadwal yang kita inginkan. Dan seringkali pengguna jasa yang datang langsung ke tempat olahraga harus dikecewakan dengan jadwal kelas latihan yang sudah penuh sehingga mereka tidak dapat mengikuti kelas latihan tersebut.

Kurangnya informasi paket harga instruktur per sesi latihan, penyebab permasalahan dibagian ini adalah para instruktur olahraga di Kota Malang belum memiliki media promosi khusus untuk menyampaikan informasi harga paket untuk sesi latihan pada jasa yang ditawarkan, sehingga calon pengguna jasa kesulitan dalam mencari informasi tersebut.

Dalam hal ini memungkinkan pengguna jasa harus mencari langsung tentang informasi tersebut sendiri, dengan cara seperti datang pada tempat olahraga. Namun tempat olahraga tersebut hanya akan memberikan informasi terkait paket harga per sesi instruktur yang mengajar ditempat olahraga tersebut. Dan seringkali tempat olahraga tersebut tidak akan memberi informasi terkait kontak pribadi instruktur tersebut, sehingga pengguna jasa yang datang langsung ke tempat olahraga tidak mendapatkan informasi spesifik terkait instruktur olahraga

Berdasarkan masalah yang terjadi 88,7% dari mereka menginginkan untuk dibantu mengatasi masalah tersebut. Solusi yang didapat untuk mengatasi masalah tersebut adalah mengembangkan aplikasi untuk mencari dan melakukan pemesanan jasa instruktur olahraga Kota Malang.

Dalam pengembangan aplikasinya menggunakan *google maps api* untuk memilih lokasi latihan dengan tujuan agar lokasi latihan yang dimasukkan valid dan memanfaatkan *email gateway* untuk notifikasi jadwal latihan dengan tujuan agar pemberi dan instruktur olahraga tidak perlu membuka halaman web untuk hanya sekedar melihat jadwal, serta untuk media pengingat jadwal yang efektif.

Dalam penelitian yang berbasis web ini akan memakai *database sql* serta *framework codeigniter*. Pendekatan yang digunakan dalam pengembangan aplikasi ini adalah model *waterfall*. *Waterfall* adalah model pengembangan perangkat lunak yang paling teratur dikarenakan pengembang akan melakukan penjadwalan dan perencanaan semua proses sebelum pada tahap pengerjaan.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, maka ditarik beberapa rumusan masalah sebagai berikut:

1. Bagaimana hasil rekayasa kebutuhan aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web?
2. Bagaimana hasil perancangan aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web?
3. Bagaimana hasil implementasi aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web?
4. Bagaimana hasil pengujian aplikasi pemesanan jasa olahraga kota malang berbasis web dilihat dari kebutuhan fungsional dan non fungsionalnya?

## 1.3 Tujuan

Berdasarkan rumusan masalah sebelumnya, maka terdapat beberapa tujuan penulisan sebagai berikut:

1. Menganalisis dan memodelkan kebutuhan aplikasi untuk pemesanan jasa instruktur olahraga kota malang berbasis web.
2. Merancang aplikasi sesuai analisis dan pemodelan kebutuhan untuk aplikasi pemesanan jasa instruktur kota malang berbasis web.
3. Mengimplementasikan aplikasi berdasarkan perancangan yang telah dibuat.
4. Menguji aplikasi tersebut berdasarkan tingkat kebutuhan fungsional dan non fungsional.

## 1.4 Manfaat

Dengan adanya aplikasi tersebut diharapkan dapat memberi beberapa manfaat seperti berikut :

### 1.4.1 Bagi Pencari Jasa Instruktur Olahraga

- a. Mempermudah dalam mencari jasa instruktur olahraga yang sesuai dengan harapan.
- b. Memberikan kemudahan dalam pemesanan jasa instruktur olahraga sesuai keinginan.

### 1.4.2 Bagi Instruktur Olahraga

- a. Memudahkan dalam mencari pengguna jasa serta membantu mempromosikan keahlian dan program yang diberikan kepada pengguna jasa.
- b. Memberikan kemudahan dalam hal waktu dan tempat latihan dengan pengguna jasa.

## 1.5 Batasan Masalah

Berikut ini beberapa batasan masalah yang terdapat pada aplikasi tersebut, yaitu :

1. Aplikasi hanya akan menampilkan jasa instruktur olahraga yang berdomisili di Kota Malang.
2. Aplikasi hanya akan digunakan dalam pemesanan jasa instruktur olahraga di Kota Malang.
3. Aplikasi hanya akan digunakan dalam proses pemesanan instruktur olahraga *gym, aerobic, yoga, zumba, renang, muaythai, renang*.
4. Aplikasi dapat digunakan dalam menentukan tempat latihan meliputi : *indoor* dan *outdoor* dengan *google maps*.
5. Aplikasi dapat digunakan oleh pengguna meliputi : umum, mahasiswa, pelajar (SMP, SMA).

## 1.6 Sistematika Pembahasan

Penelitian ini disusun dalam beberapa bab pembahasan sebagai acuan mendasar, adapun rancangan sistematika pembahasan seperti berikut :

## **BAB I PENDAHULUAN**

Dalam bab ini berisi gambaran umum dari penelitian yang terdiri dari: latar belakang yang digunakan untuk mendeskripsikan masalah pada penelitian, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan pada proses pengembangan aplikasi pemesanan Jasa Instruktur Olahraga Kota Malang berbasis web.

## **BAB II LANDASAN KEPUSTAKAAN**

Dalam bab ini berisi uraian dan pembahasan tentang teori, konsep, model, metode, sistem atau analisis literatur ilmiah yang mendasari proses pengembangan aplikasi pemesanan Jasa Instruktur Olahraga Kota Malang berbasis web.

## **BAB III METODOLOGI PENELITIAN**

Dalam bab ini berisi metode atau teknik, pendekatan serta alur yang diterapkan dalam proses pengembangan aplikasi pemesanan Jasa Instruktur Kota Malang berbasis web.

## **BAB IV REKAYASA KEBUTUHAN**

Dalam bab ini berisi tentang analisis kebutuhan yang didapatkan dari hasil wawancara sehingga dapat mengidentifikasi actor yang berperan serta mengidentifikasi kebutuhan fungsional maupun non fungsional.

## **BAB V PERANCANGAN DAN IMPLEMENTASI**

Dalam bab ini berisi rancangan aplikasi dengan menggunakan pemodelan UML serta implementasi dari fungsi-fungsi aplikasi tersebut menggunakan *Framework CodeIgniter* dan menggunakan database *MySQL* sebagai media penyimpanan data pada proses pengembangan aplikasi pemesanan Jasa Instruktur Olahraga Kota Malang berbasis web.

## **BAB VI PENGUJIAN**

Dalam bab ini berisi tentang hasil pengujian dari proses pengembangan aplikasi pemesanan jasa Instruktur Olahraga Kota Malang berbasis web dengan menggunakan *blackbox testing* dan *whitebox testing* yang terdapat data dan analisis pengujiannya.

## **BAB VII PENUTUP**

Dalam bab ini berisi tentang kesimpulan dari proses pengembangan dan pengujian aplikasi pemesanan Jasa Instruktur Olahraga Kota Malang berbasis web serta saran pengembangan lebih lanjut.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Kajian Pustaka

Dalam sub bab penelitian ini, akan membahas penelitian-penelitian sebelumnya yang saling berkaitan dengan penelitian ini, telah banyak penelitian yang mengangkat topic terkait dengan pemesanan maupun pencarian.

Penelitian pertama dilakukan oleh Sucianti & Arisandy (2017) dengan judul Pembuatan Aplikasi Web Pencarian Jasa Pembantu Rumah Tangga (PRT) di Kota Ternate yang melatarbelakangi penelitian ini adalah Dinas Tenaga Kerja Kota Ternate mengalami kesulitan dalam mencari informasi PRT, hal itu terjadi karena terbatasnya informasi tentang PRT sehingga majikan yang ingin mencari PRT hingga harus datang ke dinas tenaga kerja. Penelitian ini menghasilkan sistem yang dapat mempercepat pencarian jasa PRT dengan data yang akurat, serta membantu PRT dalam mempromosikan jasa, sistem ini juga membantu dalam pengelolaan data transaksi, data customer dan data PRT.

Penelitian kedua dilakukan oleh Badie (2017) yang berjudul Aplikasi Pemesanan Catering Menggunakan *SMS Gateway* Berbasis Web, penelitian ini dilakukan pada Sedap catering yang mana permasalahan yang melatarbelakangi penelitian ini adalah Sedap Catering ingin menjangkau pemasaran jasa lebih luas dikarenakan sebelumnya hanya menggunakan media brosur sebagai promosi sehingga tidak efektif dan membuang biaya serta membantu dalam proses pemesanan jasa catering sehingga konsumen tidak perlu datang langsung ke tempat catering. Dalam pengembangannya sistem ini menggunakan bahasa *PHP*, *Framework Codeigniter* dan *database MySql*. Dari penelitian ini menghasilkan sistem yang dapat membantu memasarkan jasa sedap catering secara lebih luas, sistem juga membantu dalam pelayanan jasa catering kepada konsumen, sistem juga mampu mengirimkan notifikasi tentang pemesanan melalui *sms gateway*.

Penelitian ketiga dilakukan oleh Harri dkk (2014) yang berjudul tentang Sistem Pencarian Dan Pemesanan Rumah Kos Menggunakan Sistem Informasi *Geografi* (SIG). Yang melatarbelakangi penelitian ini adalah untuk membantu para pendatang di kota Yogyakarta membantu mencari sekaigus memesan rumah kos sesuai kebutuhannya dan mengetahui wilayah tiap kos tersebut serta membantu para pemilik rumah kos untuk memasarkannya secara online. Hasil yang didapat dari penelitian ini adalah sistem dapat membantu para pendatang untuk mencari kos berdasarkan wilayahnya karena memanfaatkan *google maps* serta para pendatang juga dapat melihat informasi siapa saja yang menyewakan kos.

Perbandingan ketiga penelitian tersebut dapat dilihat pada Tabel 2.1 Perbandingan Objek dan Metode Pengembangan Sistem, dibawah berikut ini:

Table 2.1 Perbandingan Objek dan Metode Pengembangan Sistem

No	Judul	Metode	Penelitian yang Dilakukan	Hasil Penelitian
1	Pembuatan Aplikasi Web Pencarian Jasa Pembantu Rumah Tangga(PRT) Di Kota Ternate(Sucianti & Arisandy,2017).	Metode yang digunakan adalah metode <i>waterfall</i> menggunakan pendekatan berorientasi objek	Penelitian yang dilakukan pada Dinas Tenaga Kerja Kota Ternate dikarenakan mengalami kesulitan dalam mencari informasi PRT,hal itu terjadi karena terbatasnya informasi tentang PRT sehingga majikan yang ingin mencari PRT hingga harus datang ke dinas tenaga kerja,penelitian ini dilakukan agar mempermudah majikan dan dinas tenaga kerja dalam mencari PRT	Pada hasil penelitian yang dilakukan sistem dapat mempercepat pencarian jasa PRT dengan data yang akurat,serta membantu PRT dalam mempromosikan jasa,sistem ini juga membantu dalam pengelolaan data transaksi,data customer dan data PRT
2	Aplikasi Pemesanan Catering Menggunakan SMS Gateway Berbasis Web(Badie,2017)	Metode yang digunakan menggunakan pendekatan berorientasi objek	Penelitian ini dilakukan pada Sedap Catering untuk menjangkau pemasaran jasa lebih luas dikarenakan sebelumnya hanya menggunakan media brosur sebagai	Pada hasil penelitian yang dilakukan ini sistem dapat membantu memasarkan jasa sedap catering secara lebih luas,sistem juga membantu dalam pelayanan jasa catering kepada konsumen,sistem juga mampu mengirimkan notifikasi tentang pemesanan melalui <i>sms gateway</i>

			promosi sehingga tidak efektif dan membuang biaya serta membantu dalam proses pemesanan jasa catering sehingga konsumen tidak perlu datang langsung ke tempat catering	
3	Sistem Pencarian Dan Pemesanan Rumah Kos Menggunakan Sistem Informasi <i>Geografi</i> (SIG)(Harri dkk,2014).	Metode penelitian ini yaitu dengan survey secara langsung ke lapangan (observasi) dan wawancara guna memperoleh data dan dalam pengembangan sistemnya menggunakan metode <i>waterfall</i>	Penelitian ini dilakukan untuk membantu para pendatang dikota Yogyakarta membantu mencari sekaigus memesan rumah kos sesuai kebutuhannya dan mengetahui wilayah tiap kos tersebut serta membantu para pemilik rumah kos untuk memasarkannya secara online	Pada hasil penelitian yang dilakukan sistem dapat membantu para pendatang untuk mencari kos berdasarkan wilayahnya karena memanfaatkan <i>google maps</i> serta para pendatang juga dapat melihat informasi siapa saja yang menyewakan kos

Mengacu pada penelitian sebelumnya tentang sebuah sistem yang dapat digunakan untuk mencari dan memesan suatu jasa, maka penelitian ini ingin mengembangkan suatu sistem yang belum ada di Kota Malang yaitu Aplikasi Pemesanan Jasa Instruktur Olahraga Kota Malang yang sekiranya akan membantu menyelesaikan permasalahan warga Kota Malang terkait sulitnya mencari instruktur olahraga Kota Malang yang berpengalaman dan *professional*. Dalam pengembangan aplikasi ini akan digunakan Metode *Waterfall*, dikarenakan metode ini menjaga kualitas siklus hidup software terjaga sehingga pengembangan software dapat terstruktur. Pada pengembangan aplikasi ini juga akan digunakan bahasa *PHP*, *Bootstrap* dan *Framework Codeigniter* serta akan terdapat *google maps api* sebagai fungsi pemilihan tempat latihan, *email gateway* sebagai fungsi notifikasi pemesanan jadwal. Dan dalam pengembangan aplikasi ini akan menggunakan konsep pendekatan berorientasi obyek dengan pemodelan menggunakan diagram UML.

## 2.2 Definisi Instruktur Olahraga

Pada sub bab ini akan menjelaskan mengenai pengertian instruktur olahraga beserta manfaat instruktur olahraga.

### 2.2.1 Pengertian Instruktur Olahraga

Menurut Febriana(2011), instruktur olahraga merupakan orang yang membantu pelanggan dalam menjalankan olahraga secara baik dan benar, membuat program untuk setiap pelanggan, memastikan pelanggan melakukan latihan yang benar dan sesuai dengan ketentuan latihan, serta menambah pengetahuan pelanggan dalam masalah kebugaran yang mencakup masalah nutrisi dan panduan kesehatan secara umum.

### 2.2.2 Manfaat Instruktur Olahraga

Menurut kompas.com berikut beberapa macam manfaat menggunakan jasa instruktur olahraga, antara lain :

#### 1. Dapat membimbing melakukan teknik gerakan yang benar

Instruktur olahraga sangat membantu bagi mereka para pemula. Dengan mengetahui teknik yang tepat dalam gerakan maka akan membantu mendapat hasil yang maksimal dan terhindar dari cedera.

#### 2. Dapat mengetahui rencana program yang tepat

Pada hal ini dibutuhkan instruktur olahraga yang berpengalaman dan bersertifikasi dalam membimbing kita menjalani program dikarenakan jika program yang dilakukan tidak tepat maka kita juga akan kurang mendapat hasil yang maksimal atau bahkan berpotensi cedera.

### 3. Dapat mengetahui saran gizi yang tepat

Instruktur olahraga yang sudah berpengalaman dan bersertifikasi pasti dapat memberikan saran nutrisi yang sesuai dengan program kita, mereka akan merekomendasikan makanan apa yang akan menunjang hasil latihan kita. Karena pola makan yang tepat akan dapat memaksimalkan hasil latihan yang kita lakukan.

### 4. Dapat memotivasi kita mencapai goal yang diinginkan

Tidak jarang orang yang bosan dengan beberapa latihan olahraga, disini peran instruktur olahraga sangat penting dimana mereka akan membantu seseorang menetapkan tujuan yang spesifik dari latihan yang sudah dijalani. Mereka juga akan membantu menelaraskan gaya hidup kita agar hasil yang kita dapatkan maksimal.

## 2.3 Pengertian Aplikasi Pencarian Jasa

Aplikasi adalah sebuah sistem dua arah yang digunakan untuk mengolah data menjadi suatu informasi yang akan berguna untuk pihak bersangkutan (Sri Widianti, 2000). Aplikasi pencarian jasa sendiri merupakan suatu sistem yang memberikan informasi terkait jasa yang ingin digunakan, sistem ini dapat diakses sewaktu-waktu jika pengguna ingin menggunakan layanannya. Manfaat dari aplikasi pencarian jasa adalah membantu para pengguna jasa mendapatkan informasi dan melakukan transaksi terkait jasa apa yang ingin mereka gunakan.

## 2.4 Pengertian Website

Website merupakan kumpulan dari halaman-halaman web yang berhubungan dengan *file-file* lain yang terkait. Dalam sebuah *website* terdapat suatu halaman yang dikenal dengan sebutan *home page*. *Home page* adalah sebuah halaman yang pertama kali dilihat ketika seseorang mengunjungi *website*. Dari *home page*, pengunjung dapat mengklik *hyperlink* untuk pindah kehalaman lain yang terdapat dalam *website* tersebut (Jhonsen, 2004).

## 2.5 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak merupakan proses pembuatan dan penggunaan prinsip dan teknik untuk mendapat kualitas perangkat lunak yang ekonomis, handal dan bekerja secara efisien (Pressman, 2010).

Sehingga rekayasa perangkat lunak memiliki hubungan dengan semua aspek dari proses pengembangan sistem mulai dari tahap spesifikasi kebutuhan sistem hingga *maintenance* sistem (Sommerville, 2001).

## 2.6 Pengembangan Perangkat Lunak

Pengembangan perangkat lunak yaitu: proses rekayasa, perancangan, implementasi, pengujian serta pemeliharaan terhadap sistem untuk mewujudkan kualitas perangkat lunak yang diinginkan.

Karena itu, pengembangan perangkat lunak bisa mencakup penelitian, pengembangan baru, modifikasi, pemakaian kembali, rekayasa ulang, pengelolaan, atau aktivitas lain yang menghasilkan produk perangkat lunak (Sommerville,2011).

## 2.7 Software Development Live Cycle

*System Development Life Cycle* (SDLC) adalah salah satu metode proses pengembangan perangkat lunak dengan menggunakan model dan metodologi dalam pengembangannya. SDLC merupakan siklus daur ulang hidup sebuah perangkat lunak yang menjelaskan setiap proses yang dilalui dari tiap tahapannya (Sommerville,2011). Dalam SDLC terdapat beberapa model atau metode yang digunakan dalam pengembangannya, yaitu: *Waterfall*, *Prototype*, *Rapid Application Development(RAD)*, *Spirall* dan *Agile*.

Dimana langkah-langkah dalam proses pengembangannya adalah :

### 1.Rekayasa Kebutuhan

Merupakan sebuah tahapan menggali informasi yang akan dibutuhkan pada sistem. Proses ini akan mengetahui masalah yang terjadi dan akan menghasilkan kebutuhan-kebutuhan pada sistem untuk membantu menyelesaikan permasalahan yang terjadi.

### 2.Perancangan

Tahapan merancang sistem berdasarkan tahapan rekayasa kebutuhan.Yang mana dalam proses ini perancangan yang dibuat pada sistem harus sesuai dengan hasil rekayasa kebutuhan.

### 3.Implementasi

Tahap dimana hasil rancangan akan diimplementasikan dalam bentuk coding.

### 4.Pengujian

Sebuah tahap yang mana sistem akan dilakukan uji coba, dengan tujuan apakah sudah sesuai dengan fungsi awal yang sudah dirancang.

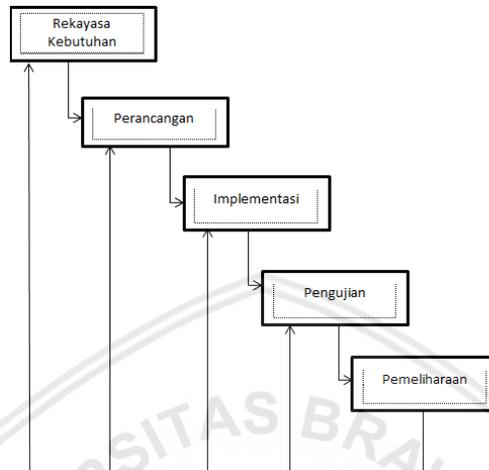
### 5.Pemeliharaan

Suatu tahapan untuk melakukan perbaikan pada sistem yang terjadi kesalahan atau kegagalan dalam menjalankan fungsinya.

## 2.8 Metode Pengembangan *Waterfall*

Model *Waterfall* merupakan salah satu model pengembangan perangkat lunak yang ada di dalam model SDLC (*Sequential Development Life Cycle*). Di jelaskan bahwa model *waterfall* sering juga disebut model *sekuensi linear* atau alur hidup klasik.

Pengembangan sistem dikerjakan secara terurut mulai dari analisis, desain, pengkodean, pengujian dan tahap pemeliharaan (Yoki & Udi,2018). Pada Gambar 2.1 *Waterfall Model* merupakan gambar pengembangan metode *waterfall*. Dan pada Tabel 2.2 Pengertian Tahapan dalam Metode *Waterfall* menjelaskan tentang pengertian masing-masing tahapannya.



**Gambar 2.2 *Waterfall Model* (Sommerville,2011)**

**Table 2.2 Pengertian Tahapan dalam Metode *Waterfall***

Tahap	Pengertian
Rekayasa Kebutuhan	Merupakan tahap elisitasi kebutuhan sistem berdasarkan keinginan pengguna
Perancangan	Merupakan tahapan perancangan kebutuhan sistem berdasarkan pemodelan yang sudah dilakukan
Implementasi	Merupakan tahapan dimana pengembang mengimplementasikan rancangan sistem dengan kode-kode
Pengujian	Merupakan tahap dimana sistem diuji dan dilihat hasil pengujiannya, apakah sudah sesuai dengan kebutuhan pengguna
Pemeliharaan	Merupakan tahap pemulihan sistem jika terjadi kesalahan dalam proses pengembangannya

### 2.9 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek merupakan sebuah pendekatan yang mana solusi dari masalah-masalahnya adalah berfokus pada sebuah objek. Dalam analisis dan perancangannya mengaplikasikan teknik pemodelan objek dengan menganalisis kebutuhan pada sebuah konteks.

Dalam analisis berorientasi objek(OOA) metode analisisnya dengan memeriksa persyaratan dari sudut pandang kelas dan objek sehingga dapat tercipta interaksi antar objek dalam suatu kelas tertentu. Sedangkan dalam

perancangan berorientasi objeknya digunakan notasi untuk menggambarkan pemodelan sistem tersebut. Dalam pemrograman berorientasi objeknya(OOP)digunakan bahasa pemrograman tertentu untuk mendefinisikan tipe dari masing-masing obyek ( Satzinger ,2010).

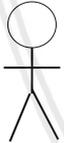
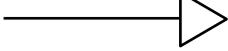
### 2.10 Pemodelan *Unified Modelling Language (UML)*

Pemodelan *Unified Modelling Language (UML)* adalah suatu standar teknik pemodelan yang digunakan untuk spesifikasi, deskripsi dan dokumentasi sebuah sistem perangkat lunak yang akan dibangun dimana berkaitan dengan suatu objek (Whitten & Bentley, 2007). UML terdiri dari beberapa tipe diagram, yaitu:

#### a. *Use Case Diagram*

*Use case diagram* adalah diagram yang menggambarkan perilaku menggambarkan actor yang berperan dalam sistem dan apa saja yang dilakukan actor kepada sistem. *Use Case* adalah sebuah langkah yang berurutan dan saling berkaitan. Actor suatu obyek yang akan berinteraksi dengan sistem agar terjadi pertukaran informasi. Hubungan (*Relationship*) pada diagram *use case* adalah sebuah garis yang digambarkan antara 2 simbol. Arti dari garis perhubungan yang terjadi berbeda-beda sesuai dengan garis dan tipe symbol yang digambar. Relasi-relasi tersebut dapat dilihat pada Tabel 2.3 Komponen *Use Case Diagram* dibawah ini:

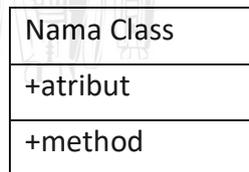
**Table 2.3 Komponen *Use Case Diagram***

Simbol	Nama	Fungsi
	Actor	Obyek yang berkomunikasi dengan sistem
	<i>Use Case</i>	Gambaran fungsional sistem
	<i>Association</i>	Penghubung antara aktor dengan use case
	<i>Generalization</i>	Spesialisasi dari elemen yang lain

	<i>Dependency</i>	Relasi antar elemen yang saling bergantung
	<i>Aggregation</i>	Hubungan antar elemen dimana elemen berisi elemen lainnya
<< include >>	<i>Include</i>	Kondisi dimana use case bergantung dengan use case yang lain
<< extends >>	<i>Extends</i>	Tambahan fungsional dari use case yang lain

b. *Class Diagram*

*Class diagram* menggambarkan relasi antar objek dengan sistem yang mana terdapat struktur objek pada sistem (Whitten & Bentley, 2007). *Class diagram* berisi nama *class*, *atribut* dan *method*. Ada beberapa hubungan antar *class diagram* yaitu: *assocaton*, *composition*, *aggregation*.



**Gambar 3.2 Contoh *Class Diagram***

*Association* adalah suatu hubungan umum yang terjadi antar 2 *class* dan digambarkan dengan sebuah garis yang menghubungkan antara 2 *class*. Garis ini dapat melambangkan beberapa tipe hubungan. Contohnya *one-to-one*, *one-to-many*, *many-to-many*.

**Gambar 2.4 *Association***

*Composition* adalah jika sebuah *class* yang tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* lain, sehingga digambarkan dengan garis yang ujungnya berbentuk jajar genjang berisi/solid yang direlasikan dengan *class* tempat dia bergantung.



**Gambar 2.5 Composition**

*Agregation* adalah sebuah yang akan berdiri sendiri walaupun merupakan bagian dari *class* lain.



**Gambar 2.6 Agregation**

c. *Sequence* Diagram

*Sequence* diagram menggambarkan interaksi antara objek berdasarkan urutan waktu dengan suatu tahapan yang menghasilkan output tertentu (Whitten & Bentley, 2007). *Sequence* diagram dapat membantu dalam identifikasi keluar masuknya pesan tingkat tinggi dari sistem. Komponen yang biasanya terdapat pada *sequence* antara lain : actor, *boundary*, *controller*, *lifeline*, *line message*.

**Table 2.4 Komponen *Sequence* Diagram**

Simbol	Nama	Fungsi
	Aktor	Obyek yang berhubungan dengan sistem
	<i>Boundary</i>	Berhubungan dengan aktor. Berisi tampilan suatu sistem.
	<i>Controller</i>	Penghubung antara <i>boundary</i> dan <i>entity</i> .
	<i>Entity</i>	Berhubungan dengan data



	<i>Lifeline</i>	Menggambarkan pesan dimulai dan diakhiri
	<i>Line Message</i>	Perintah untuk mengirim pesan

## 2.11 Bahasa Pemrograman PHP

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Maksud dari *server-side scripting* adalah *sintaks* dan perintah-perintah yang diberikan akan sepenuhnya akan dijalankan diserver tetapi disertakan pada dokumen HTML. (Bimo,2002).

## 2.12 HTML (*Hyper Text Markup Language*)

HTML merupakan sebuah bahasa markup yang digunakan sebagai tanda pada suatu dokumen teks. Simbol *mark up* ditandai dengan tanda lebih kecil(<) dan tanda lebih besar(>). Dua tanda tersebut disebut tag. Yang mana tag yang digunakan untuk tanda penutup ditambahkan garis miring didalamnya(</..>) (Yusi dkk,2012).

## 2.13 CSS (*Cascading Style Sheet*)

*Cascading Style Sheet* adalah bahasa *stye sheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. CSS bekerja sebagai pelengkap pada elemen HTML yang kesemuanya itu dapat dikendalikan dengan menggunakan dengan menggunakan sebuah bahasa *script* CSS. Penggunaan CSS dilakukan untuk memperluas kemampuan HTML dalam memformat dokumen web atau untuk memperindah tampilan web. Penulisan kode CSS disisipkan pada tak HTML. Kode CSS ditulis dengan tag <style> dan penutupnya </style> dengan mendefinisikan suatu *style* baru didalamnya sehingga dapat digunakan berulang kali (Yusi dkk,2012).

## 2.14 *Boostrap*

*Bootstrap* merupakan *tools template* untuk membuat desain tampilan *front end* sebuah *website*. *Bootstrap* terdiri dari CSS dan HTML untuk di dalam *bootstrap* terdapat *jQuery plugins* yang menghasilkan komponen *user interface* yang menarik. *Bootstrap* dapat membuat sebuah web *responsive* dengan cepat dan mudah serta dapat berjalan pada *browser-browser* populer (Rivaldi dkk,2015).

## 2.15 Javascript

*Javascript* adalah bahasa *scripting* yang populer, kelebihanannya yaitu dapat bekerja pada sebagian *browser* populer. Kode *Javascript* dapat disisipkan dalam halaman web menggunakan *tag SCRIPT*.

*Javascript* merupakan beberapa kumpulan *skrip* yang fungsinya untuk menambah interaksi antar halaman web dengan pengguna, yang mana dalam penggunaannya dilakukan dalam sisi *klien* saja sehingga dapat memberikan fitur tambahan yang lebih baik dibanding HTML (Yusi dkk,2012).

## 2.16 JQuery

*JQuery* merupakan *javascript library* yang mana merupakan kumpulan kode ataupun fungsi *javascript* yang siap pakai, sehingga dapat mempermudah dan mempercepat kita dalam membuat kode *javascript*. *JQuery* bersifat menyederhanakan kode *javascript* (Rivaldi dkk,2015)

## 2.17 Framework Codeigniter

*Framework Codeigniter* merupakan *framework* PHP bermodel MVC (*Model, View, Controller*) kelebihanannya dapat menciptakan *website* dinamis dengan bahasa PHP. *Framework Codeigniter* mempermudah dan membuat lebih cepat seorang *programmer* dalam menciptakan aplikasi berbasis web dengan simple berdasarkan pengelompokan fungsi dan tugasnya sehingga performanya akan lebih termaksimalkan dan cepat. Pemasangan *framework codeigniter* akan lebih mudah di bandingkan dengan *framework* PHP yang lain, karena hanya dengan mengekstrak file *framework* serta menempatkannya pada publikasi aplikasi atau pada server (Causa dkk,2013).

## 2.18 Database MySql

*MySQL* merupakan sebuah sistem *database* yang sifatnya *Open Source* karena dapat digunakan dalam berbagai OS. *MySql* dilengkapi *source code* yang dapat dijalankan langsung dari OS dengan mengunduhnya diinternet secara gratis. *MySql* berlisensi *General Public Lisence* (GPL) dan dapat diintegrasikan dengan beberapa bahasa Pemrograman seperti : *Net, Java, Python, Perl*, yang merupakan bahasa pemrograman paling dominan. *MySQL* akan mendeteksi pesan kesalahan pada klien menggunakan lebih dari 20 bahasa( Causa dkk 2013).

## 2.19 Google Maps API

*Google Maps API* adalah sebuah layanan (*service*) yang diberikan oleh *Google* kepada para pengguna untuk memanfaatkan *google maps* dalam mengembangkan aplikasi. *Google Maps API* menyediakan beberapa fitur untuk memanipulasi peta, dan menambah konten melalui berbagai jenis *service* yang dimiliki, serta mengizinkan kepada pengguna untuk membangun aplikasi *enterprise* di dalam websitenya (Mahdia & Noviyanto,2013).

## 2.20 Email Gateway

*Email Gateway* merupakan layanan yang akan memproses pesan pemberitahuan secara otomatis. *Email Gateway* akan bertindak seperti penerima *email* SMTP yang mana selain dapat memproses pemberitahuan pesan secara otomatis juga dapat digunakan untuk menentukan *script* dalam memproses pesan *email* (Teddy & Oscar,2010).

## 2.21 Teori Pengujian

### 2.21.1 Black-box Testing

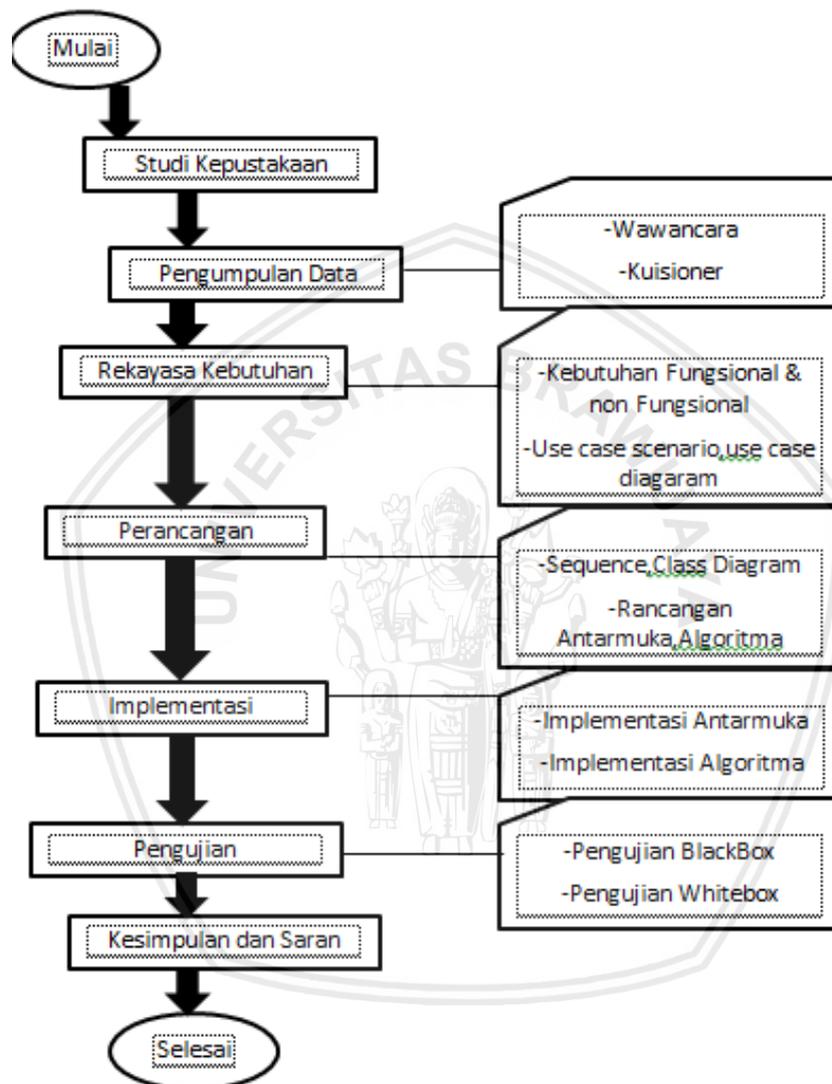
*Black-box testing* merupakan tahapan pengujian perangkat lunak yang focus pada fungsionalitas dari perangkat lunak tersebut (Pressman, 2010). *Black-box testing* digunakan untuk menemukan kesalahan pada fungsi yang salah, fungsi yang belum diimplementasikan maupun kesalahan pada *interfacenya*, dan kesalahan *databasenya*.

### 2.21.2 White-box Testing

*White-box testing* adalah pengujian untuk mengetes program berdasarkan struktur program dan komponennya (Pressman, 2010). Pada *white-box testing* pengujian didasarkan pada kasus uji yang diperoleh melalui pencarian jalur *independen source code*. Jalur *independen source code* dapat diketahui melalui perancangan *flow graph*. *Flow graph* adalah diagram yang menggambarkan alur *source code* terdiri dari *node* (simpul), biasanya direpresentasikan dalam bulatan berisi data (biasanya angka yang merepresentasikan urutan alur program dan *edge* yaitu penghubung antar *node* yang memiliki arah pada satu simpul tertentu.

## BAB 3 METODOLOGI

Langkah-langkah yang dilakukan selama proses pengembangan aplikasi pencarian jasa Instruktur Olahraga Kota Malang dapat dijelaskan pada Gambar 3.1 Diagram Alir Metodologi Penelitian dibawah berikut ini:



Gambar 3.1 Diagram Alir Metodologi Penelitian

### 3.1 Studi Kepustakaan

Landasan kepustakaan yang sudah dijelaskan pada bab II merupakan studi kepustakaan untuk digunakan dalam menunjang penulisan laporan. Studi kepustakaan juga penting dalam menghindari dari plagiasi. Studi kepustakaan yang digunakan dalam penelitian ini menggunakan beberapa buku dan jurnal diantaranya :

- 1.Kajian Pustaka
- 2.Definisi Instruktur Olahraga
  - ❖ Pengertian Instruktur Olahraga
  - ❖ Manfaat Instruktur Olahraga
- 3.Pengertian Aplikasi Pencarian Jasa
- 4.Pengertian *Website*
- 5.Rekayasa Perangkat Lunak
- 6.Pengembangan Perangkat Lunak
- 7.*Software Development Life Cycle (SDLC)*
- 8.Metode Pengembangan *Waterfall*
- 9.Pendekatan Berorientasi Obyek
- 10.Pemodelan UML(*Unified Modelling Language*)
  - ❖ *Use Case Diagram*
  - ❖ *Class Diagram*
  - ❖ *Sequence Diagram*
- 11.Bahasa Pemrograman PHP
- 12.HTML(*Hyper Text Markup Language*)
- 13.CSS(*Cascading Style Sheet*)
- 14.*Bootstrap*
- 15.*Javascript*
- 16.*Jquery*
- 17.*Framework Codeigniter*
- 18.*Database MySql*
- 19.*Google Maps Api*
- 20.*Email Gateway*
- 21.Teori Pengujian Perangkat Lunak
  - ❖ *Black Box Testing*
  - ❖ *White Box Testing*

### 3.2 Pengumpulan Data

Pengumpulan data digunakan untuk mendapat data yang akan digunakan pada penelitian. Pengumpulan data dilakukan dengan proses wawancara dan kuisioner, yang mana pada proses wawancara dilakukan pada instruktur olahraga Kota Malang yang bertujuan memperoleh data profil instruktur,kendala yang

dihadapai dalam pemasaran jasanya, proses bisnis yang dilakukan biasanya dilakukan dalam pemasaran jasanya, wawancara ini dilakukan dengan tujuan untuk proses identifikasi kebutuhan yang diinginkan instruktur olahraga pada tahapan rekayasa kebutuhan sistem nanti, sedangkan penyebaran kuisisioner online dilakukan pada calon pengguna sistem yang bertujuan untuk proses identifikasi rekayasa kebutuhan yang diinginkan pengguna.

### 3.3 Rekayasa Kebutuhan

Rekayasa kebutuhan merupakan tahapan dalam mengidentifikasi seluruh kebutuhan fungsional dan non fungsional yang akan diperlukan dalam pengembangan perangkat lunak. Yang mana rekayasa kebutuhan juga dijadikan dalam proses menentukan lingkungan target dari perangkat lunak yang dibangun. Langkah awal dalam tahap ini adalah mengidentifikasi data yang sudah terkumpul lalu mulai mengidentifikasi actor yang berperan didalamnya serta kebutuhan tiap aktornya, lalu memodelkannya dalam bentuk use case diagram guna mengetahui gambaran secara keseluruhan sistem, kemudian membuat use case diagram guna mengetahui deskripsi masing-masing fungsionalitas sistem beserta actor yang berperan. Pada tahap rekayasa kebutuhan ini kebutuhan yang sudah dibuat akan terlebih dulu diverifikasi dan divalidasi terhadap actor-aktor yang terkait sehingga pada tahap perancangan seluruh kebutuhan yang berkaitan dengan masing-masing aktor sudah tidak mengalami perubahan lagi.

### 3.4 Perancangan

Perancangan sistem merupakan tahapan dimana suatu perangkat lunak dirancang berdasarkan hasil yang didapat dari tahapan rekayasa kebutuhan. Perancangan didasarkan pada teori-teori yang sudah dibahas pada bab sebelumnya dan rekayasa kebutuhan sebagai sumber acuan dari sistem agar sebuah sistem tidak melenceng dari apa yang diinginkan dan sistem dapat sesuai dengan kebutuhan yang sudah didefinisikan. Perancangan sistem pada penelitian ini menggunakan pendekatan berorientasi objek dan pemodelan UML sehingga setelah melalui tahap rekayasa kebutuhan maka perancangan akan dimodelkan dengan *sequence* diagram guna menunjukkan hubungan interaksi antar objek dengan *class* tertentu dan *class* diagram guna mengetahui hubungan antar *class*. Selain itu pada penelitian ini juga akan melakukan perancangan *interface* guna mendefinisikan letak komponen sistem dan perancangan *algoritma* guna mengetahui *method* yang akan digunakan dalam sistem.

### 3.5 Implementasi

Implementasi merupakan tahapan untuk merealisasikan rancangan perangkat lunak yang akan dibangun kedalam *coding*. *Implementasi* terdiri dari dua tahapan yaitu: *implementasi algoritma* dan *implementasi interface* dari perangkat lunak. *Implementasi* didasarkan pada perancangan sistem sebelumnya dan harus disesuaikan dengan kebutuhan yang sudah didefinisikan.

### 3.6 Pengujian

Pengujian Aplikasi merupakan tahapan dilakukannya testing pada perangkat lunak yang bertujuan menunjukkan bahwa perangkat lunak sudah memenuhi kebutuhan yang didefinisikan serta mampu berjalan sesuai spesifikasi yang melandasi. Pengujian dilakukan menggunakan 2 metode yaitu: *white box testing* dan *black box testing*.

Secara garis besar, pengujian *white box testing* adalah pengujian yang dilakukan berdasarkan pada pengecekan detail perancangan, dengan menggunakan struktur control dari desain program secara procedural dan membaginya kedalam beberapa kasus uji.

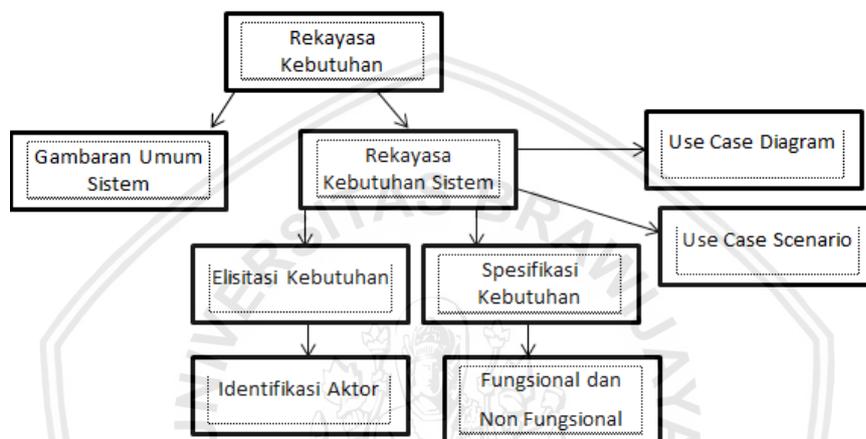
Sedangkan pengujian *black box testing* adalah pengujian yang dilakukan dalam mencari beberapa kesalahan dalam fungsi-fungsi yang salah, fungsi yang belum diimplementasikan, kesalahan pada *interface*, kesalahan *database*.

### 3.7 Kesimpulan dan Saran

Kesimpulan merupakan tahapan terakhir yang dilakukan setelah seluruh tahap sebelumnya selesai dilakukan. Kesimpulan diambil dari hasil analisis keseluruhan kinerja perangkat lunak yang sudah dibangun dan digunakan sebagai sarana kritik dan saran untuk membantu pengembangan perangkat lunak yang mungkin akan dikembangkan pada masa mendatang.

## BAB 4 REKAYASA KEBUTUHAN

Bab rekayasa kebutuhan menjabarkan tentang kebutuhan-kebutuhan dalam pengembangan aplikasi pemesanan jasa instruktur olahraga Kota Malang. Rekayasa kebutuhan merupakan suatu tahapan dalam suatu model SDLC yang harus dilakukan terlebih dahulu sebelum masuk tahapan perancangan. Pada tahap rekayasa kebutuhan di penelitian ini dimulai dari elisitasi kebutuhan, deskripsi sistem, identifikasi actor, spesifikasi kebutuhan, *use case* diagram dan *use case scenario*. Struktur diagram rekayasa kebutuhan dapat dilihat pada Gambar 4.1 Struktur Diagram Rekayasa Kebutuhan dibawah ini.



Gambar 4.1 Struktur Diagram Rekayasa Kebutuhan

### 4.1 Deskripsi Umum Sistem

Aplikasi pemesanan jasa instruktur olahraga Kota Malang merupakan sebuah perangkat lunak yang digunakan dalam mencari dan memesan jasa instruktur olahraga Kota Malang. Aplikasi pemesanan ini dapat diakses pada berbagai browser. Untuk mempermudah dalam melakukan validasi pemesanan terdapat admin yang akan mengontrol aplikasi tersebut. Layanan yang disediakan oleh aplikasi ini antara lain pemesanan jasa instruktur olahraga, menjadwalkan latihan, pendaftaran member dan instruktur olahraga, dll.

Pengguna dari aplikasi ini adalah instruktur olahraga, member dan juga admin aplikasi. Setiap pengguna wajib memiliki username dan password untuk masuk didalam sistem dan menggunakan layanannya, namun untuk calon instruktur olahraga sebelum dapat menggunakan layanan jasa untuk mempromosikan dirinya harus melalui proses pendaftaran dengan mengisi beberapa persyaratan yang akan digunakan sebagai profil instruktur serta digunakan untuk bahan validasi admin, dinyatakan layak atau tidaknya untuk mendaftar sebagai instruktur olahraga, untuk calon member juga harus mendaftar terlebih dahulu sebelum melakukan pemesanan instruktur olahraga.

Pada aplikasi ini instruktur olahraga dapat membuat kelas yang terdiri dari event, regular dan paket. Pada pembuatan kelas event, instruktur olahraga menginputkan antara lain jumlah sesi, jumlah murid, durasi, harga, jadwal, tempat dan agendanya namun pada kelas regular dan paket untuk tempat dan jadwal latihan akan dipilih member sendiri. Jumlah sesi disini memiliki artian pertemuan dalam melakukan latihan olahraga tersebut sehingga pada kelas paket terdapat beberapa sesi antara lain 4, 8, 12 dikarenakan pada kelas paket, pertemuan dilakukan rutin beberapa kali pada satu bulan, sehingga instruktur olahraga dapat memilih berapa sesi yang diinginkan dalam pembuatan kelas paket tersebut. Namun pada kelas regular dan event sesi hanya dilakukan 1 kali. Jumlah murid disini memiliki artian berapa murid yang harus dibawa saat proses latihan. Dalam pembuatan kelas paket dan regular, instruktur olahraga dapat memilih rentang jam.

Perbedaan pemesanan jenis kelas dalam aplikasi ini adalah jika kelas event maka jadwal kelas sudah diatur oleh instruktur olahraga sehingga member tidak perlu memilih jadwal lagi, namun jika kelas regular maka member diperkenankan memilih tanggal, jam dan tempat latihan, sedangkan jika kelas paket maka member akan memilih hari dimana untuk 4 sesi cukup memilih hari ke 1, untuk 8 sesi cukup akan memilih hari 1 dan hari 2 dan jika 12 sesi maka akan memilih hari 1, hari 2 dan hari 3.

Pembayaran yang dilakukan pada aplikasi ini adalah dengan cara manual validasi dari admin yang mana sebelumnya member harus mengupload bukti pembayarannya kemudian jika sudah sesuai akan dikonfirmasi oleh admin. Setelah proses konfirmasi selesai maka akan terdapat notifikasi ke email instruktur dan member tentang jadwal latihan. Dan jika jadwal latihan sudah selesai maka member dapat memberi rating kepada instruktur.

## **4.2 Proses Rekayasa Kebutuhan**

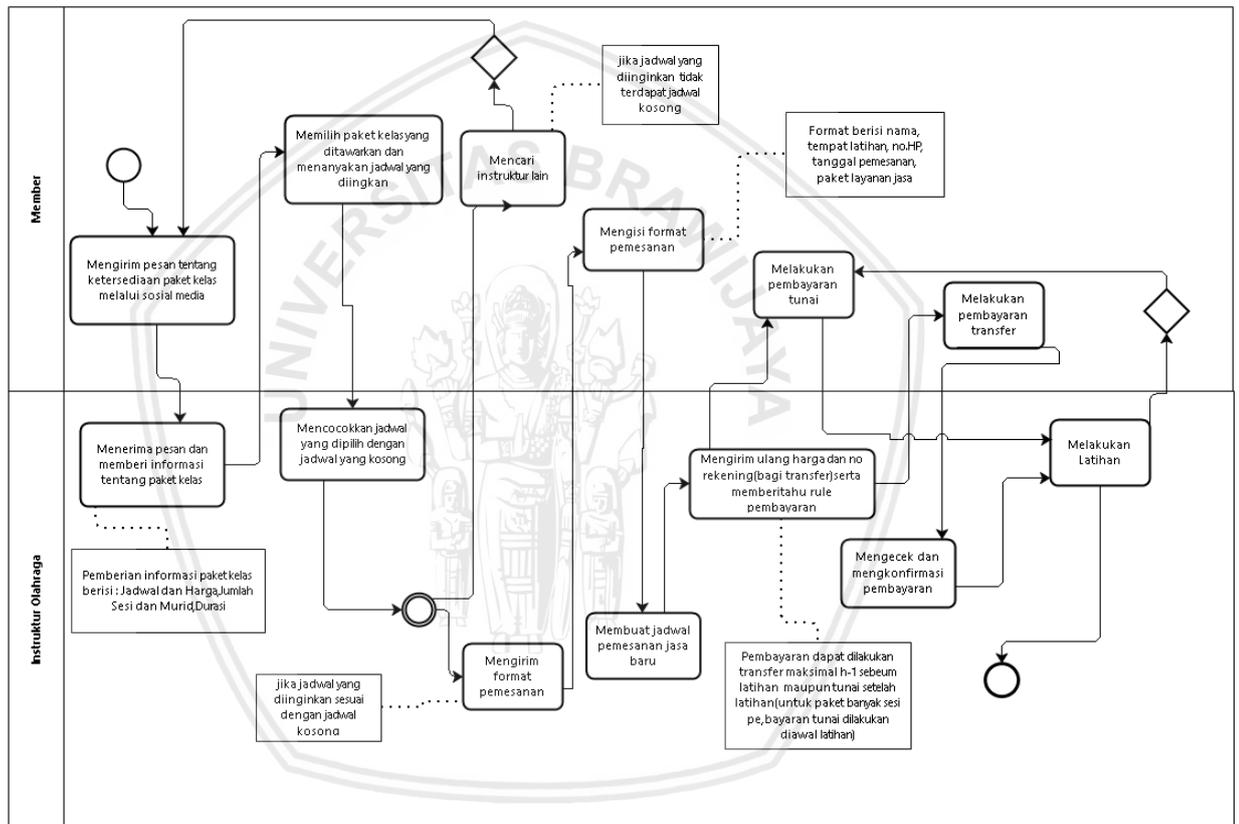
Pada tahap rekayasa kebutuhan di penelitian ini dimulai dari elisitasi kebutuhan yang terdiri dari: identifikasi actor. Spesifikasi kebutuhan yang terdiri dari: daftar kebutuhan fungsional dan non fungsional, pemodelan *use case* diagram dan *use case scenario*.

### **4.2.1 Elisitasi Kebutuhan**

Proses elisitasi kebutuhan merupakan tahapan untuk mengetahui suatu permasalahan yang terjadi pada penelitian ini. Untuk mengetahui permasalahannya secara mendalam serta untuk proses pengumpulan data kebutuhan, maka dilakukan dua buah metode yaitu: metode wawancara dan menggunakan kuisisioner. Wawancara dilakukan kepada 14 instruktur olahraga Kota Malang dan kuisisioner disebar kepada calon pengguna jasa yang terdiri dari 3 pekerjaan yaitu: pelajar, mahasiswa dan umum. Dari hasil wawancara yang terdapat dalam lampiran A, dapat diketahui bahwa terjadi kendala dalam proses pemesanan dan pemasaran diantaranya yaitu, media pemasarannya masih mulut

ke mulut ataupun sebatas media social sehingga dirasa kurang efektif karena pengguna jasa akan menghubungi instruktur olahraga tersebut untuk mengetahui tentang informasi jadwal, harga paket serta ketersediaan waktu instruktur untuk mengajar, sebelum pengguna melakukan pemesanan.

Dari hasil kuisisioner yang disebar terhadap 70 responden yang terdapat pada lampiran B terdiri dari 31% laki-laki dan 69% perempuan, rentang umur 15-60 tahun dengan kriteria pekerjaan 67,6% mahasiswa, 15,5% pelajar dan 16,9% umum, menyatakan 63,4% dari mereka menemui permasalahan dalam mencari instruktur, masalah yang sering terjadi adalah sulitnya mencari profil instruktur, kurangnya referensi, sulit mencari instruktur yang dapat menyesuaikan dengan jadwal kita dan kurangnya informasi paket harga instruktur per sesi latihan.



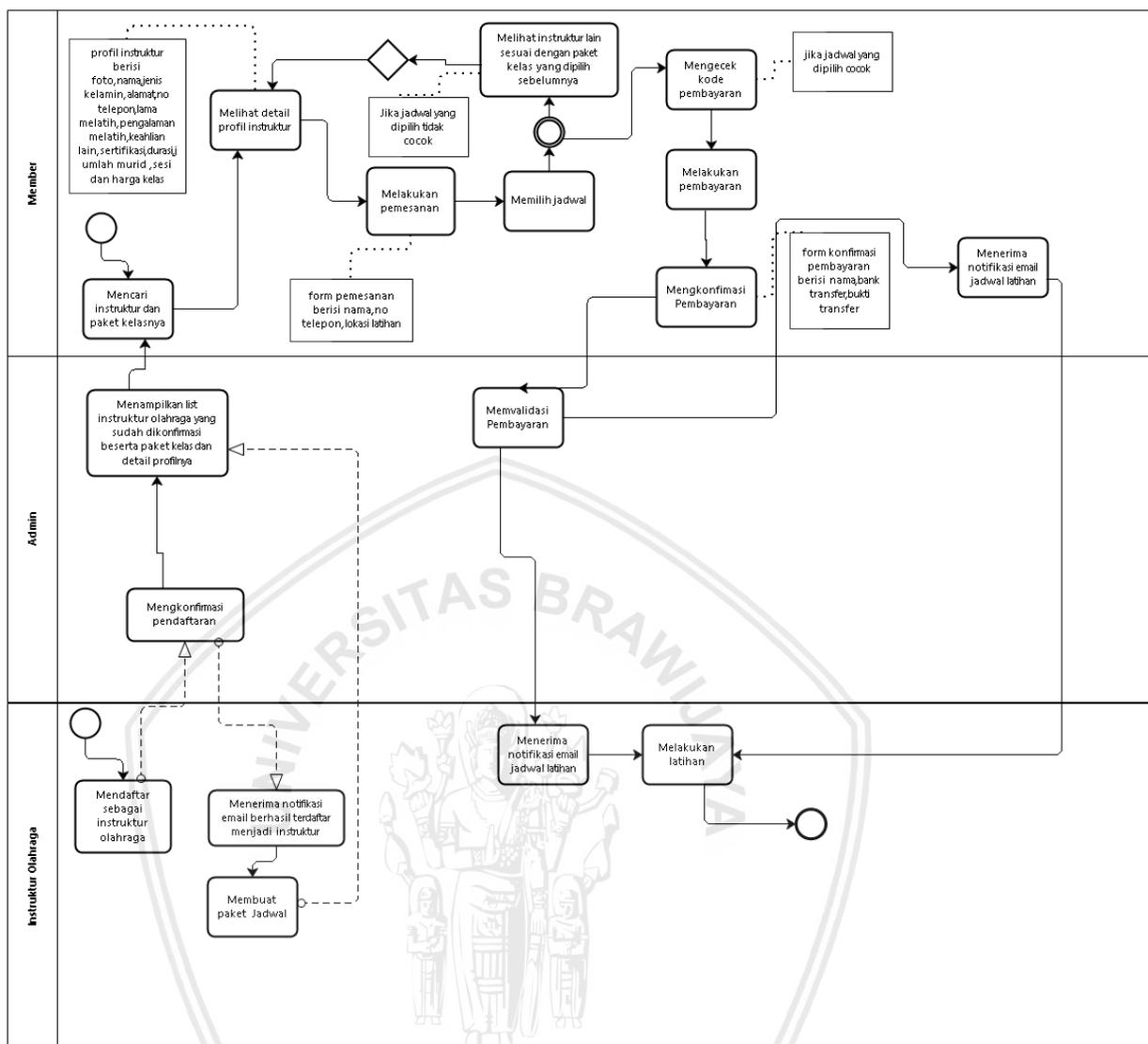
**Gambar 4.2 Standar Operasional Procedure(SOP) As Is Pemesanan Jasa Instruktur Olahraga Kota Malang**

Dari gambar 4.2, dapat dijelaskan bahwa untuk melakukan pemesanan jasa instruktur olahraga kota Malang diperlukan adanya mencari beberapa informasi lewat media social maupun mulut ke mulut, yang kemudian akan menghubungi nomer yang bisa dihubungi untuk melakukan transaksi pemesanan. Sebelum melakukan pemesanan hal yang pertama kali dilakukan member adalah menanyakan ketersediaan jenis paket kelas yang ada, lalu instruktur akan memberitahukan mengenai paket kelas yang ditawarkan.

Kemudian member melakukan pemilihan paket kelas dan juga jadwal yang diinginkan, lalu menanyakan jadwal kepada instruktur apakah jadwal tersebut bisa sesuai dengan jadwal instruktur. Instruktur olahraga kemudian mencocokkan dengan jadwal yang masih kosong, jika jadwal yang dipilih member sesuai dengan jadwal instruktur yang masih kosong maka kesepakatan pemesanan akan dibuat namun jika jadwal yang dipilih member tidak ada yang sesuai dengan jadwal kosong instruktur maka member akan mencoba mencari informasi tentang instruktur lain dan mengulang proses pemesanan dari awal.

Jika terdapat kesepakatan jadwal antara instruktur dan member maka proses pemesanan akan berlanjut yang mana instruktur akan mengirimkan format pemesanan kemudian member akan mengisi format pemesanan yang berisi nama, no telepon, tempat latihan, tanggal latihan dan paket kelas yang dipilih, yang kemudian member akan mengirimkannya kepada instruktur agar instruktur membuat note jadwal baru latihan dengan member.

Setelah semuanya sepakat maka instruktur akan mengirim ulang harga yang harus dibayar beserta no rekening yang ingin ditransfer serta memberikan rule metode pembayaran yaitu, jika dilakukan transaksi pembayaran transfer maka minimal h-1 sebelum latihan atau jika pembayaran bayar tunai dapat dilakukan ditempat latihan, untuk latihan dengan banyak sesi maka pembayaran secara tunai akan dilakukan pada sesi latihan pertama. Pada proses pembayaran transfer jika instruktur sudah mengecek dan mengkonfirmasi transfer berhasil maka latihan dapat dilaksanakan.



**Gambar 4.3 Standar Operational Procedure(SOP) To Be Pemesanan Jasa Instruktur Olahraga Kota Malang**

Dari gambar 4.3,dapat dijelaskan bahwa untuk melakukan pemesanan jasa instruktur olahraga kota Malang dengan mencari instruktur dan paket kelas yang diinginkan, kemudian admin selaku pengelola sistem akan menampilkan daftar list instruktur olahraga yang sebelumnya sudah mendaftar dan berhasil diverifikasi pendaftarannya oleh admin sehingga dalam hal ini untuk instruktur yang akan dipesan benar-benar sudah professional. Instruktur yang berhasil diverifikasi pendaftarannya akan mendapatkan notifikasi melalui email, kemudian instruktur dapat membuat paket jadwal mereka yang kemudian akan ditampilkan sistem sehingga member dapat menemukan list instruktur dan paket kelasnya. Setelah mengetahui dan memilih instruktur beserta paket kelasnya maka member akan melihat detail profil instruktur untuk melakukan pemesanan, jika member cocok dengan profil instruktur maka member akan melakukan pemesanan dengan mengisi form kemudian memilih jadwal yang diinginkan, jika

tidak ditemukan jadwal yang diinginkan maka member akan mencari instruktur lain, jika jadwal ditemukan maka member akan mengecek kode pembayaran dan kemudian melakukan pembayaran, setelah melakukan pembayaran melalui transfer maka member akan mengkonfirmasi bukti pembayaran dengan memasukkan form berisi nama, nama bank transfer dan bukti pembayaran.

Kemudian admin akan memvalidasi pembayaran melalui sistem, sehingga setelah validasi pembayaran berhasil maka member dan instruktur akan menerima email notifikasi jadwal latihan dan latihan bisa dilakukan.

#### 4.2.1.1 Identifikasi Aktor

Aktor merupakan seseorang yang terlibat secara langsung terhadap sistem, artinya actor inilah yang selalu berinteraksi dengan sistem. Identifikasi actor dilakukan untuk mengetahui siapa saja yang akan berperan pada sistem.

**Tabel 4.1 Identifikasi Aktor**

No	Aktor	Deskripsi
1	<i>Guest</i>	<i>Guest</i> adalah actor yang dapat melihat tampilan halaman awal pada sistem (halaman login, halaman pencarian), melakukan login, melakukan pencarian, melakukan registrasi pendaftaran
2	<i>Admin</i>	<i>Admin</i> merupakan actor yang memiliki otorisasi atas sistem dan dapat mengakses sebagian besar fitur yang digunakan untuk mengelola seluruh data dalam sistem
3	Instruktur Olahraga	Instruktur olahraga merupakan actor yang memiliki username dan password untuk dapat mengakses sistem. Instruktur olahraga dapat melihat jadwal latihan, mengatur jadwal untuk kelasnya, mengedit profil, dan lain-lain
4	<i>Member</i>	<i>Member</i> merupakan actor yang memiliki username dan password untuk dapat mengakses sistem. Member dapat melakukan pemesanan instruktur, melihat jadwal latihan dan tempat latihan, melakukan pembayaran

#### 4.2.2 Spesifikasi Kebutuhan

Spesifikasi kebutuhan merupakan tahapan menemukan solusi permasalahan serta mengetahui setiap pemangku kepentingan dalam sistem. Langkah-langkah yang harus dilakukan antara lain adalah memahami alur pemesanan instruktur olahraga yang nanti akan disesuaikan dengan gambaran sistemnya.

Dalam penelitian ini terdapat dua macam kebutuhan, yaitu: kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional merupakan kebutuhan yang menggambarkan kegiatan yang dilakukan saat sistem berinteraksi dengan aktornya. Sedangkan kebutuhan non fungsional merupakan kebutuhan penunjang untuk meningkatkan kualitas sistem yang akan dikembangkan. Dalam proses validasi dan verifikasi kebutuhannya, dalam penelitian ini pengembang secara langsung bertemu dengan calon pengguna jasa dan instruktur olahraga. Tahapan ini berlangsung setelah peneliti melakukan tahap wawancara terhadap instruktur olahraga sehingga diperoleh 16 daftar kebutuhan fungsional yang dapat dilakukan instruktur olahraga tersebut. Sedangkan untuk validasi dan verifikasi kebutuhan terhadap calon pengguna jasa, peneliti melakukan penyebaran kuisisioner terlebih dahulu untuk mengetahui permasalahan apa saja yang dihadapi oleh pengguna jasa, serta mengetahui kriteria apa saja yang dibutuhkan pengguna jasa dalam melakukan pencarian dan pemesanan instruktur olahraga. Setelah itu peneliti melakukan analisis terhadap hasil kuisisioner untuk mengidentifikasi kebutuhannya, lalu setelah itu peneliti melakukan pertemuan dengan perwakilan dari 70 responden Kota Malang yang akan menjadi calon pengguna jasa.

Proses pertemuan secara langsung ini bertujuan untuk memvalidasi dan memverifikasi hasil analisis kebutuhan yang dilakukan peneliti kepada calon pengguna jasa, jika calon pengguna jasa sudah cocok dengan hasil analisis kebutuhan dari peneliti maka tidak perlu ada perubahan kebutuhan lagi. Namun dalam hal ini peneliti menemui sedikit tambahan dalam kebutuhan aplikasi yang mana dampaknya terdapat pada fitur tambahan seperti notifikasi jadwal email.

Berbeda dengan tahap validasi dan verifikasi kebutuhan pada instruktur olahraga yang dilakukan secara langsung setelah melakukan wawancara, pada calon pengguna jasa terlebih dahulu dilakukan penyebaran kuisisioner online dikarenakan calon pengguna sistem terdiri dari berbagai pekerjaan seperti mahasiswa, umum, pelajar. Tujuannya menggunakan kuisisioner online agar dalam proses penggalian permasalahan dan kebutuhan dapat efektif dan tidak secara langsung melakukan observasi ke sembarang aktor. Dari tahapan proses validasi dan verifikasi dengan perwakilan dari 70 responden Kota Malang yang akan menjadi calon pengguna jasa didapatkan 22 kebutuhan fungsional pada aplikasi yang dapat dilakukan calon pengguna sistem.

Tahapan verifikasi dan validasi hasil analisis kebutuhan ini dilakukan sebelum pengembang melakukan tahap perancangan, bertujuan agar hasil daftar kebutuhan fungsional aplikasi tidak berubah pada tahap perancangan. Pada kebutuhan non fungsional aplikasi ini didapat ketika peneliti melakukan pertemuan langsung dengan calon pengguna jasa dan instruktur olahraga yang mana dalam hal ini peneliti mengamati bahwa browser yang digunakan pada instruktur olahraga seperti opera, chrome pada iOS maupun android, sedangkan pada calon pengguna jasa ditemukan browser seperti Chrome, IE, Firefox pada personal komputer/laptop yang mereka miliki.

Aturan penomoran kebutuhan akan dijelaskan pada Gambar 4.4 Aturan Penomoran Kebutuhan dan deskripsi aturan penomorannya dijelaskan pada Tabel 4.2 Penjelasan Aturan Penomoran Kebutuhan berikut :

## APIKO-F/NF-XX-YY

**Gambar 4.4 Aturan Penomoran Kebutuhan**

**Tabel 4.2 Penjelasan Aturan Penomoran Kebutuhan**

Aturan Penomoran	Deskripsi
APIKO	Merupakan singkatan nama dari aplikasi yang akan dibangun yaitu, Aplikasi Pemesanan Instruktur Olahraga
F/NF	F menyatakan kebutuhan fungsional dan NF menyatakan kebutuhan non-fungsional
XX	Menyatakan jenis kebutuhan berdasarkan actor(hanya untuk kebutuhan fungsional)
YY	Menyatakan nomor urut kebutuhan

### 4.2.2.1 Kebutuhan Fungsional *Guest*

Daftar kebutuhan fungsional dapat dilihat pada Tabel 4.3 Kebutuhan Fungsional *Guest* dibawah ini

**Tabel 4.3 Daftar Kebutuhan Fungsional *Guest***

No	Kode Kebutuhan	Deskripsi	Nama <i>UseCase</i>
1	APIKO-F-1-001	<p>Aplikasi dapat menyimpan data <i>registrasi guest</i> sebagai <i>member</i>.</p> <ol style="list-style-type: none"> <li>1. Aplikasi menampilkan form <i>regitrasi Guest</i> sebagai <i>member</i></li> <li>2. Aplikasi menyimpan data <i>registrasi Guest</i> pada <i>database</i>.</li> <li>3. Aplikasi dapat menampilkan notifikasi berhasil mendaftar dilakukan oleh <i>guest</i> melalui <i>email</i></li> </ol>	<i>Register Member</i>
2	APIKO-F-1-002	<p>Aplikasi menyimpan data <i>registrasi guest</i> sebagai instruktur olahraga</p> <ol style="list-style-type: none"> <li>1. Aplikasi menampilkan form <i>regitrasi Guest</i> sebagai instruktur olahraga</li> </ol>	<i>Register Instruktur Olahraga</i>

		<ol style="list-style-type: none"> <li>Aplikasi menyimpan data <i>registrasi Guest</i> pada <i>database</i>.</li> <li>Aplikasi menampilkan notifikasi berhasil mendaftar dilakukan oleh <i>guest</i> melalui <i>email</i></li> </ol>	
3	APIKO-F-1-003	<p>Aplikasi dapat melakukan proses <i>autentikasi</i> bagi <i>Guest</i> yang berperan sebagai <i>admin, member</i> maupun instruktur olahraga</p> <ol style="list-style-type: none"> <li>Aplikasi dapat menampilkan <i>login Guest</i></li> <li>Aplikasi melakukan validasi username dan password <i>Guest</i></li> </ol>	<i>Login</i>
4	APIKO-F-1-004	<p>Aplikasi dapat menampilkan pencarian instruktur olahraga</p> <p>Aplikasi dapat menampilkan pencarian instruktur olahraga berdasarkan inputan jenis olahraganya dan nama instruktur olahraga</p>	Mencari instruktur olahraga
5	APIKO-F-1-005	Aplikasi dapat menampilkan pencarian paket event yang dibuat instruktur	Mencari kelas event
6	APIKO-F-1-006	<p>Aplikasi dapat menampilkan profil instruktur olahraga dan jenis kelas yang ditawarkan</p> <p>1. Aplikasi dapat menampilkan detail informasi profil instruktur olahraga dan semua jenis kelas yang ditawarkan</p>	Melihat detail profil instruktur olahraga
7	APIKO-F-1-007	<p>Aplikasi dapat menampilkan pencarian berdasarkan banyak kriteria</p> <p>Aplikasi dapat menampilkan pencarian berdasarkan banyak kriteria diantara lain : kriteria jenis olahraga, jenis kelas, urutan</p>	Mencari berdasarkan kriteria
8	APIKO-F-1-008	Aplikasi dapat menampilkan pencarian kelas reguler yang dibuat instruktur	Mencari kelas reguler
9	APIKO-F-1-009	Aplikasi dapat menampilkan pencarian kelas paket yang dibuat instruktur	Mencari kelas paket

#### 4.2.2.2 Kebutuhan Fungsional *Member*

Daftar kebutuhan fungsional dapat dilihat pada Tabel 4.4 Kebutuhan Fungsional *Member* dibawah ini

**Tabel 4.4 Daftar Kebutuhan Fungsional *Member***

No	Kode Kebutuhan	Deskripsi	Nama <i>Use Case</i>
1	APIKO-F-2-001	Aplikasi menampilkan pencarian instruktur olahraga	Mencari instruktur

		Aplikasi dapat menampilkan pencarian instruktur olahraga berdasarkan inputan jenis olahraganya	olahraga
2	APIKO-F-2-002	Aplikasi dapat menampilkan pencarian berdasarkan kriteria Aplikasi dapat menampilkan pencarian berdasarkan kriteria diantara lain : kriteria jenis olahraga,jenis kelas,urutan	Mencari berdasarkan kriteria
3	APIKO-F-2-003	Aplikasi menampilkan pencarian kelas event yang dibuat instruktur	Mencari kelas event
4	APIKO-F-2-004	Aplikasi dapat menyimpan data pemesanan 1. Aplikasi dapat menampilkan form pemesanan paket event yang default isi formnya sudah diisi oleh instruktur olahraga 2. Aplikasi menyimpan data pemesanan kelas event pada database	Melakukan pemesanan paket event
5	APIKO-F-2-005	Aplikasi menampilkan detail profil instruktur olahraga dan jenis kelas olahraga yang ditawarkan 1. Aplikasi menampilkan detail informasi profil instruktur olahraga dan semua jenis kelas olahraga yang ditawarkan	Melihat detail profil instruktur olahraga
6	APIKO-F-2-006	Aplikasi menampilkan kode pembayaran Aplikasi menampilkan form kode pembayaran	Melihat kode pembayaran
7	APIKO-F-2-007	Aplikasi menampilkan profil member 1. Aplikasi menampilkan detail informasi profil member itu sendiri	Melihat profil member
8	APIKO-F-2-008	Aplikasi melakukan penyimpanan edit data profil 1. Aplikasi menampilkan form edit profil member 2. Aplikasi melakukan edit foto profil member 3. Aplikasi menyimpan perubahan data profil pada database	Edit profil member
9	APIKO-F-2-009	Aplikasi dapat melakukan pemberian rating instruktur yang sudah melakukan kegiatan latihan	Memberi rating instruktur
10	APIKO-F-2-010	Aplikasi menyimpan data pembayaran member	Melakukan konfirmasi

		1.Aplikasi menampilkan form konfirmasi pembayaran 2.Aplikasi dapat menyimpan data file struk foto pembayaran ke database untuk validasi pembayaran	pembayaran
11	APIKO-F-2-011	Aplikasi menyediakan layanan pembatalan pemesanan	Melakukan pembatalan pemesanan
12	APIKO-F-2-012	Aplikasi menyediakan layanan menghapus akun member	Hapus akun member
13	APIKO-F-2-013	Aplikasi dapat menampilkan detail jadwal latihan 1.Aplikasi menampilkan detail jadwal latihan member 2.Aplikasi memberi notifikasi jadwal latihan melalui email.	Melihat jadwal latihan member
14	APIKO-F-2-014	Aplikasi menyediakan layanan bagi Member keluar dari akun aplikasi.	Logout
15	APIKO-F-2-015	Aplikasi menyediakan layanan mengganti foto	Edit Foto
16	APIKO-F-2-016	Aplikasi menampilkan pencarian kelas reguler yang dibuat instruktur	Mencari kelas reguler
17	APIKO-F-2-017	Aplikasi menampilkan pencarian kelas paket yang dibuat instruktur	Mencari kelas paket
18	APIKO-F-2-018	Aplikasi menampilkan detail pemesanan	Melihat detail pemesanan
19	APIKO-F-2-019	Aplikasi menampilkan bukti pembayaran	Melihat bukti pembayaran
20	APIKO-F-2-020	1.Aplikasi menampilkan form pemesanan kelas paket 2.Aplikasi menyimpan data pemesanan kelas paket pada database	Melakukan pemesanan kelas paket
21	APIKO-F-2-021	1.Aplikasi menampilkan jadwal kelas reguler 2.Aplikasi menyimpan data pemesanan kelas reguler pada database	Melakukan pemesanan kelas reguler
22	APIKO-F-2-022	Aplikasi menampilkan daftar list pembayaran	Melihat daftar pembayaran

#### 4.2.2.3 Kebutuhan Fungsional Instruktur Olahraga

Daftar kebutuhan fungsional dapat dilihat pada Tabel 4.5 Kebutuhan Fungsional Instruktur Olahraga dibawah ini

**Tabel 4.5 Daftar Kebutuhan Fungsional Instruktur Olahraga**

No	Kode Kebutuhan	Deskripsi	Nama UseCase
1	APIKO-F-3-001	Aplikasi menampilkan detail profil	Melihat profil instruktur

		instruktur olahraga itu sendiri	olahraga
2	APIKO-F-3-002	<p>Aplikasi melakukan penyimpanan edit data profil</p> <p>1. Aplikasi menampilkan form edit profil instruktur olahraga</p> <p>2. Aplikasi melakukan edit foto profil instruktur olahraga</p> <p>3. Aplikasi menyimpan perubahan data profil pada database</p>	<p>Edit profil instruktur olahraga</p>
3	APIKO-F-3-003	<p>Aplikasi dapat menyediakan layanan untuk instruktur olahraga membuat kelas reguler</p> <p>1. Aplikasi dapat menampilkan form pembuatan kelas reguler</p> <p>2. Aplikasi dapat menyimpan pembuatan kelas reguler untuk nanti akan dipilih member</p>	<p>Melakukan pembuatan kelas reguler</p>
4	APIKO-F-3-004	<p>Aplikasi dapat menyediakan layanan untuk instruktur olahraga membuat pada kelas event</p> <p>1. Aplikasi dapat menampilkan form pembuatan kelas event</p> <p>2. Aplikasi dapat menyimpan pembuatan kelas event untuk nanti akan dipilih member</p>	<p>Melakukan pembuatan kelas event</p>
5	APIKO-F-3-005	<p>Aplikasi menampilkan detail jadwal latihan instruktur</p> <p>1. Aplikasi menampilkan detail jadwal latihan instruktur.</p> <p>2. Aplikasi dapat memberi notifikasi jadwal latihan melalui email.</p>	<p>Melihat jadwal latihan instruktur</p>
6	APIKO-F-3-006	<p>Aplikasi menyediakan layanan untuk menghapus akun instruktur</p>	<p>Hapus akun instruktur</p>
7	APIKO-F-3-007	<p>Aplikasi dapat menyediakan layanan untuk instruktur olahraga membuat kelas paket</p> <p>1. Aplikasi dapat menampilkan form pembuatan kelas paket</p> <p>2. Aplikasi dapat menyimpan pembuatan kelas paket untuk nanti akan dipilih</p>	<p>Melakukan pembuatan kelas paket</p>

		member	
8	APIKO-F-3-008	Aplikasi menyediakan layanan bagi Instruktur olahraga keluar dari akun aplikasi.	Logout
9	APIKO-F-3-009	Aplikasi dapat menyediakan layanan mengganti foto	Edit Foto
10	APIKO-F-3-010	Aplikasi dapat menampilkan pencarian kelas event yang dibuat instruktur	Mencari kelas event
11	APIKO-F-3-011	Aplikasi dapat menampilkan pencarian kelas reguler yang dibuat instruktur	Mencari kelas reguler
12	APIKO-F-3-012	Aplikasi dapat menampilkan pencarian kelas paket yang dibuat instruktur	Mencari kelas paket
13	APIKO-F-3-013	Aplikasi dapat menampilkan instruktur olahraga Aplikasi dapat menampilkan pencarian instruktur olahraga berdasarkan inputan jenis olahraganya dan nama instruktur	Mencari instruktur olahraga
15	APIKO-F-3-014	Aplikasi dapat menampilkan pencarian berdasarkan kriteria Aplikasi dapat menampilkan pencarian berdasarkan kriteria diantara lain : kriteria jenis olahraga, jenis kelas, urutan	Mencari berdasarkan kriteria
16	APIKO-F-3-015	Aplikasi dapat menampilkan detail profil instruktur olahraga dan jenis kelas olahraga yang ditawarkan 1. Aplikasi menampilkan detail informasi profil instruktur olahraga dan semua jenis olahraga yang ditawarkan	Melihat detail profil instruktur olahraga

#### 4.2.2.4 Kebutuhan Fungsional Admin

Daftar kebutuhan fungsional dapat dilihat pada Tabel 4.6 Kebutuhan Fungsional Admin dibawah ini.

**Tabel 4.6 Daftar Kebutuhan Fungsional Admin**

No	Kode Kebutuhan	Deskripsi	Nama UseCase
1	APIKO-F-4-001	Aplikasi menampilkan daftar instruktur yang sudah tervalidasi pendaftarannya	Melihat daftar instruktur terverifikasi
2	APIKO-F-4-002	Aplikasi dapat menampilkan detail profil instruktur yang sudah belum tervalidasi pendaftarannya	Melihat detail pendaftar

3	APIKO-F-4-003	Aplikasi menyediakan layanan untuk mengkonfirmasi pendaftaran instruktur	Melakukan validasi pendaftaran instruktur
		1.Aplikasi memberi notifikasi pendaftaran berhasil dikonfirmasi melalui email instruktur	
4	APIKO-F-4-004	Aplikasi menyediakan layanan untuk menolak pendaftaran instruktur	Menolak pendaftaran instruktur
		1.Aplikasi memberi notifikasi pendaftaran berhasil dikonfirmasi melalui email instruktur	
5	APIKO-F-4-005	Aplikasi dapat menampilkan list daftar pembayaran member	Melihat daftar pembayaran member
6	APIKO-F-4-006	Aplikasi dapat menyediakan layanan untuk memvalidasi pembayaran pemesanan instruktur olahraga	Melakukan validasi pembayaran
7	APIKO-F-4-007	Aplikasi dapat menyediakan layanan untuk membatalkan pembayaran pemesanan instruktur olahraga	Membatalkan pembayaran
8	APIKO-F-4-008	Aplikasi dapat menampilkan list daftar member	Melihat daftar member
9	APIKO-F-4-009	Aplikasi dapat menampilkan list daftar paket event	Melihat daftar kelas event
10	APIKO-F-4-010	Aplikasi dapat menyediakan layanan untuk memvalidasi paket event yang diajukan instruktur olahraga	Melakukan validasi kelas event
11	APIKO-F-4-011	Aplikasi dapat menyediakan layanan menolak paket event yang diajukan instruktur olahraga	Menolak kelas event
12	APIKO-F-4-12	Aplikasi menyediakan layanan bagi Admin keluar dari akun aplikasi.	Logout
13	APIKO-F-4-13	Aplikasi dapat menampilkan daftar instruktur yang sudah belum tervalidasi pendaftarannya	Melihat daftar instruktur yang belum terverifikasi
14	APIKO-F-4-14	Aplikasi dapat membuat member baru	Membuat Member
15	APIKO-F-4-15	Aplikasi dapat membuat instruktur baru	Membuat Instruktur
16	APIKO-F-4-16	Aplikasi dapat menghapus member	Hapus Member
17	APIKO-F-4-17	Aplikasi dapat menghapus instruktur	Hapus Instruktur

18	APIKO-F-4-18	Aplikasi dapat menampilkan dashboard	Melihat Dashboard
19	APIKO-F-4-19	Aplikasi dapat melakukan edit profil member	Edit Member
20	APIKO-F-4-20	Aplikasi dapat melakukan edit profil member	Edit Instruktur
21	APIKO-F-4-21	Aplikasi dapat menampilkan detail pemesanan member	Melihat detail pemesanan member
22	APIKO-F-4-22	Aplikasi menampilkan detail pembayaran	Melihat detail pembayaran
23	APIKO-F-4-23	Aplikasi menampilkan detail pemesanan	Melihat detail pemesanan

#### 4.2.2.5 Kebutuhan Non Fungsional

Daftar kebutuhan non fungsional dapat dilihat pada Tabel 4.7 Kebutuhan Non Fungsional dibawah ini

**Tabel 4.7 Daftar Kebutuhan Non Fungsional**

No	Kode Kebutuhan	Deskripsi	Nama Kebutuhan
1	APIKO-NF-001	Aplikasi berjalan pada beberapa browser seperti <i>mozilla, chrome, internet explore, dll</i>	<i>Compatibility</i>

#### 4.2.3 Pemodelan Diagram *Use case*

Diagram *use case* digunakan untuk menggambarkan komunikasi aktor dengan sistem. Pemodelan didasarkan pada daftar fungsionalitas. Pemodelan diagram *use case* dapat dilihat pada Gambar 4.5 *Use Case Diagram* berikut.

Gambar 4.5 *Use Case Diagram* dibawah ini merupakan diagram *use case* dari aplikasi pemesanan instruktur olahraga kota malang berbasis web. Pada aplikasi ini memiliki 4 aktor yaitu: *Guest, Member, Instruktur Olahraga dan Admin*. *Guest* dapat melakukan *Register member*, *Register instruktur*, *Login*, Mencari instruktur olahraga, Mencari berdasarkan kriteria, Mencari kelas paket, Mencari kelas regular, Mencari kelas event, melihat detail profil instruktur olahraga. *Member* dapat melakukan Mencari instruktur olahraga, Mencari berdasarkan kriteria, Mencari kelas paket, Mencari kelas regular, Mencari kelas event, melihat detail profil instruktur olahraga, Edit foto, Melakukan pemesanan event, Melakukan pemesanan regular, Melakukan pemesanan paket, Hapus akun member, Melihat jadwal latihan member, Melakukan pembatalan pemesanan, Memberi rating instruktur, ada juga fungsi Melihat kode pembayaran, Melakukan konfirmasi pembayaran, Melihat bukti pembayaran, Melihat detail pemesanan yang ketiga fungsi tersebut mengextends dari fungsi Melihat daftar pembayaran, terdapat pula fungsi Edit profil member yang mengextends dari fungsi melihat profil member dan logout. *Instruktur* dapat melakukan Mencari instruktur olahraga, Mencari berdasarkan kriteria, Mencari kelas paket, Mencari kelas regular, Mencari kelas event, melihat detail profil instruktur olahraga, Hapus akun instruktur, Edit foto, Melakukan pembuatan kelas event, Melakukan pembuatan kelas regular, Melakukan pembuatan kelas paket, Melihat jadwal latihan instruktur, terdapat fungsi Edit profil instruktur olahraga yang mengextends dari fungsi Melihat profil instruktur olahraga dan logout. *Admin* dapat melakukan Melihat detail pendaftar, Menolak pendaftar instruktur, Melakukan validasi pendaftar instruktur yang mengextends fungsi melihat daftar instruktur belum terverifikasi, terdapat fungsi Hapus instruktur, Edit instruktur, Menambah instruktur yang mengextends dari fungsi Melihat daftar instruktur terverifikasi, fungsi Edit member, Hapus akun member, Menambah member merupakan fungsi yang mengextends dari fungsi Melihat daftar member, terdapat fungsi Melakukan validasi pembayaran, Melihat detail pemesanan member, Melihat detail pemesan, Membatalkan pembayaran, Melihat detail pembayaran yang mengextends dari fungsi Melihat daftar pembayaran member dan juga ada fungsi Menolak event, Melakukan validasi kelas event yang mengextends dari fungsi Melihat daftar kelas event serta fungsi Melihat dashboard dan Logout.

#### 4.2.4 *Use Case Scenario*

*Use Case Scenario* adalah proses penjabaran deskripsi alur kerja sistem secara mendetail dari *use case* diagram. Dalam *use case scenario* berisi nomor *use case*, nama *use case*, nama aktor, tujuan, *pre-condition*, *main flow*, *alternative flow* dan *post condition*. Bagian *pre-condition* berisi kondisi yang terpenuhi sebelum sistem menjalankan *main flow*. Bagian *main flow* merupakan proses yang harus dijalankan pada sistem secara berurutan. Bagian *alternative flow* berisi alur kondisi tertentu yang tidak di deklarasikan pada *main flow*.

Bagian terakhir yaitu, *post-condition* berisi kondisi akhir setelah aktor menjalankan proses pada *main flow*. *Use case skenario* aplikasi pemesanan instruktur olahraga kota malang berbasis web dapat dilihat pada tabel dibawah ini:

**Tabel 4.8 Use Case Scenario Guest Register sebagai Member**

Tabel 4.9 *Use Case Scenario Guest Register sebagai Member* merupakan table *use case scenario* yang dijalankan *Guest* untuk *Register sebagai Member*.

Nomor Use case	APIKO-F-1-001
Nama	<i>Register</i>
Aktor	<i>Guest</i>
Tujuan	Menyediakan layanan bagi <i>guest</i> melakukan <i>registrasi</i> pada aplikasi untuk menjadi <i>member</i>
<i>Pre-condition</i>	Halaman <i>registrasi member</i> telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor mengisi form nama lengkap, alamat email, jenis kelamin, no telepon, password</li> <li>2. Aktor menekan tombol daftar</li> <li>3. Aplikasi menyimpan data registrasi aktor sebagai member pada database</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor mengisi form nama lengkap, alamat email, jenis kelamin, no telepon, password</li> <li>2. Aktor mengisi email yang sudah terdaftar</li> <li>3. Aktor menekan tombol daftar</li> <li>4. Aplikasi gagal menyimpan data registrasi dan menampilkan pesan "email telah terdaftar"</li> </ol>
<i>Alternative Flow 2</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form registrasi</li> <li>2. Aktor menekan tombol daftar</li> <li>3. Aplikasi gagal menyimpan data registasi dan menampilkan pesan "please fill out this file"</li> </ol>
<i>Post-condition</i>	<ol style="list-style-type: none"> <li>1. Data <i>registasi guest</i> sebagai <i>member</i> berhasil tersimpan pada <i>database</i></li> <li>2. Aplikasi menampilkan alert "Sukses Mendaftar Member"</li> </ol>

**Tabel 4.10 Use Case Scenario Register Guest sebagai Instruktur Olahraga**

Tabel 4.11 *Use Case Scenario Register Guest sebagai Instruktur Olahraga* merupakan table *use case scenario* yang dijalankan *Guest* untuk *Register sebagai Instruktur Olahraga*.

Nomor Use case	APIKO-F-1-002
Nama	<i>Register</i>

Aktor	<i>Guest</i>
Tujuan	Menyediakan layanan bagi <i>guest</i> melakukan registrasi pada aplikasi untuk menjadi instruktur olahraga
<i>Pre-condition</i>	Halaman registrasi instruktur telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor mengisi form nama lengkap, alamat email, jenis kelamin, alamat, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password</li> <li>2. Aktor menekan tombol daftar instruktur</li> <li>3. Aplikasi melakukan validasi data registrasi</li> <li>4. Aplikasi menyimpan data registrasi aktor sebagai instruktur olahraga pada database</li> <li>5. Aplikasi memberi notifikasi aktor berhasil terdaftar menjadi instruktur olahraga melalui email</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor mengisi form nama lengkap, alamat email, jenis kelamin, alamat, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password</li> <li>2. Aktor memasukkan email yang sudah terdaftar</li> <li>3. Aktor menekan tombol daftar instruktur</li> <li>4. Aplikasi gagal menyimpan data registrasi dan menampilkan pesan "email telah terdaftar"</li> </ol>
<i>Alternative Flow 2</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form registrasi</li> <li>2. Aktor menekan tombol daftar instruktur</li> <li>3. Aplikasi gagal menyimpan data registrasi dan menampilkan pesan "please fill out this file"</li> </ol>
<i>Post-condition</i>	<ol style="list-style-type: none"> <li>1. Data registasi <i>guest</i> sebagai instruktur olahraga berhasil tersimpan pada database</li> <li>2. Aplikasi menampilkan "Sukses Mendaftar Instruktur, Tunggu Konfirmasi dari Admin"</li> </ol>

**Tabel 4.12 Use Case Scenario Login**

Tabel 4.13 Use Case Scenario Login merupakan table use case scenario login.

Nomor Use case	APIKO-F-1-003
Nama	<i>Login</i>
Aktor	<i>Guest</i>
Tujuan	Menyediakan layanan autentikasi bagi <i>Guest</i>
<i>Pre-condition</i>	Halaman login telah tersedia

<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor memasukkan email dan password pada form login</li> <li>2. Aktor menekan tombol login</li> <li>3. Aplikasi melakukan validasi email dan password</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor memasukkan email dan password yang tidak valid</li> <li>2. Aplikasi gagal melakukan autentikasi dan menampilkan pesan “Akun Tidak Terdaftar / Password Salah Pastikan Email &amp; Password sudah benar”</li> </ol>
<i>Alternative Flow 2</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form login</li> <li>2. Aplikasi gagal melakukan autentikasi dan menampilkan pesan “please fill out this file”</li> </ol>
<i>Post-condition</i>	<ol style="list-style-type: none"> <li>1. Proses autentikasi berhasil</li> <li>2. Halaman Utama ditampilkan</li> </ol>

**Tabel 4.14 Use Case Scenario Mencari Instruktur Olahraga**

Tabel 4.15 Use Case Scenario Mencari Instruktur Olahraga merupakan table use case scenario untuk mencari instruktur olahraga

Nomor Use case	APIKO-F-1-004, APIKO-F-2-001, APIKO-F-3-015
Nama	Mencari Instruktur Olahraga
Aktor	Guest, Member, Instruktur Olahraga
Tujuan	Menyediakan layanan untuk mencari instruktur olahraga
<i>Pre-condition</i>	Halaman Home telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor memasukkan jenis olahraga pada field pencarian</li> <li>2. Aplikasi melakukan pencarian terkait inputan yang diberikan actor</li> <li>3. Aplikasi menampilkan foto, nama, jenis olahraga, harga, sesi, durasi, jumlah murid, rating</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor memasukkan inputan jenis olahraga pada field pencarian yang tidak terdaftar jenis olahraganya</li> <li>2. Aplikasi gagal melakukan pencarian terkait inputan dan menampilkan pesan “tidak ditemukan”</li> </ol>
<i>Post-condition</i>	1. Menampilkan daftar instruktur olahraga yang melatih jenis olahraga yang diinputkan

**Tabel 4.16 Use Case Scenario Mencari Kelas Event**

Tabel 4.17 Use Case Scenario Mencari Kelas Event merupakan table use case scenario untuk mencari kelas event.

Nomor Use case	APIKO-F-1-006, APIKO-F-2-003, APIKO-F-3-011
Nama	Mencari Kelas Event



Aktor	<i>Guest, Member, Instruktur Olahraga</i>
Tujuan	Menyediakan layanan untuk mencari kelas event yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman View Event telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu kelas</li> <li>2. Aktor memilih event</li> <li>3. Aplikasi melakukan pencarian terkait kelas event yang ditawarkan instruktur</li> <li>4. Aplikasi menampilkan nama paket event, foto instruktur, nama instruktur dan jenis olahraga, sesi, harga, rating</li> </ol>
<i>Alternative Flow 1</i>	Tidak ada
<i>Post-condition</i>	1. Menampilkan daftar instruktur olahraga yang membuat kelas event

**Tabel 4.18 Use Case Scenario Melihat Detail Profil Instruktur Olahraga**

Tabel 4.19 Use Case Scenario Melihat Detail Profil Instruktur Olahraga merupakan table use case scenario untuk melihat detail profil instruktur olahraga

Nomor Use case	APIKO-F-1-007, APIKO-F-2-005, APIKO-F-3-017
Nama	Melihat Detail Profil Instruktur Olahraga
Aktor	<i>Guest, Member, Instruktur Olahraga</i>
Tujuan	Menyediakan layanan untuk menampilkan detail profil instruktur olahraga dan daftar kelasnya
<i>Pre-condition</i>	Halaman Profil Instruktur telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih nama instruktur</li> <li>2. Aplikasi menampilkan seluruh informasi profil instruktur olahraga yang berisi foto, nama, jenis kelamin, alamat, no telepon, alamat email, lama melatih, pengalaman melatih, keahlian lain, sertifikasi</li> <li>3. Aplikasi menampilkan rincian kelas yang berisi durasi, jumlah murid, sesi dan harga</li> </ol>
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	1. Informasi profil instruktur olahraga dan rincian kelas berhasil ditampilkan pada halaman profil

**Tabel 4.20 Use Case Scenario Melakukan Pemesanan Kelas Event**

Tabel 4.21 Use Case Scenario Melakukan Pemesanan Kelas Event merupakan table use case scenario untuk pemesanan event.

Nomor Use case	APIKO-F-2-004
Nama	Melakukan Pemesanan Kelas Event

Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melakukan pemesanan kelas event yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman Pemesanan telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol pesan instruktur</li> <li>2. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, sisa kouta, sesi, harga, jadwal latihan, jam</li> <li>3. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon, lokasi latihan</li> <li>4. Aktor menekan tombol pesan</li> <li>5. Aplikasi menyimpan data pemesanan</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih instruktur yang ingin dipesan dengan kouta yang sudah habis</li> <li>2. Aktor menekan tombol pesan instruktur</li> <li>3. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga, jadwal latihan, jam</li> <li>4. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon, lokasi latihan</li> <li>5. Aktor menekan tombol pesan</li> <li>6. Aplikasi menyimpan data pemesanan</li> <li>7. Aplikasi menampilkan tulisan "Kouta Penuh"</li> </ol>
<i>Post-condition</i>	<ol style="list-style-type: none"> <li>1. Data pemesanan kelas event berhasil disimpan pada database</li> <li>2. Aplikasi menampilkan kode pembayaran</li> </ol>

**Tabel 4.22 Use Case Scenario Melihat Kode Pembayaran**

Tabel 4.23 Use Case Scenario Melihat Kode Pembayaran merupakan table use case scenario untuk melihat kode pembayaran.

Nomor Use case	APIKO-F-2-006
Nama	Melihat Kode Pembayaran
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melihat kode pembayaran sebelum melakukan pembayaran pemesanan
<i>Pre-condition</i>	Form Kode Pembayaran telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aplikasi menampilkan form yang berisi nama bank, kode bayar, jumlah pembayaran</li> </ol>
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi Kode Pembayaran berhasil ditampilkan pada halaman pembayaran



**Tabel 4.24 Use Case Scenario Melihat Profil Member**

Tabel 4.25 *Use Case Scenario* Melihat Profil Member merupakan table *use case scenario* untuk melihat profil member.

Nomor Use case	APIKO-F-2-007
Nama	Melihat Profil Member
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melihat profil member itu sendiri
<i>Pre-condition</i>	Halaman Profil Member telah tersedia
<i>Main Flow</i>	1.Aktor menekan Menu Akun Saya 2.Aplikasi menampilkan profil member yang berisi nama lengkap, no telepon, alamat, jenis kelamin, alamat email, password
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi profil member berhasil ditampilkan di halaman profil member

**Tabel 4.26 Use Case Scenario Edit Profil Member**

Tabel 4.27 *Use Case Scenario* Edit Profil Member merupakan table *use case scenario* untuk edit profil member.

Nomor Use case	APIKO-F-2-008
Nama	Edit Profil Member
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melakukan perubahan informasi profil member
<i>Pre-condition</i>	Halaman Edit Profil Member telah tersedia
<i>Main Flow</i>	1.Aktor menekan tombol edit profil 2.Aktor memasukkan data yang ingin dilakukan perubahan 3.Aktor menekan tombol Edit Member 4.Aplikasi menyimpan data perubahan profil
<i>Alternative Flow 1</i>	1.Aktor tidak mengisi seluruh field pada form edit profil 2.Aktor menekan tombol Edit Member 3.Aplikasi gagal menyimpan data edit profil dan menampilkan pesan "plase fill out this file "
<i>Post-condition</i>	Data perubahan profil berhasil terupdate didatabase

**Tabel 4.28 Use Case Scenario Memberi Rating Instruktur**

Tabel 4.29 *Use Case Scenario* Memberi Rating Instruktur merupakan table *use case scenario* untuk memberi rating instruktur.

Nomor Use case	APIKO-F-2-009
----------------	---------------

Nama	Memberi Rating Instruktur
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk memberi rating instruktur olahraga setelah selesai latihan
<i>Pre-condition</i>	Halaman Jadwal Member telah tersedia
<i>Main Flow</i>	1. Aktor menekan menu lihat jadwal 2. Aktor memasukkan tingkatan rating pada bintang 3. Aktor menekan tombol ok pada alert 4. Aplikasi menyimpan data rating
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data rating berhasil tersimpan didatabase

**Tabel 4.30 Use Case Scenario Melakukan Konfirmasi Pembayaran**

Tabel 4.31 Use Case Scenario Melakukan Konfirmasi Pembayaran merupakan table use case scenario untuk mengkonfirmasi pembayaran.

Nomor Use case	APIKO-F-2-010
Nama	Melakukan konfirmasi pembayaran
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk mengkonfirmasi pembayaran setelah melakukan transfer
<i>Pre-condition</i>	Halaman Pembayaran telah tersedia
<i>Main Flow</i>	1. Aktor menekan menu lihat pembayaran pada halaman profil member 2. Aktor menekan field lihat pada table konfirmasi pembayaran 3. Aktor menginputkan file foto bukti transfer 4. Aktor menekan tombol konfirmasi pembayaran 5. Aplikasi menyimpan data konfirmasi pembayaran 6. Aplikasi menampilkan pesan "Anda berhasil mengonfirmasi pembayaran. Silahkan tunggu pembayaran anda dikonfirmasi oleh admin"
<i>Alternative Flow 1</i>	1. Aktor tidak menginputkan file foto 2. Aktor menekan tombol konfirmasi pembayaran 3. Aplikasi gagal menyimpan data foto dan menampilkan pesan "please select a file"
<i>Post-condition</i>	Data konfirmasi pembayaran berhasil tersimpan didatabase

**Tabel 4.32 Use Case Scenario Melakukan Pembatalan Pemesanan**

Tabel 4.33 Use Case Scenario Melakukan Pembatalan Pemesanan merupakan table use case scenario untuk membatalkan pemesanan.

Nomor Use case	APIKO-F-2-011
----------------	---------------

Nama	Melakukan Pembatalan Pemesanan
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk membatalkan pemesanan
<i>Pre-condition</i>	Form Konfirmasi Pembayaran telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol batalkan pemesanan 2. Aplikasi mengupdate data pembatalan pemesanan
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data pembatalan pemesanan berhasil diupdate pada database

**Tabel 4.34 Use Case Scenario Hapus Akun Member**

Tabel 4.35 *Use Case Scenario* Hapus Akun Member merupakan table *use case scenario* untuk menghapus akun *member*.

Nomor <i>Use case</i>	APIKO-F-2-012
Nama	Hapus Akun Member
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk menghapus akun yang sudah tidak ingin digunakan
<i>Pre-condition</i>	Halaman Profil Member telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol hapus akun 2. Aplikasi mengupdate data akun yang terhapus
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data akun yang terhapus berhasil diupdate pada database

**Tabel 4.36 Use Case Scenario Melihat Jadwal Latihan Member**

Tabel 4.37 *Use Case Scenario* Melihat Jadwal Latihan Member merupakan table *use case scenario* untuk lihat jadwal latihan *member*.

Nomor <i>Use case</i>	APIKO-F-2-013
Nama	Melihat Jadwal Latihan Member
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melihat jadwal latihan member
<i>Pre-condition</i>	Halaman Jadwal Member telah tersedia
<i>Main Flow</i>	1. Aktor menekan menu lihat jadwal pada halaman profil member 2. Aplikasi menampilkan jadwal latihan member yang berisi sesi, jenis, nama pelatih, jam, tempat, rating, durasi, agenda, jumlah murid

<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi jadwal latihan member berhasil ditampilkan pada halaman jadwal member

**Tabel 4.38 Use Case Scenario Melihat Profil Instruktur Olahraga**

Tabel 4.39 Use Case Scenario Melihat Profil Instruktur Olahraga merupakan table use case scenario untuk melihat profil instruktur olahraga.

Nomor Use case	APIKO-F-3-001
Nama	Melihat Profil Instruktur Olahraga
Aktor	Instruktur Olahraga
Tujuan	Menyediakan layanan untuk melihat profil instruktur olahraga itu sendiri
<i>Pre-condition</i>	Halaman Profil Instruktur Olahraga telah tersedia
<i>Main Flow</i>	1. Aktor menekan Menu Akun Saya 2. Aplikasi menampilkan profil instruktur olahraga yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no ktp, nama bank, rekening bank, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi profil instruktur olahraga berhasil ditampilkan

**Tabel 4.40 Use Case Scenario Edit Profil Instruktur Olahraga**

Tabel 4.41 Use Case Scenario Edit Profil Instruktur Olahraga merupakan table use case scenario untuk edit profil instruktur olahraga.

Nomor Use case	APIKO-F-3-002
Nama	Edit Profil Instruktur Olahraga
Aktor	Instruktur Olahraga
Tujuan	Menyediakan layanan untuk melakukan perubahan informasi profil instruktur
<i>Pre-condition</i>	Halaman Edit Profil Instruktur Olahraga telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol edit profil 2. Aktor memasukkan data yang ingin dilakukan perubahan 3. Aktor menekan tombol edit instruktur 4. Aplikasi menyimpan data perubahan profil
<i>Alternative Flow 1</i>	1. Aktor tidak mengisi seluruh field pada form edit profil 2. Aktor menekan tombol edit instruktur 3. Aplikasi gagal menyimpan data edit profil dan menampilkan pesan "please fill out this file"
<i>Post-condition</i>	Data perubahan profil berhasil terupdate didatabase

**Tabel 4.42 Use Case Scenario Melakukan Pembuatan Kelas Reguler**

Tabel 4.43 Use Case Scenario Melakukan Pembuatan Kelas Reguler merupakan table *use case scenario* untuk pembuatan kelas reguler.

Nomor Use case	APIKO-F-3-003
Nama	Melakukan Pembuatan Jadwal Paket Kelas Reguler
Aktor	Instruktur Olahraga
Tujuan	Menyediakan layanan untuk melakukan pembuatan kelas reguler
Pre-condition	Halaman Add Reguler telah tersedia
Main Flow	<ol style="list-style-type: none"> <li>1. Aktor menekan menu Tambah Jadwal Reguler</li> <li>2. Aktor mengisi form pembuatan jadwal yang berisi kelas, sesi, jumlah murid, durasi, harga, agenda</li> <li>3. Aktor memilih rentang jam latihan</li> <li>4. Aktor menekan tombol tambah reguler</li> <li>5. Aplikasi menyimpan data kelas reguler</li> <li>6. Aplikasi menampilkan pesan "Sukses mendaftarkan event. Sukses mendaftarkan event. Silahkan tunggu konfirmasi admin"</li> </ol>
Alternative Flow 1	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form pembuatan jadwal kelas</li> <li>2. Aktor menekan tombol tambah reguler</li> <li>3. Aplikasi gagal menyimpan data pembuatan kelas dan menampilkan pesan "please fill out this file"</li> </ol>
Post-condition	Data pembuatan kelas reguler berhasil dibuat didatabase

**Tabel 4.44 Use Case Scenario Melakukan Pembuatan Kelas Event**

Tabel 4.45 Use Case Scenario Melakukan Pembuatan Kelas Event merupakan table *use case scenario* untuk pembuatan kelas event.

Nomor Use case	APIKO-F-3-004
Nama	Melakukan Pembuatan Kelas Event
Aktor	Instruktur Olahraga
Tujuan	Menyediakan layanan untuk melakukan pembuatan kelas event
Pre-condition	Halaman Add Event telah tersedia
Main Flow	<ol style="list-style-type: none"> <li>1. Aktor menekan menu Tambah Jadwal Event</li> <li>2. Aktor mengisi form pembuatan jadwal yang berisi agenda, sesi, durasi, jumlah murid, harga, jadwal, lokasi, jam</li> <li>3. Aktor menekan tombol tambah event</li> <li>4. Aplikasi menyimpan data kelas reguler</li> <li>5. Aplikasi menampilkan pesan "Sukses mendaftarkan event. Sukses mendaftarkan event. Silahkan tunggu konfirmasi"</li> </ol>

	admin”
<i>Alternative Flow 1</i>	1.Aktor tidak mengisi seluruh field pada form pembuatan jadwal event 2.Aktor menekan tombol tambah reguler 3.Aplikasi gagal menyimpan data pembuatan kelas dan menampilkan pesan “please fill out this file”
<i>Post-condition</i>	Data pembuatan kelas event berhasil dibuat didatabase

**Tabel 4.46 Use Case Scenario Melihat Jadwal Latihan Instruktur**

Tabel 4.47 *Use Case Scenario* Melihat Jadwal Latihan Instruktur merupakan table *use case scenario* untuk melihat jadwal latihan instruktur.

Nomor <i>Use case</i>	APIKO-F-3-005
Nama	Melihat Jadwal Latihan Instruktur
Aktor	Instruktur Olahraga
Tujuan	Menyediakan layanan untuk melihat jadwal latihan instruktur olahraga
<i>Pre-condition</i>	Halaman Jadwal Instruktur Olahraga telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu lihat jadwal pada halaman profil instruktur olahraga 2.Aplikasi menampilkan jadwal latihan member yang berisi sesi, jenis, nama member, jam, agenda, tempat , jumlah murid, rating, durasi
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi jadwal latihan instruktur olahraga berhasil ditampilkan pada halaman jadwal member

**Tabel 4.48 Use Case Scenario Melakukan Pembuatan Kelas Paket**

Tabel 4.49 *Use Case Scenario* Melakukan Pembuatan Kelas Paket merupakan table *use case scenario* untuk pembuatan kelas paket.

Nomor <i>Use case</i>	APIKO-F-3-008
Nama	Melakukan Pembuatan Kelas Paket
Aktor	Instruktur Olahraga
Tujuan	Menyediakan layanan untuk melakukan pembuatan kelas paket
<i>Pre-condition</i>	Halaman Add Paket telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu Tambah Jadwal Paket 2.Aktor mengisi form pembuatan jadwal yang berisi kelas, sesi, jumlah murid, durasi, harga, agenda 3.Aktor memilih rentang jam latihan 4.Aktor menekan tombol tambah paket 5.Aplikasi menyimpan data kelas reguler

	6. Aplikasi menampilkan pesan “Sukses mendaftar event. Sukses mendaftar event.Silahkan tunggu konfirmasi admin”
<i>Alternative Flow 1</i>	1.Aktor tidak mengisi seluruh field pada form pembuatan jadwal kelas 2.Aktor menekan tombol tambah group 3.Aplikasi gagal menyimpan data pembuatan kelas dan menampilkan pesan “please fill out this file ”
<i>Post-condition</i>	Data pembuatan kelas paket berhasil terupdate didatabase

**Tabel 4.50 Use Case Scenario Hapus Akun Instruktur Olahraga**

Tabel 4.51 Use Case Scenario Hapus Akun Instruktur merupakan table use case scenario untuk menghapus untuk instruktur olahraga.

Nomor Use case	APIKO-F-3-007
Nama	Hapus Akun Instruktur
Aktor	Instruktur
Tujuan	Menyediakan layanan untuk menghapus akun yang sudah tidak ingin digunakan
<i>Pre-condition</i>	Halaman Profil Instruktur telah tersedia
<i>Main Flow</i>	1.Aktor menekan tombol hapus akun 2.Aplikasi mengupdate data akun yang terhapus
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data akun yang terhapus berhasil diupdate pada database

**Tabel 4.52 Use Case Scenario Melihat Daftar Instruktur Terverifikasi**

Tabel 4.53 Use Case Scenario Melihat Daftar Instruktur Terverifikasi merupakan table use case scenario untuk melihat daftar instruktur yang terverifikasi.

Nomor Use case	APIKO-F-4-001
Nama	Melihat Daftar Instruktur Terverifikasi
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat daftar instruktur yang sudah diverifikasi pendaftarannya
<i>Pre-condition</i>	Halaman View Instruktur telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu Instruktur 2.Aktor memilih menu Instruktur Terverifikasi 2.Aplikasi menampilkan nama, jenis kelamin, alamat email, no telepon
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi daftar instruktur olahraga terverifikasi berhasil ditampilkan pada halaman View Instruktur

**Tabel 4.54 Use Case Scenario Melihat Detail Profil Pendaftaran Instruktur**

Tabel 4.55 *Use Case Scenario* Melihat Detail Profil Pendaftaran Instruktur merupakan table *use case scenario* untuk melihat detail profil pendaftaran instruktur.

Nomor <i>Use case</i>	APIKO-F-4-002
Nama	Melihat Detail Daftar Instruktur
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat detail profil instruktur yang mendaftar
<i>Pre-condition</i>	Halaman View Requested Instruktur telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol konfirmasi 2. Aktor memilih tombol detail pendaftaran 3. Aplikasi menampilkan nama lengkap, alamat email, jenis kelamin, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi detail daftar instruktur olahraga berhasil ditampilkan pada halaman view requested instruktur

**Tabel 4.56 Use Case Scenario Melakukan Validasi Pendaftaran Instruktur**

Tabel 4.57 *Use Case Scenario* Melakukan Validasi Pendaftaran Instruktur merupakan table *use case scenario* untuk memvalidasi pendaftaran instruktur.

Nomor <i>Use case</i>	APIKO-F-4-003
Nama	Melakukan Validasi Pendaftaran Instruktur
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk memvalidasi pendaftaran instruktur
<i>Pre-condition</i>	Halaman View Requested Instruktur telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol konfirmasi 2. Aktor memilih tombol terima pendaftaran 3. Aplikasi menyimpan data validasi pendaftaran instruktur olahraga 4. Aplikasi memberi notifikasi pendaftaran berhasil divalidasi melalui email instruktur olahraga
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data validasi pendaftaran instruktur olahraga berhasil tersimpan didatabase

**Tabel 4.58 Use Case Scenario Menolak Pendaftaran Instruktur**

Tabel 4.59 Use Case Scenario Menolak Pendaftaran Instruktur merupakan table *use case scenario* untuk menolak pendaftaran instruktur.

Nomor Use case	APIKO-F-4-004
Nama	Menolak Pendaftaran Instruktur
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk menolak pendaftaran instruktur
<i>Pre-condition</i>	Halaman Pendaftaran telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol konfirmasi</li> <li>2. Aktor memilih tombol tolak pendaftaran</li> <li>3. Aplikasi menyimpan data penolakan pendaftaran instruktur olahraga</li> <li>4. Aplikasi memberi notifikasi pendaftaran ditolak melalui email instruktur olahraga</li> </ol>
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data penolakan pendaftaran instruktur olahraga berhasil tersimpan didatabase

**Tabel 4.60 Use Case Scenario Melihat Daftar Pembayaran Member**

Tabel 4.61 Use Case Scenario Melihat Daftar Pembayaran Member merupakan table *use case scenario* untuk melihat daftar pembayaran *member*.

Nomor Use case	APIKO-F-4-005
Nama	Melihat Daftar Pembayaran Member
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat daftar pembayaran yang dilakukan member
<i>Pre-condition</i>	Halaman View Pembayaran telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan menu Pembayaran</li> <li>2. Aplikasi menampilkan daftar tabel detail pemesan, detail pemesanan, detail pembayaran, status pembayaran dan option yang berisi tombol konfirmasi pembayaran dan tolak pembayaran</li> </ol>
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi daftar pembayaran member berhasil ditampilkan pada halaman pembayaran

**Tabel 4.62 Use Case Scenario Melakukan Validasi Pembayaran**

Tabel 4.63 Use Case Scenario Melakukan Validasi Pembayaran merupakan table *use case scenario* untuk memvalidasi pembayaran.

Nomor Use case	APIKO-F-4-006
Nama	Melakukan Validasi Pembayaran
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk memvalidasi pembayaran yang dilakukan member
<i>Pre-condition</i>	Halaman View Pembayaran telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol option 2. Aktor menekan tombol konfirmasi 3. Aplikasi menyimpan data validasi pembayaran
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data validasi pembayaran berhasil tersimpan didatabase

**Tabel 4.64 Use Case Scenario Membatalkan Pembayaran**

Tabel 4.65 Use Case Scenario Membatalkan Pembayaran merupakan table *use case scenario* untuk membatalkan pembayaran.

Nomor Use case	APIKO-F-4-007
Nama	Membatalkan Pembayaran
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk membatalkan pembayaran yang dilakukan member
<i>Pre-condition</i>	Halaman View Pembayaran telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol option 2. Aktor memilih tombol tolak 3. Aplikasi menyimpan data perubahan pembatalan pembayaran
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data perubahan pembatalan pembayaran berhasil terupdate didatabase

**Tabel 4.66 Use Case Scenario Melihat Daftar Member**

Tabel 4.67 Use Case Scenario Melihat Daftar Member merupakan table *use case scenario* untuk melihat daftar member.

Nomor Use case	APIKO-F-4-008
Nama	Melihat Daftar Member

Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat daftar member
<i>Pre-condition</i>	Halaman View Member telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu Member 2.Aplikasi menampilkan berisi nama lengkap, no telepon, jenis kelamin, alamat email
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi daftar member berhasil ditampilkan pada halaman View Member

**Tabel 4.68 Use Case Scenario Melihat Daftar Kelas Event**

Tabel 4.69 *Use Case Scenario* Melihat Daftar Kelas Event merupakan table *use case scenario* untuk melihat daftar kelas event.

Nomor Use case	APIKO-F-4-009
Nama	Melihat Daftar Paket Event
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat daftar kelas event
<i>Pre-condition</i>	Halaman view event telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu Event 2.Aplikasi menampilkan berisi agenda, status, sesi, durasi, jumlah murid, harga, jadwal, lokasi
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi daftar kelas event berhasil ditampilkan pada halaman event

**Tabel 4.70 Use Case Scenario Melakukan Validasi Kelas Event**

Nomor Use case	APIKO-F-4-010
Nama	Melakukan Validasi Kelas Event
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk memvalidasi kelas event yang akan diadakan instruktur olahraga
<i>Pre-condition</i>	Halaman View event telah tersedia
<i>Main Flow</i>	1.Aktor menekan tombol option 2.Aktor memilih tombol terima event 3.Aplikasi menyimpan data validasi kelas event
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data validasi paket event berhasil tersimpan didatabase

**Tabel 4.71 Use Case Scenario Menolak Kelas Event**

Tabel 4.72 *Use Case Scenario* Menolak Kelas Event merupakan table *use case scenario* untuk menolak kelas event.

Nomor <i>Use case</i>	APIKO-F-4-011
Nama	Menolak Kelas Event
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk menolak kelas event yang akan diadakan instruktur olahraga
<i>Pre-condition</i>	Halaman view event telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol option 2. Aktor memilih tombol tolak event 3. Aplikasi menyimpan data perubahan penolakan kelas event
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data perubahan penolakan paket event berhasil terupdate didatabase

**Tabel 4.73 Use Case Scenario Logout**

Tabel 4.74 *Use Case Scenario Logout* merupakan table *use case scenario* untuk melakukan *logout*.

Nomor <i>Use case</i>	APIKO-F-4-012 ,APIKO-F-3-009, APIKO-F-2-014
Nama	<i>Logout</i>
Aktor	Admin, Member, Instruktur Olahraga
Tujuan	Menyediakan layanan bagi aktor untuk keluar dari akun aplikasi
<i>Pre-condition</i>	1. Aktor sudah terautentifikasi pada proses login
<i>Main Flow</i>	1. Aktor menekan menu Logout 2. Aplikasi menghapus data session pada aplikasi
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	1. Data session pada aplikasi berhasil dihapus 2. Aplikasi menampilkan halaman utama

**Tabel 4.75 Use Case Scenario Melihat Daftar Instruktur Belum Terverifikasi**

Tabel 4.76 *Use Case Scenario* Melihat Daftar Instruktur Belum Terverifikasi merupakan *use case scenario* untuk melihat daftar instruktur yang belum terverifikasi.

Nomor <i>Use case</i>	APIKO-F-4-013
Nama	Melihat Daftar Instruktur Terverifikasi

Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat daftar instruktur yang belum diverifikasi pendaftarannya
<i>Pre-condition</i>	Halaman View Request Instruktur telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu Instruktur 2.Aktor memilih menu Pendaftar Instruktur 2.Aplikasi menampilkan nama, alamat email, no telepon
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi daftar instruktur olahraga yang belum terverifikasi berhasil ditampilkan pada halaman View Request Instruktur

**Tabel 4.77 Use Case Scenario Menambahkan Instruktur**

Tabel 4.78 Use Case Scenario Menambahkan Instruktur merupakan table use case scenario untuk menambahkan instruktur.

Nomor Use case	APIKO-F-4-015
Nama	Menambahkan Instruktur
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan bagi admin melakukan pembuatan instruktur olahraga
<i>Pre-condition</i>	Halaman View Instruktur telah tersedia
<i>Main Flow</i>	1. Aktor mengisi form nama lengkap,alamat email, jenis kelamin, alamat, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password 2.Aktor menekan tombol tambah instruktur 3.Aplikasi menyimpan data pembuataninstruktur olahragapada database
<i>Alternative Flow 1</i>	1.Aktor memasukan email yang sudah terdaftar 2.Aktor menekan tombol tambah instruktur 3.Aplikasi gagal menyimpan data pembuatan dan menampilkan pesan "email telah terdaftar"
<i>Alternative Flow 2</i>	1.Aktor tidak mengisi seluruh field pada form tambah instruktur 2.Aktor menekan tombol tambah instruktur 3.Aplikasi gagal menyimpan data pembuatan dan menampilkan pesan "please fill out this file"
<i>Post-condition</i>	Data registasi pembuatan instruktur olahraga berhasil tersimpan pada database



**Tabel 4.79 Use Case Scenario Menambahkan Member**

Tabel 4.80 *Use Case Scenario* Menambahkan Member merupakan table *use case scenario* untuk menambahkan *member*.

Nomor <i>Use case</i>	APIKO-F-4-014
Nama	Menambahkan Member
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan bagi admin melakukan pembuatan member
<i>Pre-condition</i>	Halaman View Member telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor mengisi form nama lengkap, alamat email, jenis kelamin, no telepon, password</li> <li>2. Aktor menekan tombol tambah member</li> <li>3. Aplikasi menyimpan data pembuatan member pada database</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor memasukkan email yang sudah terdaftar</li> <li>2. Aktor menekan tombol tambah member</li> <li>3. Aplikasi gagal menyimpan data registrasi dan menampilkan pesan "email telah terdaftar"</li> </ol>
<i>Alternative Flow 2</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form registrasi</li> <li>2. Aktor menekan tombol tambah instruktur</li> <li>3. Aplikasi gagal menyimpan data registrasi dan menampilkan pesan "please fill out this file"</li> </ol>
<i>Post-condition</i>	Data pembuatan member berhasil tersimpan pada database

**Tabel 4.81 Use Case Scenario Hapus Member**

Tabel 4.82 *Use Case Scenario* Hapus Member merupakan table *use case scenario* untuk menghapus *member*.

Nomor <i>Use case</i>	APIKO-F-4-016
Nama	Hapus Member
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk menghapus member
<i>Pre-condition</i>	Halaman View Member telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol option</li> <li>2. Aktor memilih tombol delete profile</li> <li>3. Aplikasi mengupdate data akun member yang terhapus</li> </ol>

<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data akun member yang terhapus berhasil diupdate pada database

**Tabel 4.83 Use Case Scenario Hapus Instruktur**

Tabel 4.84 *Use Case Scenario* Hapus Instruktur merupakan table *use case scenario* untuk menghapus instruktur.

Nomor <i>Use case</i>	APIKO-F-4-017
Nama	Hapus Instruktur
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk menghapus instruktur
<i>Pre-condition</i>	Halaman View Instruktur telah tersedia
<i>Main Flow</i>	1.Aktor menekan tombol option 2.Aktor memilih tombol delete profile 3.Aplikasi mengupdate data akun instruktur yang terhapus
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Data akun instruktur yang terhapus berhasil diupdate pada database

**Tabel 4.85 Use Case Scenario Melihat Dashboard**

Tabel 4.86 *Use Case Scenario* Melihat Dashboard merupakan table *use case scenario* untuk melihat *dashboard*.

Nomor <i>Use case</i>	APIKO-F-4-018
Nama	Melihat Dashboard
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat dashboard
<i>Pre-condition</i>	Halaman Dashboard telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu overview 2.Aplikasi menampilkan dashboard admin
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi dashboard dapat ditampilkan pada halaman dashboard

**Tabel 4.87 Use Case Scenario Edit Instruktur**

Tabel 4.88 *Use Case Scenario* Edit Instruktur merupakan table *use case scenario* untuk edit instruktur.

Nomor Use case	APIKO-F-4-020
Nama	Edit Instruktur
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melakukan perubahan informasi profil instruktur
<i>Pre-condition</i>	Halaman View Instruktur telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol option</li> <li>2. Aktor memilih tombol edit profile</li> <li>3. Aktor memasukkan data yang ingin dilakukan perubahan</li> <li>4. Aktor menekan tombol edit instruktur</li> <li>5. Aplikasi menyimpan data perubahan profil</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form edit instruktur</li> <li>2. Aktor menekan tombol edit instruktur</li> <li>3. Aplikasi gagal menyimpan data edit profil dan menampilkan pesan "please fill out this file "</li> </ol>
<i>Post-condition</i>	Data perubahan profil berhasil terupdate didatabase

**Tabel 4.89 Use Case Scenario Edit Member**

Tabel 4.90 *Use Case Scenario* Edit Member merupakan table *use case scenario* untuk edit member.

Nomor Use case	APIKO-F-4-019
Nama	Edit Member
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melakukan perubahan informasi profil edit
<i>Pre-condition</i>	Halaman View Member telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol option</li> <li>2. Aktor memilih tombol edit profile</li> <li>3. Aktor memasukkan data yang ingin dilakukan perubahan</li> <li>4. Aktor menekan tombol edit member</li> <li>5. Aplikasi menyimpan data perubahan profil</li> </ol>
<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form edit member</li> <li>2. Aktor menekan tombol edit member</li> <li>3. Aplikasi gagal menyimpan data edit profil dan menampilkan pesan "please fill out this file "</li> </ol>



<i>Post-condition</i>	Data perubahan profil berhasil terupdate didatabase
-----------------------	---

**Tabel 4.91 Use Case Scenario Edit Foto**

Tabel 4.92 Use Case Scenario Edit Foto merupakan table *use case scenario* untuk edit foto.

Nomor Use case	APIKO-F-2-014 & APIKO-3-010
Nama	Edit Foto
Aktor	Member dan Instruktur
Tujuan	Menyediakan layanan untuk mengganti foto
<i>Pre-condition</i>	Form edit foto sudah tersedia
<i>Main Flow</i>	1.Aktor menekan tombol choose file 2.Aktor memilih foto 3.Aktor menekan tombol upload foto 2.Aplikasi menampilkan foto terbaru
<i>Alternative Flow 1</i>	1.Aktor menekan tombol choose file 2.Aktor tidak memasukkan foto 3.Aktor menekan tombol upload foto 2.Aplikasi menampilkan pesan eror "Please out this file"
<i>Post-condition</i>	Informasi dashboard dapat ditampilkan pada halaman dashboard

**Tabel 4.93 Use Case Scenario Mencari Kelas Reguler**

Tabel 4.94 Use Case Scenario Mencari Kelas Reguler merupakan table *use case scenario* untuk mencari kelas reguler.

Nomor Use case	APIKO-F-1-009, APIKO-F-2-014, APIKO-F-3-012
Nama	Mencari Kelas Reguler
Aktor	<i>Guest ,Member, Instruktur Olahraga</i>
Tujuan	Menyediakan layanan untuk mencari kelas reguler yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman View Reguler telah tersedia
<i>Main Flow</i>	1.Aktor memilih menu kelas 2.Aktor memilih reguler 2.Aplikasi melakukan pencarian terkait kelas reguler yang ditawarkan instruktur 3.Aplikasi menampilkan nama instruktur, jenis olahraga ,foto instruktur, sesi, harga, durasi, jumlah murid, rating
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	1.Menampilkan daftar instruktur olahraga yang membuat kelas reguler



**Tabel 4.95 Use Case Scenario Mencari Kelas Paket**

Tabel 4.96 *Use Case Scenario* Mencari Kelas Reguler merupakan table *use case scenario* untuk mencari kelas paket.

Nomor <i>Use case</i>	APIKO-F-1-010, APIKO-F-2-015, APIKO-F-3-013
Nama	Mencari Kelas Paket
Aktor	<i>Guest, Member, Instruktur Olahraga</i>
Tujuan	Menyediakan layanan untuk mencari kelas paket yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman View Paket telah tersedia
<i>Main Flow</i>	Tidak Ada
<i>Alternative Flow 1</i>	1.Aktor memilih menu kelas 2.Aktor memilih paket 3.Aplikasi melakukan pencarian terkait kelas paket yang ditawarkan instruktur namun tidak ada kelas paket yang ditawarkan instruktur 4.Aplikasi menampilkan “Tidak Ditemukan”
<i>Post-condition</i>	1.Menampilkan daftar instruktur olahraga yang membuat kelas paket

**Tabel 4.97 Use Case Scenario Mencari Berdasarkan Kriteria**

Tabel 4.98 *Use Case Scenario* Mencari Berdasarkan Kriteria merupakan table *use case scenario* untuk mencari berdasarkan kriteria.

Nomor <i>Use case</i>	APIKO-F-1-008, APIKO-F-2-002, APIKO-F-3-016
Nama	Mencari Berdasarkan Harga
Aktor	<i>Guest, Member, Instruktur Olahraga</i>
Tujuan	Menyediakan layanan untuk mencari berdasarkan harga yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman Pencarian telah tersedia
<i>Main Flow</i>	1.Aktor memilih tombol dropdown jenis olahraga, urutan, kelas 2.Aplikasi melakukan pencarian kriteria yang dimasukkan 3.Aplikasi menampilkan nama instruktur, jenis olahraga, foto instruktur, sesi, harga, durasi, jumlah murid, rating
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	1.Menampilkan daftar instruktur olahraga berdasarkan harganya

**Tabel 4.99 Use Case Scenario Melihat Daftar Pembayaran**

Tabel 4.100 *Use Case Scenario* Melihat Daftar Pembayaran merupakan table *use case scenario* untuk melihat daftar pembayaran.

Nomor <i>Use case</i>	APIKO-F-2-022
Nama	Melihat Daftar Pembayaran
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melihat daftar pembayaran
<i>Pre-condition</i>	Halaman Pembayaran telah tersedia
<i>Main Flow</i>	1.Aktor menekan menu Pembayaran 2.Aplikasi menampilkan daftar table detail pemesan, detail pemesanan, detail pembayaran, status pembayaran, bukti pembayaran dan konfirmasi pembayaran
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi daftar pembayaran berhasil ditampilkan pada halaman pembayaran

**Tabel 4.101 Use Case Scenario Melihat Detail Pemesanan**

Tabel 4.102 *Use Case Scenario* Melihat Detail Pemesanan merupakan table *use case scenario* untuk melihat detail pemesanan.

Nomor <i>Use case</i>	APIKO-F-2-017
Nama	Melihat Detail Pemesanan
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melihat detail pemesanan
<i>Pre-condition</i>	Halaman Pembayaran telah tersedia
<i>Main Flow</i>	1.Aktor menekan field lihat pada table detail pemesanan 2.Aplikasi menampilkan daftar pemesanan yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi jadwal, lokasi
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi detail pemesanan berhasil ditampilkan pada halaman pembayaran

**Tabel 4.103 Use Case Scenario Melihat Bukti Pembayaran**

Tabel 4.104 *Use Case Scenario* Melihat Bukti Pembayaran merupakan table *use case scenario* untuk melihat bukti pembayaran.

Nomor <i>Use case</i>	APIKO-F-2-018
Nama	Melihat Bukti Pembayaran

Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melihat bukti pembayaran
<i>Pre-condition</i>	Halaman Pembayaran telah tersedia
<i>Main Flow</i>	1.Aktor menekan field lihat pada table bukti pembayaran 2.Aplikasi menampilkan foto bukti pembayaran
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi bukti pembayaran berhasil ditampilkan pada halaman pembayaran

**Tabel 4.105 Use Case Scenario Melihat Detail Pemesanan Member**

Tabel 4.106 Use Case Scenario Melihat Detail Pemesanan Member merupakan table *use case scenario* untuk melihat detail pemesanan *member*.

Nomor Use case	APIKO-F-4-021
Nama	Melihat Detail Pemesanan Member
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat detail pemesanan member
<i>Pre-condition</i>	Halaman Order telah tersedia
<i>Main Flow</i>	1.Aktor menekan field lihat pada table detail pemesanan 2.Aplikasi menampilkan daftar pemesanan yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi jadwal, lokasi
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi detail pemesanan berhasil ditampilkan pada halaman order

**Tabel 4.107 Use Case Scenario Melihat Detail Pembayaran**

Tabel 4.108 Use Case Scenario Melihat Detail Pembayaran merupakan table *use case scenario* untuk melihat detail pembayaran.

Nomor Use case	APIKO-F-4-024
Nama	Melihat Detail Pembayaran
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat detail pembayaran
<i>Pre-condition</i>	Halaman Order telah tersedia
<i>Main Flow</i>	1.Aktor menekan field lihat pada table detail pembayaran 2.Aplikasi menampilkan foto bukti pembayaran, logo bank, total pembayaran, kode bayar
<i>Alternative Flow 1</i>	Tidak Ada

<i>Post-condition</i>	Informasi detail pembayaran berhasil ditampilkan pada halaman pembayaran
-----------------------	--

**Tabel 4.109 Use Case Scenario Melihat Detail Pemesan**

Tabel 4.110 *Use Case Scenario* Melihat Detail Pemesan merupakan table *use case scenario* untuk melihat detail pemesanan.

Nomor <i>Use case</i>	APIKO-F-4-023
Nama	Melihat Detail Pemesan
Aktor	<i>Admin</i>
Tujuan	Menyediakan layanan untuk melihat detail pemesanan
<i>Pre-condition</i>	Halaman Order telah tersedia
<i>Main Flow</i>	1. Aktor menekan field lihat pada table detail pemesanan 2. Aplikasi menampilkan nama pemesanan dan no telepon
<i>Alternative Flow 1</i>	Tidak Ada
<i>Post-condition</i>	Informasi detail pemesanan berhasil ditampilkan pada halaman pembayaran

**Tabel 4.111 Use Case Scenario Melakukan Pemesanan Kelas Paket**

Tabel 4.112 *Use Case Scenario* Melakukan Pemesanan Kelas Paket merupakan table *use case scenario* untuk pemesanan paket.

Nomor <i>Use case</i>	APIKO-F-2-019
Nama	Melakukan Pemesanan Kelas Paket
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melakukan pemesanan kelas paket yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman pemesanan-paket telah tersedia
<i>Main Flow</i>	1. Aktor menekan tombol pesan instruktur 2. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga, jumlah murid 3. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon 4. Aktor memilih hari, jam dan lokasi latihan 5. Aktor menekan tombol pesan 6. Aplikasi menyimpan data pemesanan
<i>Alternative Flow 1</i>	1. Aktor tidak mengisi seluruh field pada form pemesanan 2. Aktor menekan tombol pesan 3. Aplikasi gagal menyimpan data pemesanan dan menampilkan pesan "please fill out this file"

<i>Alternative Flow 2</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol pesan instruktur</li> <li>2. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga, jam</li> <li>3. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon</li> <li>4. Aktor memilih hari, jam dan lokasi latihan</li> <li>5. Aktor mengisi paket hari yang sama untuk satu minggu pada paket 8/12 sesi latihan</li> <li>6. Aktor menekan tombol pesan</li> <li>7. Aplikasi menampilkan pesan "Error ! Anda memasukkan hari yang sama. Silahkan masukkan hari yang berbeda"</li> </ol>
<i>Alternative Flow 3</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol pesan instruktur</li> <li>2. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga, jam</li> <li>3. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon</li> <li>4. Aktor memilih hari, jam dan lokasi latihan</li> <li>5. Aktor mengisi jadwal yang sudah dipesan member lain</li> <li>6. Aktor menekan tombol pesan</li> <li>7. Aplikasi menampilkan pesan "Jadwal penuh pada nama hari, dd-mm-yyyy pukul X"</li> </ol>
<i>Post-condition</i>	Data pemesanan kelas paket berhasil disimpan pada database

**Tabel 4.61 Use Case Scenario Melakukan Pemesanan Kelas Reguler**

Tabel 4.61 Use Case Scenario Melakukan Pemesanan Kelas Reguler merupakan table *use case scenario* untuk pemesanan reguler.

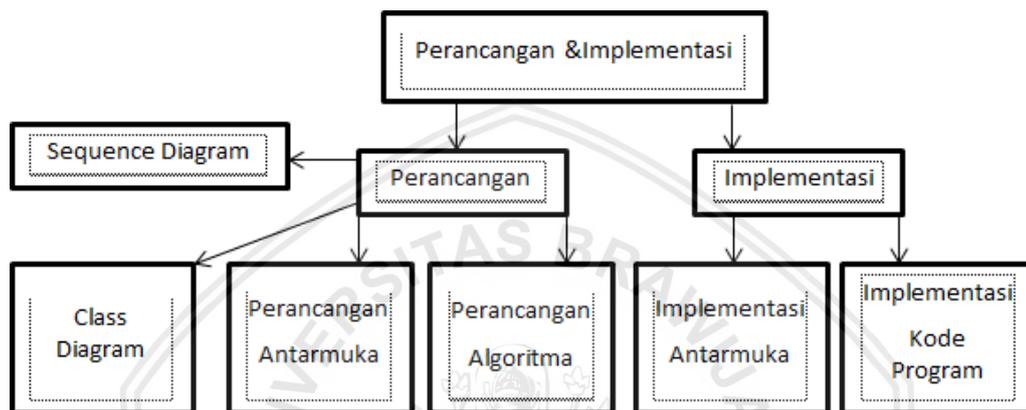
Nomor Use case	APIKO-F-2-020
Nama	Melakukan Pemesanan Kelas Reguler
Aktor	<i>Member</i>
Tujuan	Menyediakan layanan untuk melakukan pemesanan kelas reguler yang ditawarkan instruktur olahraga
<i>Pre-condition</i>	Halaman pemesanan-reguler telah tersedia
<i>Main Flow</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol pesan instruktur</li> <li>2. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga, jumlah murid</li> <li>3. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon</li> <li>4. Aktor memilih jadwal latihan dan lokasi latihan</li> <li>5. Aktor menekan tombol pesan</li> <li>6. Aplikasi menyimpan data pemesanan</li> </ol>

<i>Alternative Flow 1</i>	<ol style="list-style-type: none"> <li>1. Aktor tidak mengisi seluruh field pada form pemesanan</li> <li>2. Aktor menekan tombol pesan</li> <li>3. Aplikasi gagal menyimpan data pemesanan dan menampilkan pesan "please fill out this file "</li> </ol>
<i>Alternative Flow 2</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol pesan instruktur</li> <li>2. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga</li> <li>3. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon</li> <li>4. Aktor memilih jadwal latihan dan lokasi latihan</li> <li>5. Aktor mrmilih jadwal latihan yang sudah dipesan member lain</li> <li>6. 7. Aplikasi menampilkan pesan "Tanggal yyyy-mm-dd Jam X penuh"</li> </ol>
<i>Post-condition</i>	Data pemesanan kelas reguler berhasil disimpan pada database



## BAB 5 PERANCANGAN DAN IMPLEMENTASI

Pada bab ini menjelaskan mengenai 2 tahap yaitu: perancangan dan *implementasi*. Perancangan sistem menjadi sebuah dasar untuk *implementasi*. Perancangan sistem dibuat setelah melakukan tahap analisis dan pemodelan kebutuhan. Sedangkan pada implementasi pada sistem dilakukan untuk menerapkan hasil rancangan untuk membangun sistem. Struktur diagram perancangan dan *implementasi* dapat dilihat pada Gambar 5.1 Struktur Diagram Perancangan dan Implementasi dibawah ini.



Gambar 5.1 Struktur Diagram Perancangan dan *Implementasi*

### 5.1 Perancangan

Perancangan dapat dilakukan setelah selesai merekayasa kebutuhan dan memodelkan kebutuhannya menjadi *use case diagram* dan *use case scenario*. Dalam hal ini *use case diagram* dan *use case scenario* yang akan menjadi dasar dari rancangan sistem. Proses perancangan aplikasi pembelajaran pemrograman java ini dibagi ke dalam 3 tahap yakni perancangan *sequence diagram*, *class diagram*, perancangan *algoritma* dan perancangan *interface*.

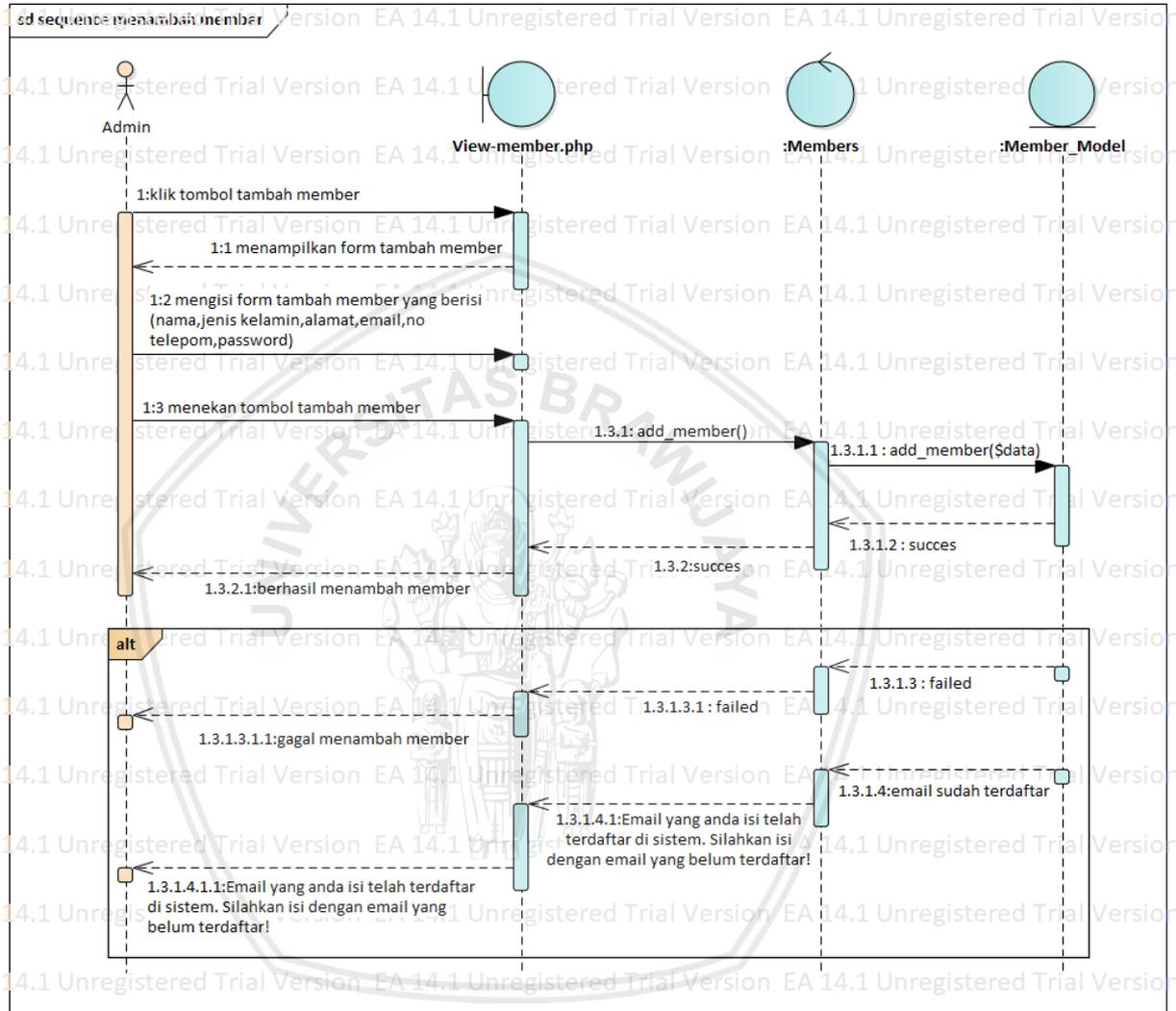
#### 5.1.1 Perancangan *Sequence Diagram*

*Sequence diagram* digunakan untuk mengetahui alur sebuah sistem dari tiap fungsionalitasnya dimana akan terdapat objek yang berperan didalamnya. Pembuatan *sequence diagram* didasarkan pada *use case scenario* tiap fungsionalitas.

##### 5.1.1.1 *Sequence Diagram* Membuat *Member*

Pada Gambar 5.2 *Sequence Diagram* Menambahkan *Member* berikut, proses membuat member dimulai dari admin menekan tombol tambah member pada halaman view-member kemudian halaman view-member menampilkan form tambah member, lalu admin mengisi form tambah member yang berisi : nama, jenis kelamin, alamat, email, no telepon dan password, kemudian admin menekan tombol tambah

member yang mana saat tombol tambah member ditekan maka kelas view-member memanggil method `add_member()` dari kelas `Members`, kemudian kelas `Members` akan memanggil method `addmember($data)` dengan parameter data pada kelas `Member_Model`. Jika proses berhasil maka member berhasil dibuat. Jika proses gagal member gagal dibuat.

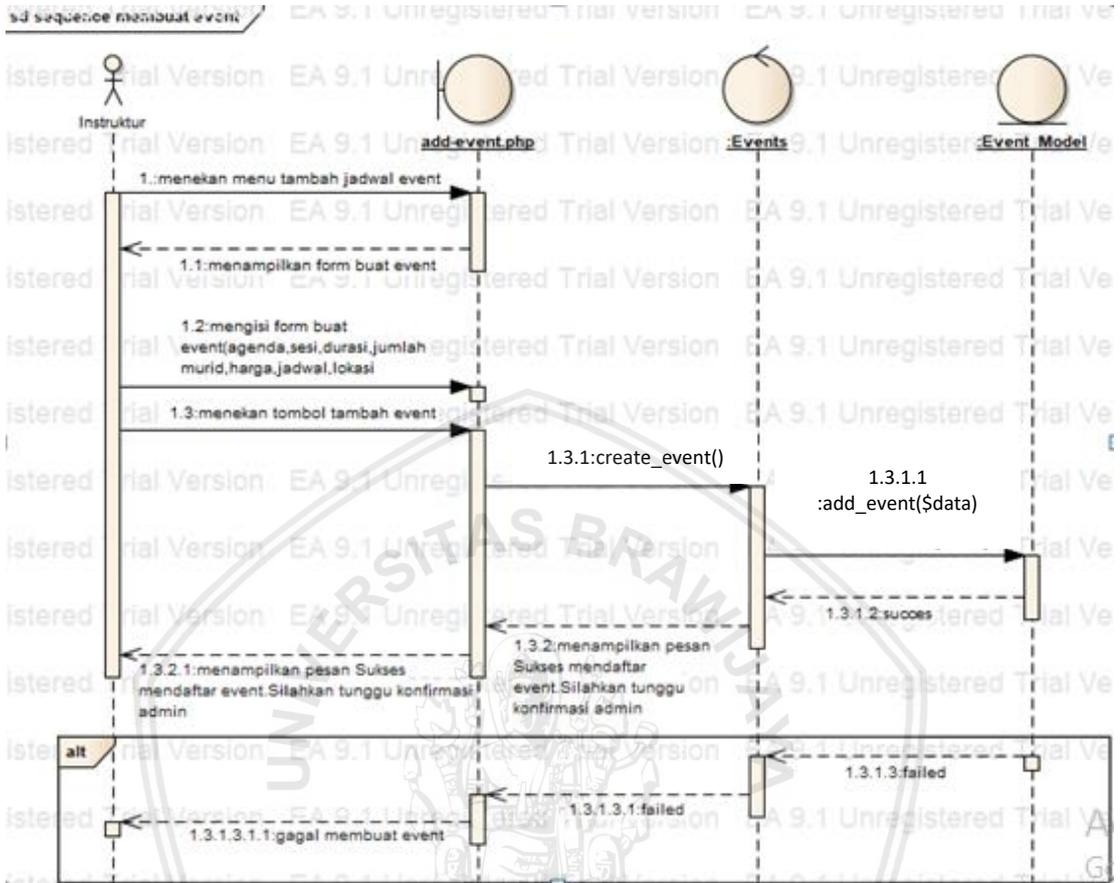


Gambar 5.2 Sequence Diagram Menambahkan Member

### 5.1.1.2 Sequence Diagram Membuat Kelas Event

Pada Gambar 5.3 Sequence Diagram Membuat Kelas Event berikut proses membuat event dimulai saat instruktur menekan menu tambah jadwal event pada halaman view-event kemudian halaman view-event menampilkan form buat event. Setelah instruktur mengisi form buat event yang berisi agenda ,sesi, durasi, jumlah murid, harga ,jadwal dan lokasi lalu menekan tombol tambah event. Pada saat menekan tombol tambah event kelas view-event akan memanggil method `create_event()` pada kelas `Events` yang kemudian Kelas `Events` memanggil method `addEvent($data)` dengan parameter data. Pada kelas `Event_Model`. Jika proses sukses

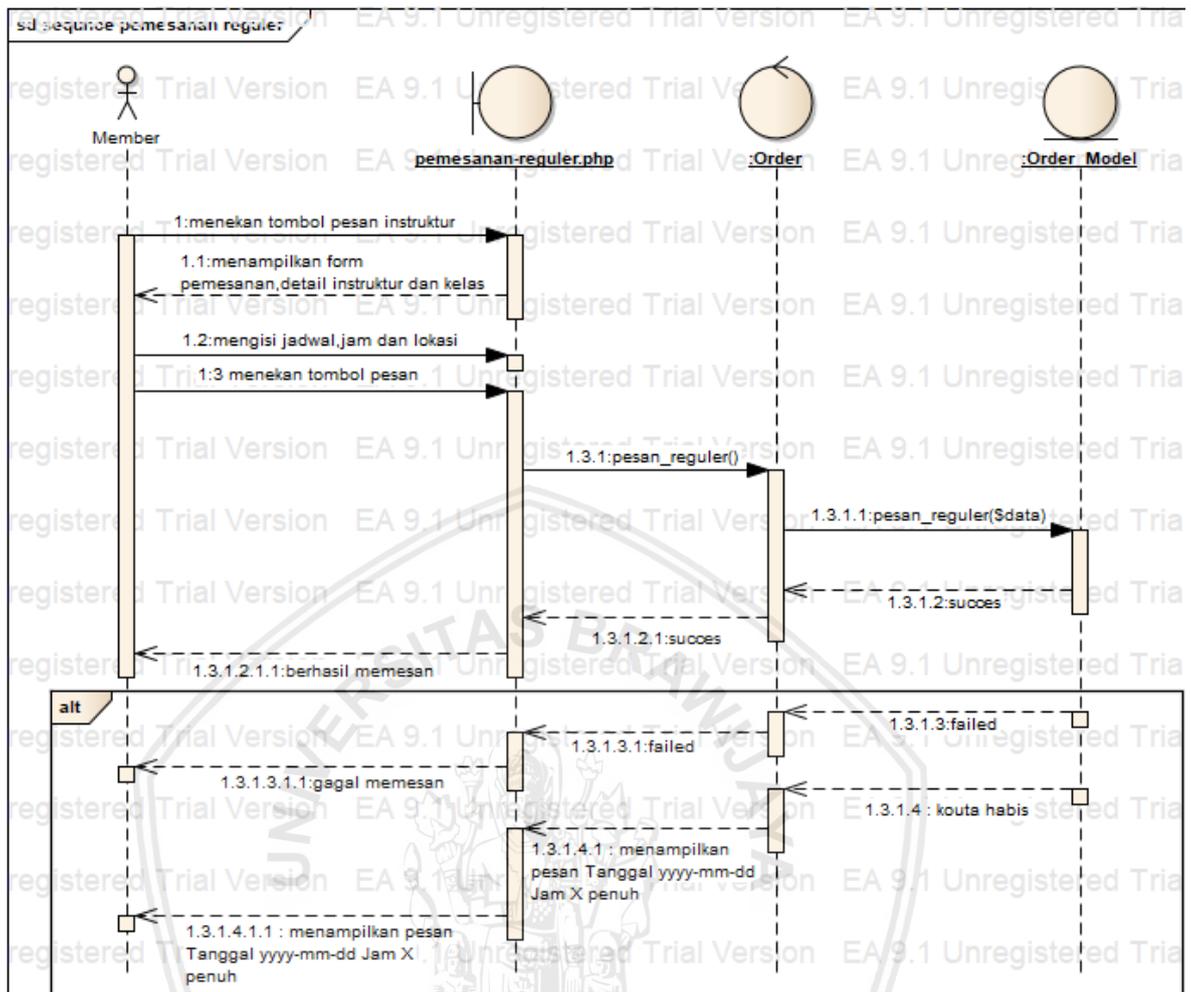
maka pembuatan event berhasil dibuat. Jika proses gagal maka pembuatan event gagal dibuat.



Gambar 5.3 Sequence Diagram Membuat Kelas Event

### 5.1.1.3 Sequence Diagram Melakukan Pemesanan Kelas Reguler

Pada Gambar 5.4 Sequence Diagram Melakukan Pemesanan Kelas Reguler berikut, proses melakukan pemesanan dimulai saat member menekan tombol pesan instruktur pada kelas view-pemesanan kemudian kelas view-pemesanan menampilkan form pemesanan yang berisi nama, no telepon kemudian mengisi jadwal, jam dan lokasi latihan lalu menekan tombol pesan. Pada saat tombol pesan ditekan kelas view-pemesanan memanggil method pesan\_reguler() pada kelas Pemesanans, lalu kemudian Kelas Order memanggil method pesan\_reguler(\$data) dengan parameter data pada kelas Order\_Model. Jika proses sukses maka pemesanan berhasil dibuat. Jika proses gagal maka pemesanan gagal dibuat.



Gambar 5.4 Sequence Diagram Melakukan Pemesanan Kelas Reguler

### 5.1.2 Perancangan Class Diagram

Class diagram akan menggambarkan hubungan antar class dengan objek tersebut. Class diagram dari aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web ditunjukkan pada Gambar 5.5 Class Diagram dibawah ini. Pada Gambar 5.5 Class Diagram ini menjelaskan hubungan class yang mengextends dari class CI\_Controller yaitu: Members, Pemesans, Events, Regulars , Pakets, Admin, Instruktur, Order, Dashboard, Home, Events, Member, Instruktur yang mana didalam kelasnya memiliki method dan atribut yang berbeda-beda. Hubungan class lain yang mengextends dari class CI\_Model yaitu: Member\_Model, Order\_Model, Event\_Model, Reguler\_Model, Paket\_Model, Instruktur\_Model, Administrator\_Model yang mana didalamnya memiliki method dan atribut yang berbeda. Antara class yang mengextends CI\_Controller dan CI\_Model memiliki hubungan dimana hubungan antar class tersebut diantaranya memiliki method yang sama pada masing-masing class, namun dengan fungsi method yang berbeda.



### 5.1.3 Perancangan *Algoritma*

Perancangan *algoritma* berisi *method-method* dari *algoritma* yang menyusun sistem. Perancangan *algoritma* digunakan untuk mempermudah pembuatan *source code* sehingga memberikan alur logika yang tepat

#### 5.1.3.1 Perancangan *Algoritma* Konfirmasi Pembayaran

Rancangan *algoritma* dapat dilihat pada Tabel 5.1 *Algoritma* Konfirmasi Pembayaran dibawah ini.

Nama Kelas : Order

Nama Method : konfirmasi\_pembayaran()

**Tabel 5.2 *Algoritma* Konfirmasi Pembayaran**

1	START
2	Initiate konfigurasi resource image
3	Initiate konfigurasi type image png or jpg
4	Initiate konfigurasi max size 3000
5	Initiate konfigurasi max width 5000
6	Initiate konfigurasi max height 5000
7	Initiate konfigurasi encrypt name true
8	Set data image
9	If set data image false
10	Set data fail
11	Show alert "pelase select a file"
12	Else set data image true
13	Get array data
14	Call compile method konfirmasi pembayaran on order model
15	Set data succes
16	Show alert "anda berhasil mengonfirmasi pembayaran.
17	Silahkan tunggu pembayaran anda dikonfirmasi oleh admin"
17	END

#### 5.1.3.2 Perancangan *Algoritma* Mengkonfirmasi Pendaftaran Instruktur

Rancangan *algoritma* dapat dilihat pada Tabel 5.3 *Algoritma* Mengkonfirmasi Pendaftaran Instruktur dibawah ini.

Nama Kelas : Instruktors

Nama Method : confirm\_instruktur(id)

**Tabel 5.4 *Algoritma* Mengkonfirmasi Pendaftaran Instruktur**

1	STRAT
2	Get data array
3	Id = id
4	Verified = 1
5	Call compile method get detail instruktur id on instruktur model
6	Call compile method edit instruktur data on instruktur model
7	Set configuration email
8	Call library email



9	Set email respons
10	If email send sukses
11	Notification success
12	Show message "Hi 'X' , Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima, Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima, Silahkan login menggunakan email dan password yang sudah terdaftar untuk melakukan perubahan profil, pembuatan kelas dan event"
13	Else email send fail
14	Notification failed
15	END

### 5.1.3.3 Perancangan *Algoritma* Melakukan Pemesanan Reguler

Rancangan *algoritma* dapat dilihat pada Tabel 5.5 *Algoritma* Melakukan Pemesanan Reguler pada dibawah ini.

Nama Kelas : Order

Nama Method : pesen\_reguler()

**Tabel 5.6 *Algoritma* Melakukan Pemesanan Reguler**

1	STRAT
2	i=inputan
3	variable cek = array
4	Inisialisasi id instruktur get form field
5	Id-instruktur
6	Inisialisasi jadwal get from field jadwal
8	Inisialisasi jam get from field jam
9	Inisialisasi variabel num get form method cek-order-reguler in class Order-Model
10	If variable num > 0
11	Set data fail
12	Show message "Tanggal-jadwal-jam- Penuh"
13	Else
14	Get data array
15	Inisialisasi id-pemesanan get form field method pesan- reguler parameter data in method Order-Model
16	Set data success
17	Show message "Sukses memesan regular"
18	END

### 5.1.3.4 Perancangan *Algoritma* Menambahkan Member

Rancangan *algoritma* dapat dilihat pada Tabel 5.7 *Algoritma* Menambahkan Member dibawah ini.

Nama Kelas : Members

Nama Method : add\_member()



**Tabel 5.8 Algoritma Menambahkan Member**

1	START
2	Get form validation
3	Set data nama, jenis kelamin, alamat, email, no telepon, password
4	If set data false
5	Call compile method get all member on member model
6	Set data fail
7	Show alert "lengkapi data"
8	Else set data email true
9	Inisialisasi variable num call compile method check email with parameter where on Instruktur Model
10	Inisialisasi variable num2 call compile method check email with parameter where on Member Model
11	If variable num > 0    variable num2 > 0
12	Show alert "mail yang anda isi telah terdaftar di sistem. Silahkan isi dengan email yang belum terdaftar!"
13	Else
14	Set data array nama, jenis kelamin, password, alamat, email, no telepon
15	Show alert "Berhasil menambah member"
16	END

**5.1.3.5 Perancangan Algoritma Membuat Kelas Event**

Rancangan *algoritma* dapat dilihat pada Tabel 5.9 *Algoritma Membuat Kelas Event* dibawah ini.

Nama Kelas : Events

Nama Method : create\_event()

**Tabel 5.10 Algoritma Membuat Kelas Event**

1	START
2	Get form validation
3	Set data nama event, nama instruktur, sesi, durasi, jumlah murid, harga
4	If set data false
5	Call compile method get all event on event model
6	Set data fail
7	Show message "lengkapi data"
8	Else set data true
9	Get array data
10	Call compile method add event on event model
11	Set data succes
12	Show message "sukses mendaftar event"
13	END



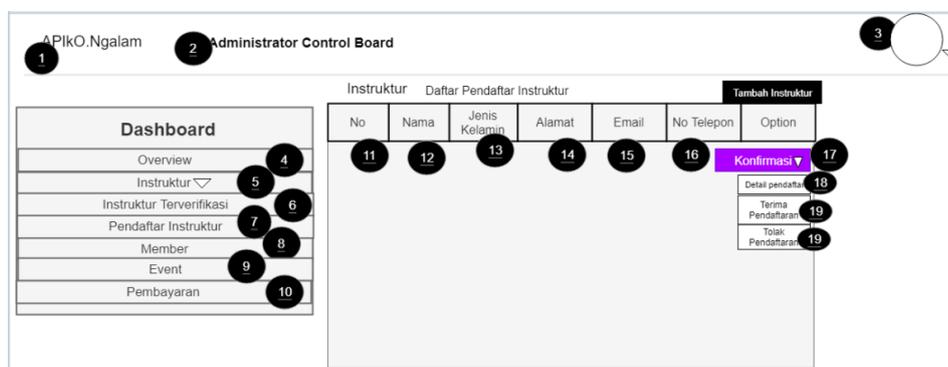
#### 5.1.4 Perancangan *Interface*

Perancangan *interface* memodelkan dalam bentuk *low level design* yang berisi halaman-halaman *interface* yang menyusun aplikasi. Seluruh perancangan *interface* setiap halaman pada aplikasi ini menggunakan konsep *Usability Heuristic* diantaranya :

1. Memiliki tampilan warna yang natural dan tidak berlebihan, memberi warna menonjol pada obyek seperti tombol pada aplikasi agar pengguna tidak ragu dalam menggunakan sebuah perintah (*Efisiensi*).
2. Memiliki desain yang simple dan menampilkan perintah-perintah penting saja yang ditampilkan dengan bahasa natural seperti notifikasi alert pada aplikasi sehingga aplikasi mudah dipelajari (*Learnabilitas*).
3. Memiliki menu yang disajikan secara konsisten sehingga tidak membebani daya ingatan user terhadap aplikasi (*Memorabilitas*).
4. Memiliki menu help pada halaman pemesanan paket untuk menghindari kesalahan pengguna dalam melakukan pemesanan paket (*Kesalahan/eror*).
5. Memiliki ukuran layout yang konsisten sehingga ketika dibrowser seperti *Mozilla*, *Chrome*, dll tidak merubah ukuran dan desain tampilannya tetap stabil (*Efisiensi*).
6. Memiliki tampilan halaman pencarian dengan berdasarkan banyak kategori sehingga sistem dapat digunakan dengan nyaman (*Satisfaction*).
7. Memiliki tampilan yang menampilkan jalan keluar/logout dari sistem yang jelas, memiliki tampilan untuk halaman mengedit profil dan mengganti foto profil, serta memiliki tampilan yang mudah dipelajari saat memilih jadwal dan jam pada seluruh halaman buat jadwal kategori semua kelas (*Useful*).

##### 5.1.4.1 Perancangan *Interface* Halaman Lihat Daftar Pendaftar Instruktur

Perancangan *interface* dapat dilihat pada Gambar 5.6 Perancangan *Interface* Halaman Lihat Daftar Pendaftar Instruktur dan untuk dekripsinya dapat dilihat pada Tabel 5.11 Keterangan Perancangan *Interface* Halaman Daftar Pendaftar Instruktur.



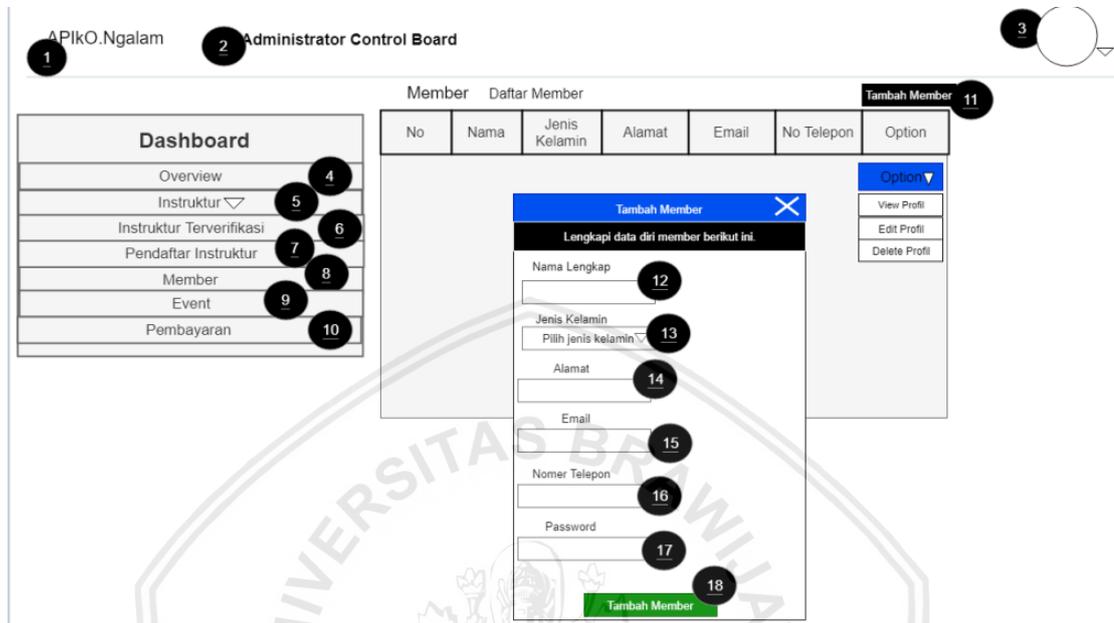
Gambar 5.7 Perancangan *Interface* Halaman Lihat Daftar Pendaftar Instruktur

**Tabel 5.12 Keterangan Perancangan *Interface* Halaman Daftar Pendaftar Instruktur**

No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tabel no berfungsi menampilkan nomer urutan daftar instruktur pendaftar
12	Tabel nama berfungsi menampilkan nama daftar pendaftar instruktur
13	Label jenis kelamin berfungsi menampilkan jenis kelamin daftar pendaftar instruktur
14	Tabel alamat berfungsi menampilkan alamat daftar pendaftar instruktur
15	Tabel email berfungsi menampilkan email daftar pendaftar instruktur
16	Tabel no telepon berfungsi menampilkan no telepon daftar pendaftar instruktur
17	Tombol konfirmasi untuk menampilkan drop down
18	Tombol detail pendaftar untuk menampilkan detail pendaftar
19	Tombol terima pendaftaran untuk menerima pendaftaran
20	Tombol tolak pendaftaran untuk menolak pendaftaran

### 5.1.4.2 Perancangan Interface Halaman Membuat Member

Perancangan *interface* dapat dilihat pada Gambar 5.8 Perancangan *Interface* Halaman Membuat Member dan untuk deskripsinya pada Tabel 5.13 Keterangan Perancangan *Interface* Halaman Membuat *Member*.



**Gambar 5.9 Perancangan *Interface* Halaman Membuat *Member***

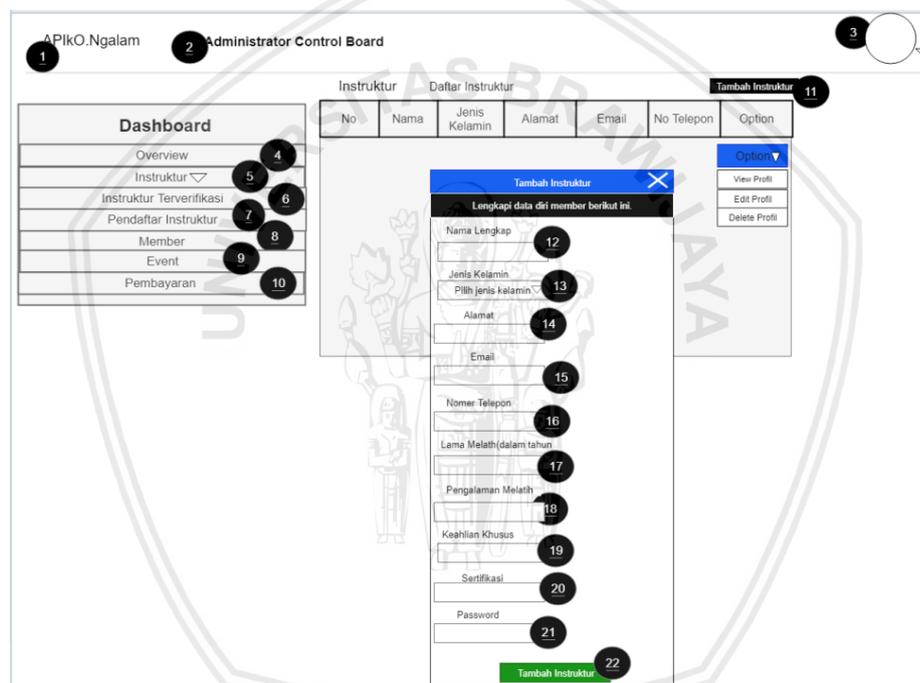
**Tabel 5.14 Keterangan Perancangan *Interface* Halaman Membuat *Member***

No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tombol tambah member berfungsi untuk menampilkan form tambah member
12	Field untuk isian nama

13	Tombol dropdown untuk memilih jenis kelamin
14	Field untuk isian alamat
15	Field untuk isian email
16	Field untuk isian nomer telepon
17	Field untuk isian password
18	Tombol tambah member berfungsi untuk menambah member

### 5.1.4.3 Perancangan *Interface* Halaman Membuat Instruktur

Perancangan *interface* dapat dilihat pada Gambar 5.10 Perancangan *Interface* Halaman Membuat Instruktur dan untuk deskripsinya pada Tabel 5.15 Keterangan Perancangan *Interface* Halaman Membuat Instruktur.



Gambar 5.11 Perancangan *Interface* Halaman Membuat Instruktur

Tabel 5.16 Keterangan Perancangan *Interface* Halaman Membuat Instruktur

No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi



7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tombol tambah instruktur berfungsi untuk menampilkan form tambah instruktur
12	Field untuk isian nama
13	Tombol dropdown untuk memilih jenis kelamin
14	Field untuk isian alamat
15	Field untuk isian email
16	Field untuk isian nomer telepon
17	Field untuk isian lama melatih
18	Field untuk isian pengalaman melatih
19	Field untuk isian keahlian khusus
20	Field untuk isian sertifikasi
21	Field untuk isian password
22	Tombol tambah instruktur berfungsi untuk menambah instruktur

#### 5.1.4.4 Perancangan *Interface* Halaman Buat Kelas Event

Perancangan *interface* dapat dilihat pada Gambar 5.12 Perancangan *Interface* Halaman Buat Kelas Event dan untuk deskripsinya pada Tabel 5.17 Keterangan Perancangan *Interface* Halaman Buat Kelas Event.



**Gambar 5.13 Perancangan Interface Halaman Buat Kelas Event**

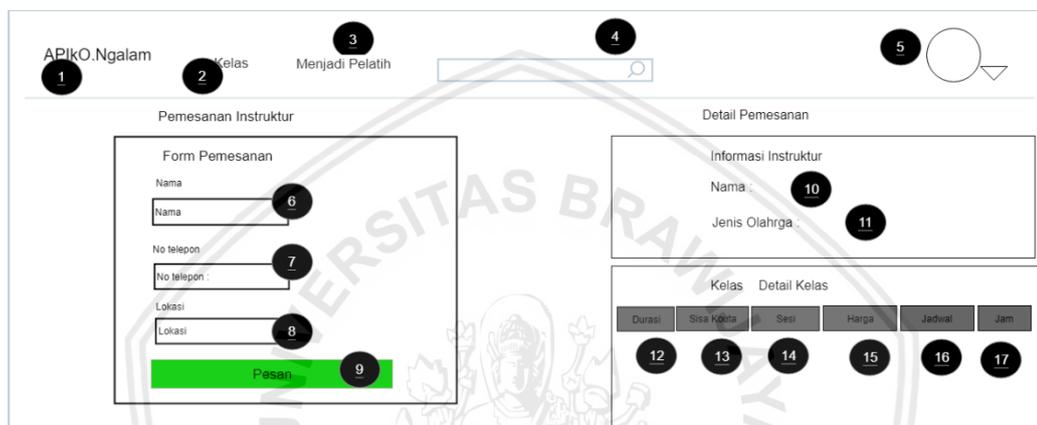
**Tabel 5.18 Keterangan Perancangan Interface Halaman Buat Kelas Event**

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
7	Label profil untuk melihat profil
8	Label Lihat Jadwal untuk menampilkan jadwal
9	Label tambah jadwal event untuk menampilkan form tambah jadwal event
10	Label tambah jadwal regular untuk menampilkan form tambah jadwal regular
11	Label tambah jadwal group untuk menampilkan form tambah jadwal paket
12	Label hapus akun berfungsi menghapus akun instruktur
13	Field untuk isian jumlah sesi
14	Field untuk isian jumlah murid
15	Field untuk isian durasi
16	Field untuk isian harga
17	Field untuk isian jadwal

18	Field untuk isian jam
19	Field untuk isian tempat
20	Field untuk isian agenda
21	Tombol tambah event berfungsi untuk menambah kelas event

#### 5.1.4.5 Perancangan *Interface* Membuat Halaman Pemesanan Kelas Event

Perancangan *interface* dapat dilihat pada Gambar 5.14 Perancangan *Interface* Halaman Pemesanan Kelas Event dan deskripsinya pada Tabel 5.19 Keterangan Perancangan *Interface* Halaman Pemesanan Kelas Event.



Gambar 5.15 Perancangan *Interface* Halaman Pemesanan Kelas Event

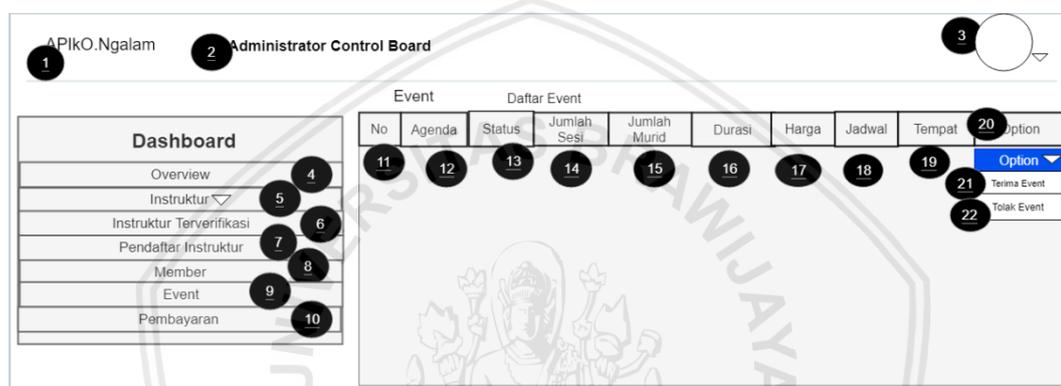
Tabel 5.20 Keterangan Perancangan *Interface* Halaman Pemesanan Kelas Event

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Field untuk isian nama
7	Field untuk isian no telepon
8	Field untuk isian lokasi
9	Tombol pesan untuk melakukan pemesanan
10	Tabel untuk menampilkan nama instruktur
11	Tabel untuk menampilkan jenis olahraga
12	Tabel untuk menampilkan durasi

13	Tabel untuk menampilkan jumlah murid
14	Tabel untuk menampilkan jumlah sesi
15	Tabel untuk menampilkan harga
16	Tabel untuk menampilkan jadwal
17	Tabel untuk menampilkan jam

#### 5.1.4.6 Perancangan *Interface* Halaman Lihat Event

Perancangan *interface* dapat dilihat pada Gambar 5.16 Perancangan *Interface* Halaman Lihat Event dan deskripsinya pada Tabel 5.21 Keterangan Perancangan *Interface* Halaman Lihat Event.



Gambar 5.17 Perancangan *Interface* Halaman Lihat Event

Tabel 5.22 Keterangan Perancangan *Interface* Halaman Lihat Event

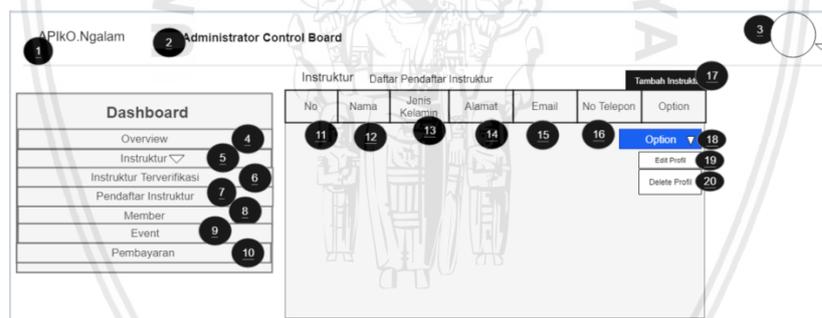
No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tabel no berfungsi menampilkan nomer urutan daftar event



12	Tabel agenda berfungsi menampilkan agenda daftar event
13	Tabel status berfungsi menampilkan status daftar event
14	Tabel jumlah sesi berfungsi menampilkan jumlah sesi dalam event
15	Tabel jumlah murid berfungsi menampilkan jumlah murid dalam event
16	Tabel durasi berfungsi menampilkan durasi dalam event
17	Tabel harga berfungsi menampilkan harga dalam event
18	Tabel jadwal berfungsi menampilkan jadwal dalam event
19	Tabel tempat berfungsi menampilkan tempat dalam event
20	Tabel option berfungsi menampilkan tombol option untuk pilihan dropdown
21	Tombol terima event untuk menerima event
22	Tombol tolak event untuk menolak event

**5.1.4.7 Perancangan *Interface* Halaman Lihat Instruktur Terverifikasi**

Perancangan *interface* dapat dilihat pada Gambar 5.18 Perancangan *Interface* Halaman Lihat Instruktur Terverifikasi dan deskripsinya pada Tabel 5.23 Keterangan Perancangan *Interface* Halaman Lihat Instruktur Terverifikasi.



**Gambar 5.19 Perancangan *Interface* Halaman Lihat Instruktur Terverifikasi**

**Tabel 5.24 Keterangan Perancangan *Interface* Halaman Lihat Instruktur Terverifikasi**

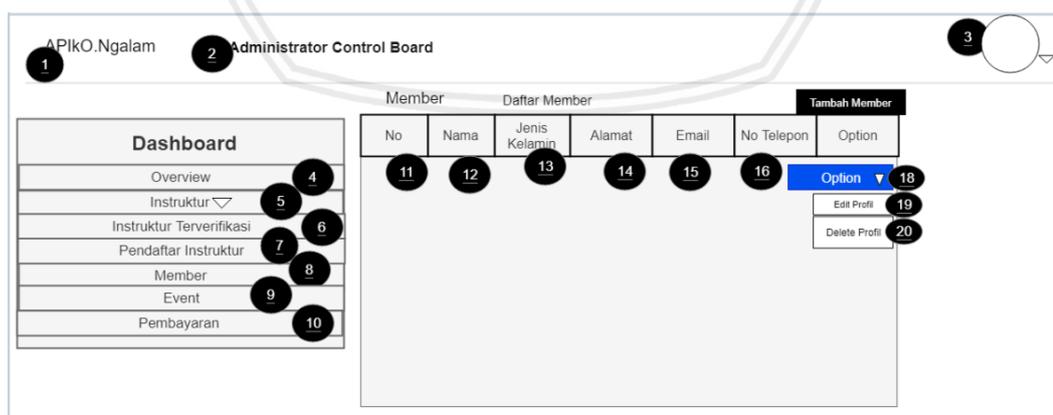
No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi



7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tabel no berfungsi menampilkan nomer urutan daftar instruktur yang terverifikasi
12	Tabel nama berfungsi menampilkan nama daftar instruktur yang terverifikasi
13	Tabel jenis kelamin berfungsi menampilkan jenis kelamin daftar instruktur yang terverifikasi
14	Tabel alamat berfungsi menampilkan alamat daftar instruktur yang terverifikasi
15	Tabel email berfungsi menampilkan email daftar instruktur yang terverifikasi
16	Tabel no telepon berfungsi menampilkan no telepon daftar yang terverifikasi
17	Tombol tambah instruktur untuk menambah instruktur
18	Tombol option untuk menampilkan drop down
19	Tombol edit profil untuk mengedit profil instruktur
20	Tombol delete profil untuk menghapus instruktur

#### 5.1.4.8 Perancangan *Interface* Halaman Lihat Member

Perancangan *interface* dapat dilihat pada Gambar 5.20 Perancangan *Interface* Halaman Lihat Member dan deskripsinya pada Tabel 5.25 Keterangan Perancangan *Interface* Halaman Lihat Member.



Gambar 5.21 Perancangan *Interface* Halaman Lihat Member

Tabel 5.26 Keterangan Perancangan *Interface* Halaman Lihat Member

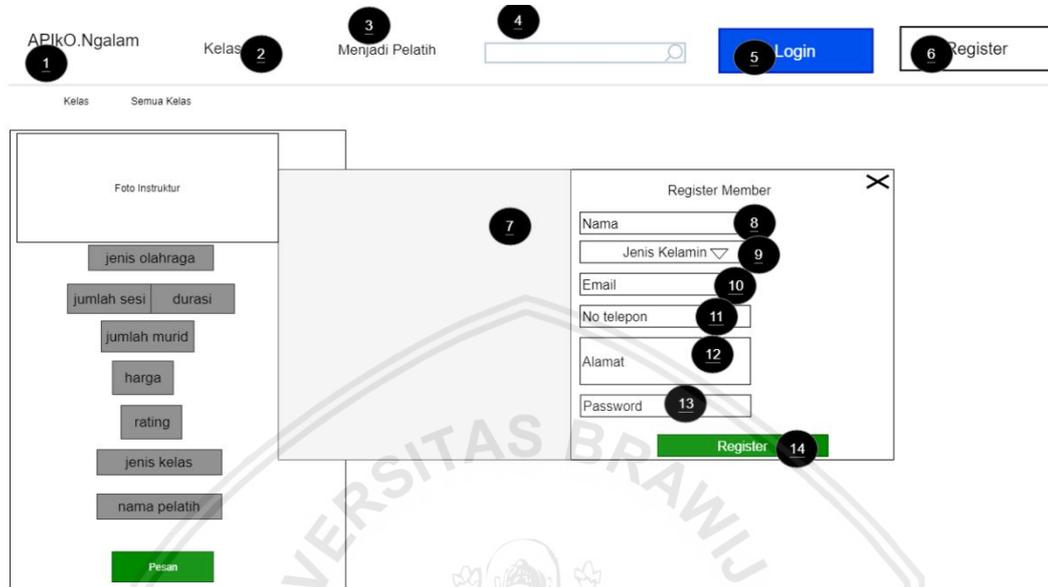
No	Deskripsi
----	-----------



1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tabel no berfungsi menampilkan nomer urutan daftar member
12	Tabel nama berfungsi menampilkan nama daftar member
13	Label jenis kelamin berfungsi menampilkan jenis kelamin daftar member
14	Tabel alamat berfungsi menampilkan alamat daftar member
15	Tabel email berfungsi menampilkan email daftar member
16	Tabel no telepon berfungsi menampilkan no telepon daftar member
17	Tombol tambah member untuk menambah member
18	Tombol option untuk menampilkan drop down
19	Tombol edit profil untuk mengedit profil member
20	Tombol delete profil untuk menghapus member

### 5.1.4.9 Perancangan *Interface* Halaman *Registrasi Member*

Perancangan *interface* dapat dilihat pada Gambar 5.22 Perancangan *Interface* Halaman *Register Member* dan deskripsinya pada Tabel 5.27 Keterangan Perancangan *Interface* Halaman *Register Member*.



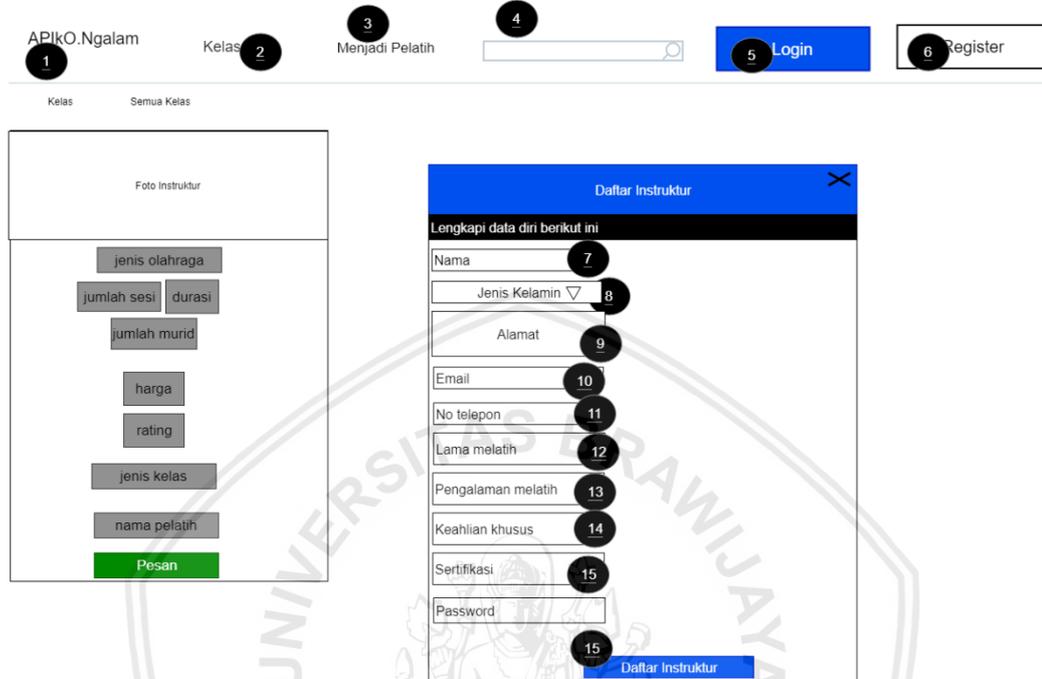
Gambar 5.23 Perancangan *Interface* Halaman *Register Member*

Tabel 5.28 Keterangan Perancangan *Interface* Halaman *Register Member*

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Login untuk menampilkan form login
6	Label register untuk menampilkan form register member
7	Field unuk gambar
8	Field untuk isian nama
9	Dropdown untuk memilih jenis kelamin
10	Field untuk isian email
11	Field untuk isian no telepon
12	Field untuk isian no email
13	Field untuk isian password
14	Tombol registrasi untuk memproses proses regitrasi member

### 5.1.4.10 Perancangan *Interface* Halaman Registrasi Instruktur

Perancangan *interface* dapat dilihat pada Gambar 5.24 Perancangan *Interface* Halaman *Register* Instruktur dan deskripsinya pada Tabel 5.29 Keterangan Perancangan *Interface* Halaman *Register* Instruktur.



Gambar 5.25 Perancangan *Interface* Halaman *Register* Instruktur

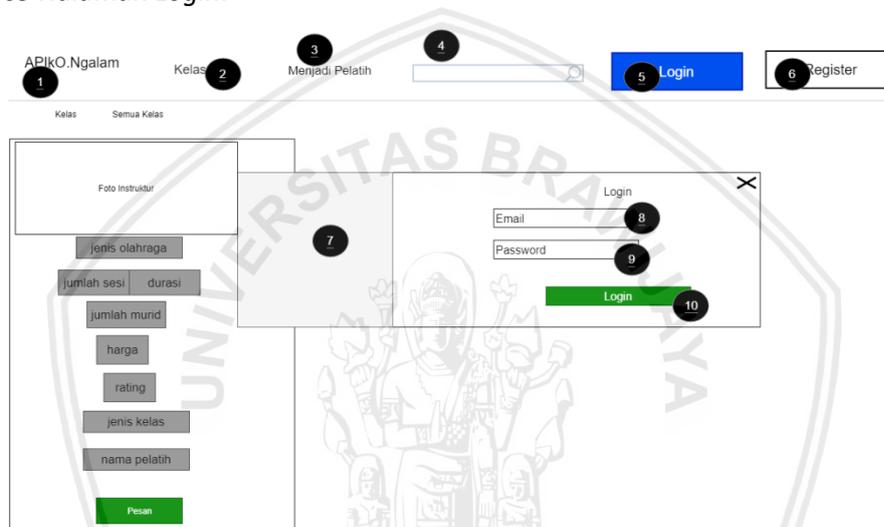
Tabel 5.30 Keterangan Perancangan *Interface* Halaman *Register* Instruktur

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Login untuk menampilkan form login
6	Label register untuk menampilkan form register member
7	Field untuk isian nama
8	Dropdown untuk memilih jenis kelamin
9	Field untuk isian alamat
10	Field untuk isian email
11	Field untuk isian no telepon
12	Field untuk isian lama melatih

13	Field untuk isian pengalaman melatih
14	Field untuk isian keahlian khusus
15	Field unuk isian sertifikasi
16	Field untuk isian password
17	Tombol registrasi untuk memproses proses regitras instruktur

#### 5.1.4.11 Perancangan *Interface* Halaman Login Member dan Instruktur

Perancangan *interface* dapat dilihat pada Gambar 5.26 Perancangan *Interface* Halaman Login dan deskripsinya pada Tabel 5.31 Keterangan Perancangan *Interface* Halaman Login.



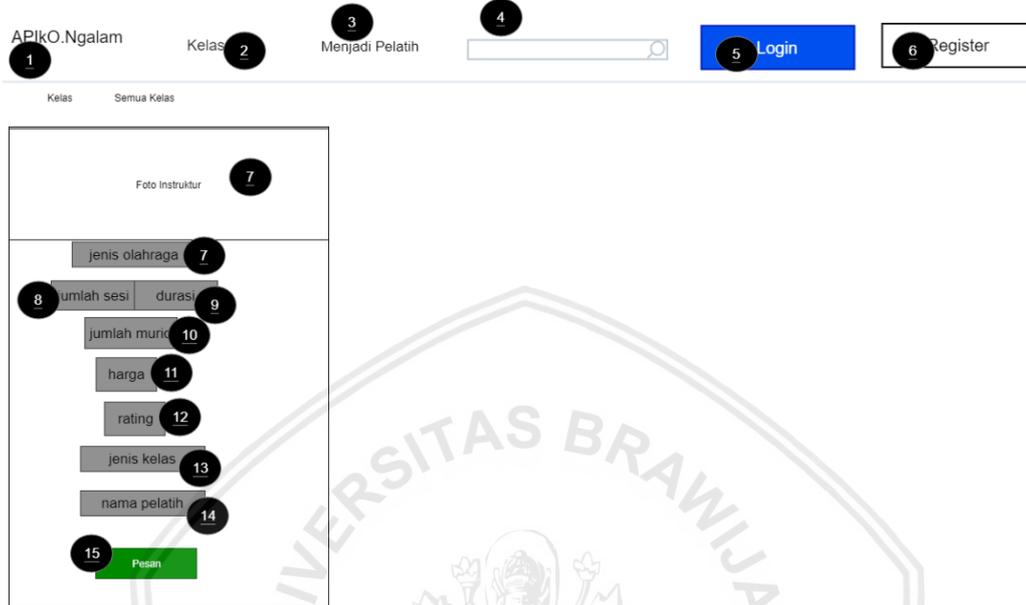
Gambar 5.27 Perancangan *Interface* Halaman Login

Tabel 5.32 Keterangan Perancangan *Interface* Halaman Login

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Login untuk menampilkan form login
6	Label register untuk menampilkan form register member
7	Field unuk gambar
8	Field untuk isian nama
9	Field untuk isian password
10	Tombol login untuk memproses proses login

#### 5.1.4.12 Perancangan *Interface* Halaman Utama/Pencarian

Perancangan *interface* dapat dilihat pada Gambar 5.28 Perancangan *Interface* Halaman Utama/Pencarian dan deskripsinya pada Tabel 5.33 Keterangan Perancangan *Interface* Halaman Utama/Pencarian.



Gambar 5.29 Perancangan *Interface* Halaman Utama/Pencarian

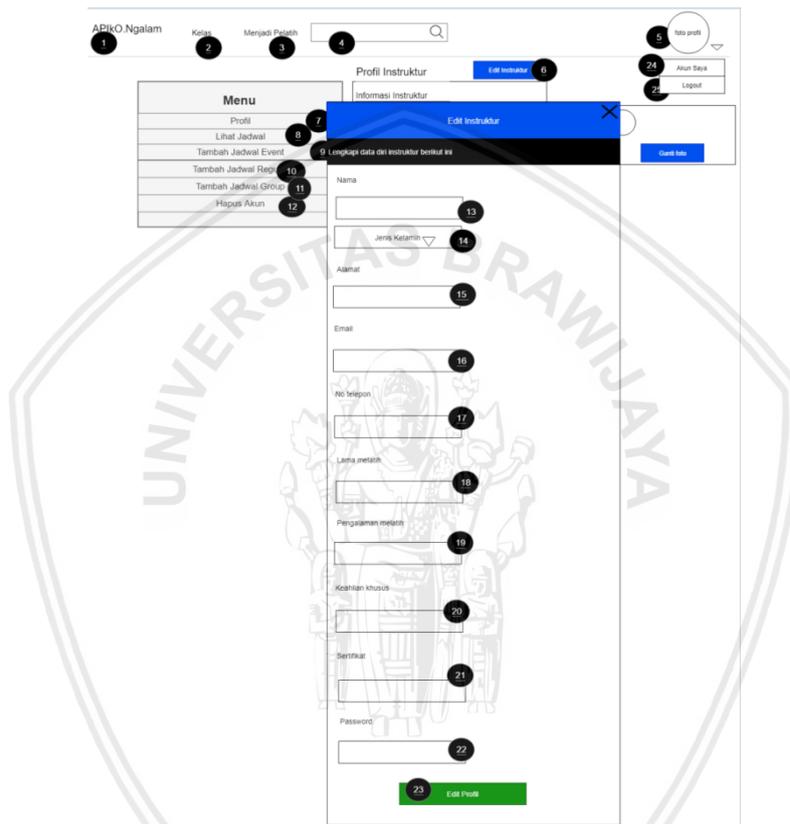
Tabel 5.34 Keterangan Perancangan *Interface* Halaman Utama/Pencarian

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Login untuk menampilkan form login
6	Label register untuk menampilkan form register member
7	Field untuk menampilkan foto
8	Field untuk menampilkan jenis olahraga
9	Field untuk menampilkan jumlah sesi
10	Field untuk menampilkan durasi
11	Field untuk menampilkan jumlah murid
12	Field untuk menampilkan harga
13	Field untuk menampilkan rating

14	Field untuk menampilkan jenis kelas
15	Field untuk menampilkan nama instruktur
16	Tombol pesan untuk memproses pemesanan

### 5.1.4.13 Perancangan *Interface* Halaman Edit Profil Instruktur

Perancangan *interface* dapat dilihat pada Gambar 5.30 Perancangan *Interface* Halaman Edit Profil Instruktur dan deskripsinya pada Tabel 5.35 Keterangan Perancangan *Interface* Halaman Edit Profil Instruktur.



Gambar 5.31 Perancangan *Interface* Halaman Edit Profil Instruktur

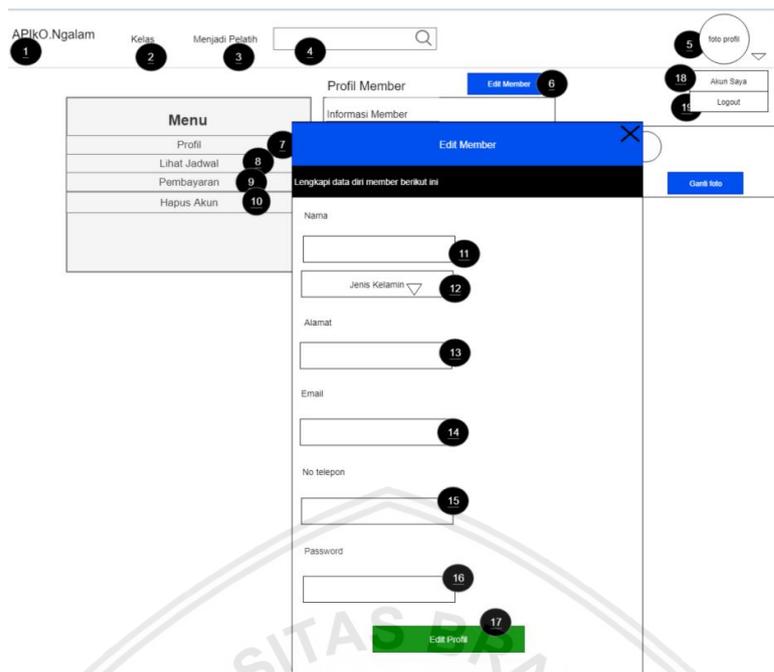
Tabel 5.36 Keterangan Perancangan *Interface* Halaman Edit Profil Instruktur

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Tombol edit instruktur untuk menampilkan form edit instruktur

7	Label profil untuk melihat profil
8	Label Lihat Jadwal untuk menampilkan jadwal
9	Label tambah jadwal event untuk menampilkan form tambah jadwal event
10	Label tambah jadwal regular untuk menampilkan form tambah jadwal regular
11	Label tambah jadwal group untuk menampilkan form tambah jadwal group
12	Label hapus akun berfungsi menghapus akun
13	Field untuk isian nama
14	Field untuk isian jenis kelamin
15	Field untuk isian alamat
16	Field untuk isian email
17	Field untuk isian lama melatih
18	Field untuk no telepon
19	Field untuk isian pengalaman melatih
20	Field untuk isian keahlian khusus
21	Field untuk isian sertifikasi
22	Field untuk isian password
23	Tombol edit instruktur untuk memproses edit profil instruktur
24	Menu Akun Saya untuk menampilkan halaman profil instruktur
25	Menu logout untuk keluar dari sistem

#### 5.1.4.14 Perancangan *Interface* Halaman Edit Profil *Member*

Perancangan *interface* dapat dilihat dari Gambar 5.32 Perancangan *Interface* Halaman Edit Profil *Member* dan deskripsinya pada Tabel 5.37 Keterangan Perancangan *Interface* Halaman Edit Profil *Member*.



**Gambar 5.33 Perancangan *Interface* Halaman Edit Profil *Member***

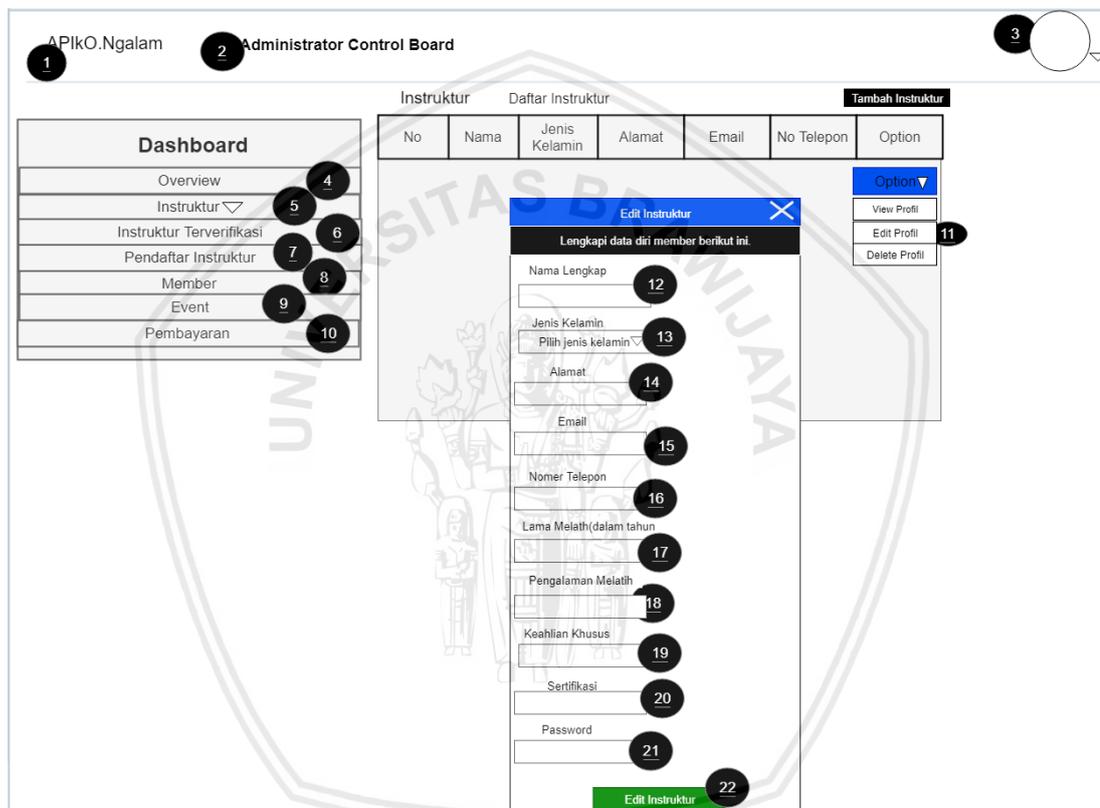
**Tabel 5.38 Keterangan Perancangan *Interface* Halaman Edit Profil *Member***

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Tombol Edit Profil untuk menampilkan form edit form member
7	Label profil untuk melihat profil
8	Label Lihat Jadwal untuk menampilkan jadwal
9	Label pembayaran untuk menampilkan list pembayaran
10	Label hapus akun berfungsi menghapus akun
11	Field untuk isian nama
12	Field untuk isian jenis kelamin
13	Field untuk isian alamat
14	Field untuk isian email
15	Field untuk isian no telepon
16	Field untuk isian password

17	Tombol edit member berfungsi untuk mengedit profil member
18	Menu Akun Saya untuk menampilkan halaman profil member
19	Menu logout untuk keluar dari sistem

**5.1.4.15 Perancangan *Interface* Halaman Edit Instruktur**

Perancangan *interface* dapat dilihat pada Gambar 5.34 Perancangan *Interface* Halaman Edit Instruktur dan deskripsinya pada Tabel 5.39 Keterangan Perancangan *Interface* Halaman Membuat Instruktur.



**Gambar 5.35 Perancangan *Interface* Halaman Edit Instruktur**

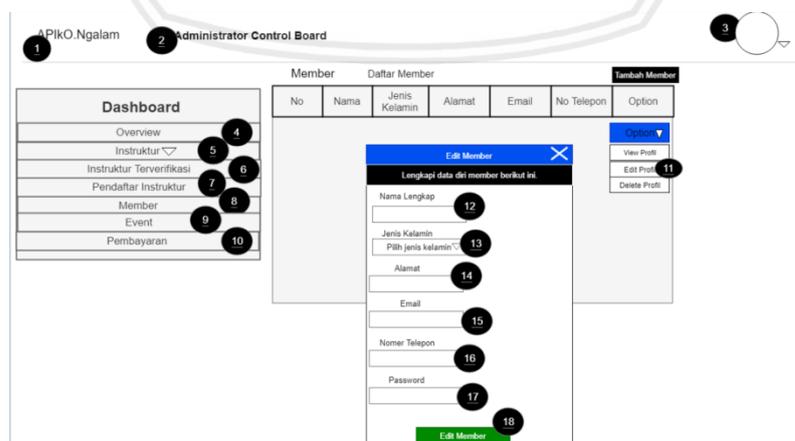
**Tabel 5.40 Keterangan Perancangan *Interface* Halaman Membuat Instruktur**

No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown

6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tombol edit instruktur berfungsi untuk menampilkan form edit instruktur
12	Field untuk isian nama
13	Tombol dropdown untuk memilih jenis kelamin
14	Field untuk isian alamat
15	Field untuk isian email
16	Field untuk isian nomer telepon
17	Field untuk isian lama melatih
18	Field untuk isian pengalaman melatih
19	Field untuk isian keahlian khusus
20	Field untuk isian sertifikasi
21	Field untuk isian password
22	Tombol edit instruktur berfungsi untuk mengedit instruktur

#### 5.1.4.16 Perancangan *Interface* Halaman Edit Member

Perancangan *interface* dapat dilihat pada Gambar 5.36 Perancangan *Interface* Halaman Edit Member dan deskripsinya pada Tabel 5.41 Keterangan Perancangan *Interface* Halaman Edit Member.



Gambar 5.37 Perancangan *Interface* Halaman Edit Member

**Tabel 5.42 Keterangan Perancangan *Interface* Halaman Edit *Member***

No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member
9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tombol edit member berfungsi untuk menampilkan form edit member
12	Field untuk isian nama
13	Tombol dropdown untuk memilih jenis kelamin
14	Field untuk isian alamat
15	Field untuk isian email
16	Field untuk isian nomer telepon
17	Field untuk isian password
18	Tombol edit member berfungsi untuk mengedit member

#### 5.1.4.17 Perancangan *Interface* Halaman *Login Admin*

Perancangan *interface* dapat dilihat pada Gambar 5.38 Perancangan *Interface* Halaman *Login Admin* dan deskripsinya pada Tabel 5.43 Keterangan Perancangan *Interface* Halaman *Login Admin*.

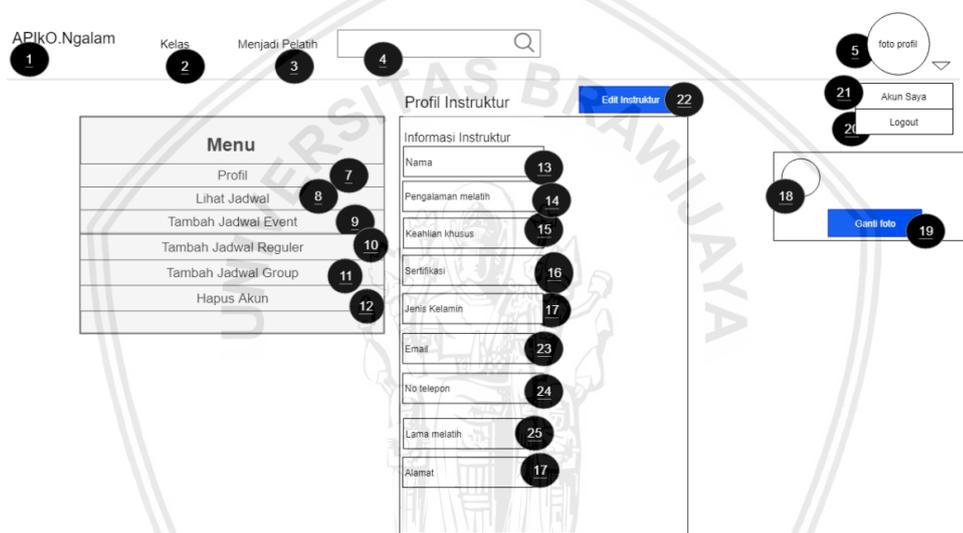
**Gambar 5.39 Perancangan *Interface* Halaman *Login Admin***

**Tabel 5.44 Keterangan Perancangan *Interface* Halaman Login Admin**

No	Deskripsi
1	Field untuk isian nama
2	Field untuk isian password
3	Tombol sign in untuk proses login admin
4	Field untuk keterangan

**5.1.4.18 Perancangan *Interface* Halaman Lihat Profil Instruktur**

Perancangan *interface* dapat dilihat pada Gambar 5.40 Perancangan *Interface* Halaman Lihat Profil Instruktur dan deskripsinya pada Tabel 5.45 Keterangan Perancangan *Interface* Halaman Lihat Profil Instruktur.



**Gambar 5.41 Perancangan *Interface* Halaman Lihat Profil Instruktur**

**Tabel 5.46 Keterangan Perancangan *Interface* Halaman Lihat Profil Instruktur**

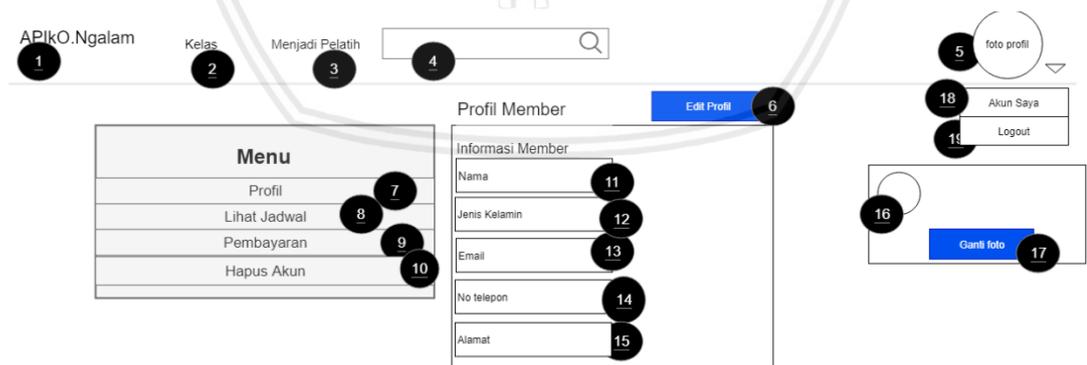
No	Deskripsi
1	Label Judul Sistem
2	Label Judul Halaman Admin
3	Menu Logo Admin
4	Label Overview berfungsi untuk melihat dashboard
5	Label Instruktur untuk menampilkan dropdown
6	Label Instruktur berfungsi untuk melihat daftar instruktur yang terverifikasi
7	Label Instruktur berfungsi untuk melihat daftar instruktur yang belum terverifikasi
8	Label member berfungsi untuk melihat daftar member



9	Label event berfungsi untuk melihat daftar event
10	Label pembayaran berfungsi untuk melihat daftar pembayaran
11	Tombol edit instruktur berfungsi untuk menampilkan form edit instruktur
12	Field untuk menampilkan nama instruktur
13	Field untuk menampilkan pengalaman melatih
14	Field untuk menampilkan keahlian khusus
15	Field untuk menampilkan sertifikasi
16	Field untuk menampilkan jenis kelamin
17	Field untuk menampilkan email
18	Field untuk menampilkan no telepon
19	Field untuk menampilkan lama melatih
20	Field untuk menampilkan alamat
21	Field untuk menampilkan foto profil instruktur
22	Tombol ganti foto berfungsi untuk mengedit foto instruktur
23	Label Menu Akun Saya menampilkan halaman profil instruktur
24	Label menu logout untuk keluar dari sistem

**5.1.4.19 Perancangan *Interface* Halaman Lihat Profil *Member***

Perancangan *interface* dapat dilihat pada Gambar 5.42 Perancangan *Interface* Halaman Lihat Profil *Member* dan deskripsinya pada Tabel 5.47 Keterangan Perancangan *Interface* Halaman Lihat Profil *Member*.



**Gambar 5.43 Perancangan *Interface* Halaman Lihat Profil *Member***

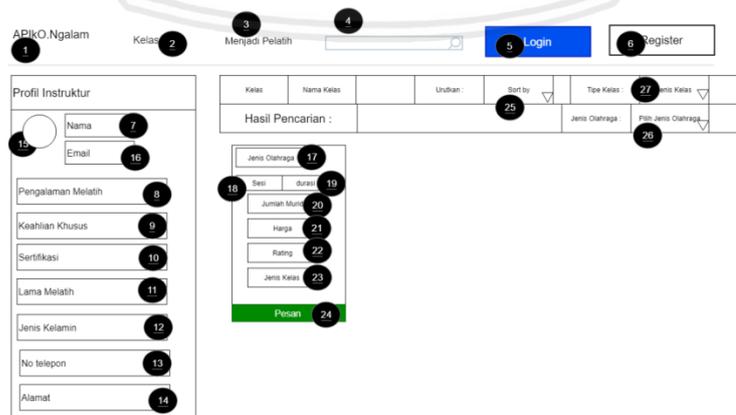
**Tabel 5.48 Keterangan Perancangan *Interface* Halaman Lihat Profil *Member***

No	Deskripsi
1	Label Judul Sistem

2	Label Kelas untuk menampilkan drop down
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Tombol Edit Profil untuk menampilkan form edit form member
7	Label profil untuk melihat profil
8	Label Lihat Jadwal untuk menampilkan jadwal
9	Label pembayaran untuk menampilkan list pembayaran
10	Label hapus akun berfungsi menghapus akun
11	Field untuk menampilkan nama member
12	Field untuk menampilkan jenis kelamin
13	Field untuk menampilkan email
14	Field untuk menampilkan no telepon
15	Field untuk menampilkan alamat
16	Field untuk menampilkan foto
17	Tombol ganti berfungsi untuk mengganti foto member
18	Menu Akun Saya untuk menampilkan halaman profil member
19	Menu logout untuk keluar dari sistem

#### 5.1.4.20 Perancangan *Interface* Halaman Lihat Detail Profil Instruktur

Perancangan *interface* dapat dilihat pada Gambar 5.44 Perancangan *Interface* Halaman Lihat Detail Profil Instruktur dan deskripsinya pada Tabel 5.49 Keterangan Perancangan *Interface* Halaman Lihat Detail Profil Instruktur.



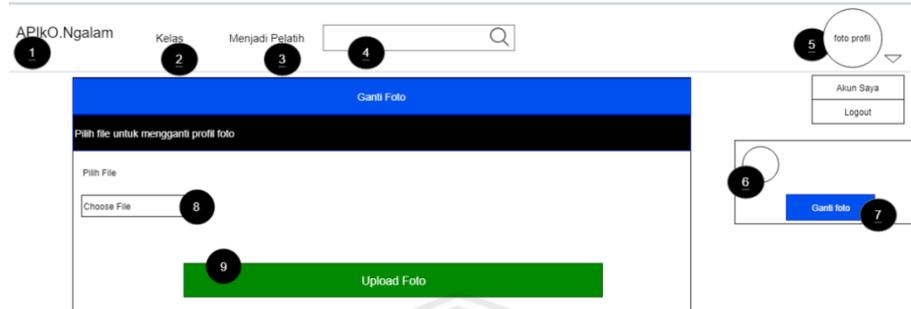
Gambar 5.45 Perancangan *Interface* Halaman Lihat Detail Profil Instruktur

**Tabel 5.50 Keterangan Perancangan *Interface* Halaman Lihat Detail Profil Instruktur**

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Login untuk menampilkan form login
6	Label register untuk menampilkan form register member
7	Field untuk menampilkan nama pelatih
8	Field untuk menampilkan nama pengalaman melatih
9	Field untuk menampilkan keahlian khusus
10	Field untuk menampilkan sertifikasi
11	Field untuk menampilkan lama melatih
12	Field untuk menampilkan jenis kelamin
13	Field untuk menampilkan no telepon
14	Field untuk menampilkan alamat
15	Field untuk menampilkan foto
16	Field untuk menampilkan email
17	Tombol untuk menampilkan jenis olahraga
18	Tombol untuk menampilkan sesi
19	Tombol untuk menampilkan durasi
20	Tombol untuk menampilkan jumlah murid
21	Tombol untuk menampilkan harga
22	Tombol untuk menampilkan rating
23	Tombol untuk menampilkan jenis kelas
24	Tombol pesan untuk memproses pemesanan
25	Dropdown untuk mengurutkan urutan pencarian berdasarkan kriteria
26	Dropdown untuk pencarian jenis olahraga
27	Dropdown untuk pencarian jenis kelas

#### 5.1.4.21 Perancangan *Interface* Halaman Edit Foto

Perancangan *interface* dapat dilihat pada Gambar 5.46 Perancangan *Interface* Halaman Edit Foto dan deskripsinya pada Tabel 5.51 Keterangan Perancangan *Interface* Halaman Edit Foto.



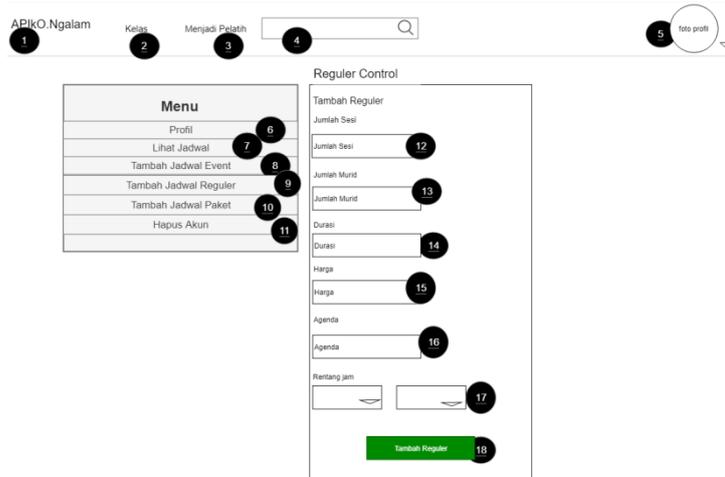
Gambar 5.47 Perancangan *Interface* Halaman Edit Foto

Tabel 5.52 Keterangan Perancangan *Interface* Halaman Edit Foto

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan foto
7	Tombol ganti foto untuk menampilkan form edit foto
8	Tombol choose file untuk memilih file foto
9	Tombol upload foto untuk memproses pergantian foto

#### 5.1.4.22 Perancangan *Interface* Halaman Buat Kelas Reguler

Perancangan *interface* dapat dilihat pada Gambar 5.48 Perancangan *Interface* Halaman Buat Kelas Reguler dan deskripsinya pada Tabel 5.53 Keterangan Perancangan *Interface* Buat Kelas Reguler.



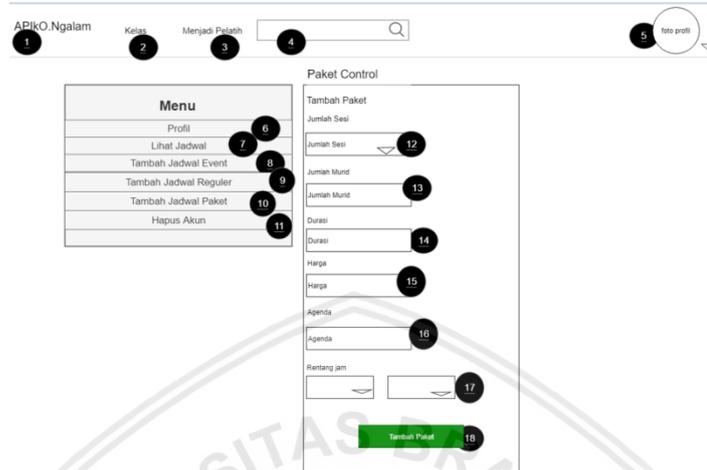
**Gambar 5.49 Perancangan *Interface* Halaman Buat Kelas Regular**

**Tabel 5.54 Keterangan Perancangan *Interface* Buat Kelas Regular**

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil instruktur
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan form buat kelas event
9	Field untuk menampilkan form buat kelas regular
10	Field untuk menampilkan form buat kelas paket
11	Field untuk menghapus instruktur
12	Field untuk mengisi jumlah sesi
13	Field untuk mengisi jumlah murid
14	Field untuk mengisi durasi
15	Field untuk mengisi harga
16	Field untuk mengisi agenda olahraga
17	Field untuk mengisi rentang jam olahraga
18	Tombol tambah regular untuk membuat kelas regular

### 5.1.4.23 Perancangan *Interface* Halaman Buat Kelas Paket

Perancangan *interface* dapat dilihat pada Gambar 5.50 Perancangan *Interface* Halaman Buat Kelas Paket dan deskripsinya pada Tabel 5.55 Keterangan Perancangan *Interface* Buat Kelas Paket.



**Gambar 5.51 Perancangan *Interface* Halaman Buat Kelas Paket**

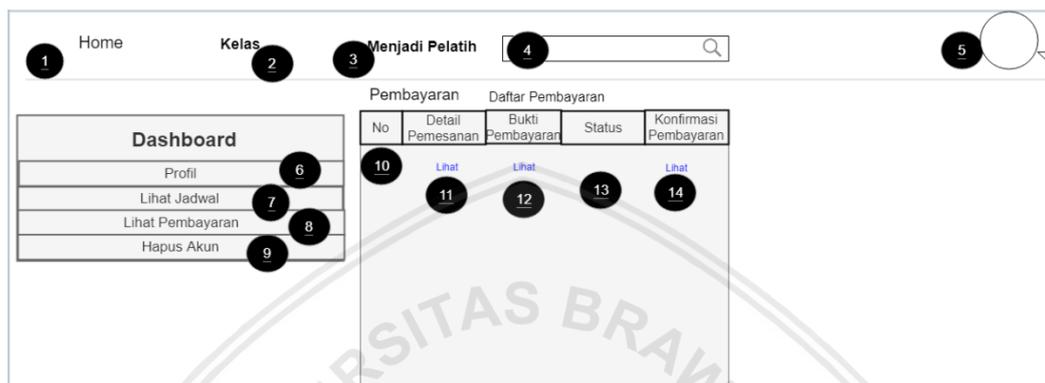
**Tabel 5.56 Keterangan Perancangan *Interface* Buat Kelas Paket**

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil instruktur
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan form buat kelas event
9	Field untuk menampilkan form buat kelas regular
10	Field untuk menampilkan form buat kelas paket
11	Field untuk menghapus akun instruktur
12	Field untuk mengisi jumlah sesi
13	Field untuk mengisi jumlah murid
14	Field untuk mengisi durasi
15	Field untuk mengisi harga
16	Field untuk mengisi agenda olahraga

17	Field untuk memilih rentang jam olahraga
18	Tombol untuk membuat kelas paket

#### 5.1.4.24 Perancangan *Interface* Halaman Lihat Daftar Pembayaran

Perancangan *interface* dapat dilihat pada Gambar 5.52 Perancangan *Interface* Halaman Lihat Daftar Pembayaran dan deskripsinya pada Tabel 5.57 Keterangan Perancangan *Interface* Halaman Lihat Daftar Pembayaran.



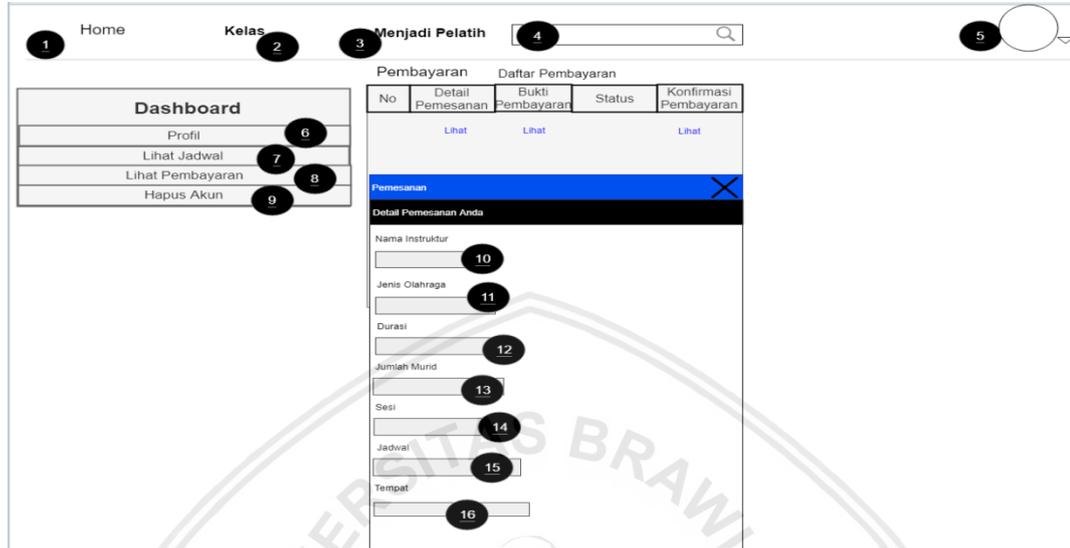
**Gambar 5.53** Perancangan *Interface* Halaman Lihat Daftar Pembayaran

**Tabel 5.58** Keterangan Perancangan *Interface* Halaman Lihat Daftar Pembayaran

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil member
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menghapus instruktur
10	Tabel untuk menampilkan no daftar pembayaran
11	Tabel untuk menampilkan detail pemesanan
12	Tabel untuk menampilkan bukti pembayaran
13	Tabel untuk menampilkan status pembayaran
14	Tabel untuk menampilkan form konfirmasi pembayaran dan kode pembayaran

### 5.1.4.25 Perancangan *Interface* Lihat Detail Pemesanan

Perancangan *interface* dapat dilihat pada Gambar 5.54 Perancangan *Interface* Lihat Detail Pemesanan dan deskripsinya pada Tabel 5.59 Keterangan Perancangan *Interface* Lihat Detail Pemesanan.



Gambar 5.55 Perancangan *Interface* Lihat Detail Pemesanan

Tabel 5.60 Keterangan Perancangan *Interface* Lihat Detail Pemesanan

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil member
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menghapus instruktur
10	Field untuk menampilkan nama instruktur
11	Field untuk menampilkan jenis olahraga
12	Field untuk menampilkan durasi
13	Field untuk menampilkan jumlah murid
14	Field untuk menampilkan sesi

15	Field untuk menampilkan jadwal
16	Field untuk menampilkan tempat

**5.1.4.26 Perancangan *Interface* Lihat Bukti Pembayaran**

Perancangan *interface* dapat dilihat pada Gambar 5.56 Perancangan *Interface* Lihat Bukti Pembayaran dan deskripsinya pada Tabel 5.61 Keterangan Perancangan *Interface* Lihat Detail Pemesanan.



**Gambar 5.57 Perancangan *Interface* Lihat Bukti Pembayaran**

**Tabel 5.62 Keterangan Perancangan *Interface* Lihat Detail Pemesanan**

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil member
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menghapus instruktur
10	Field untuk menampilkan bukti pembayaran

**5.1.4.27 Perancangan *Interface* Cek Kode Pembayaran**

Perancangan *interface* dapat dilihat pada Gambar 5.58 Perancangan *Interface* Lihat Bukti Pembayaran dan deskripsinya pada Tabel 5.63 Keterangan Perancangan *Interface* Lihat Detail Pemesanan.



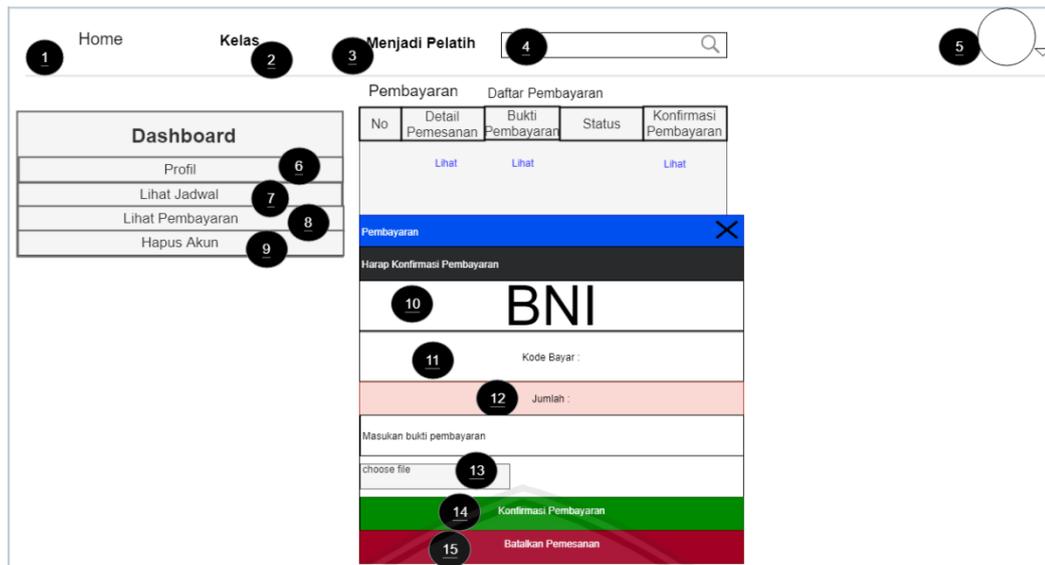
Gambar 5.59 Perancangan *Interface* Lihat Bukti Pembayaran

Tabel 5.64 Keterangan Perancangan *Interface* Lihat Detail Pemesanan

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil member
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menghapus instruktur
10	Field untuk menampilkan nama bank
11	Field untuk menampilkan kode bayar
12	Field untuk menampilkan jumlah pembayaran

#### 5.1.4.28 Perancangan *Interface* Konfirmasi Pembayaran

Perancangan *interface* dapat dilihat pada Gambar 5.60 Perancangan *Interface* Konfirmasi Pembayaran dan deskripsinya pada Tabel 5.65 Keterangan Perancangan *Interface* Konfirmasi Pembayaran.



Gambar 5.61 Perancangan *Interface* Konfirmasi Pembayaran

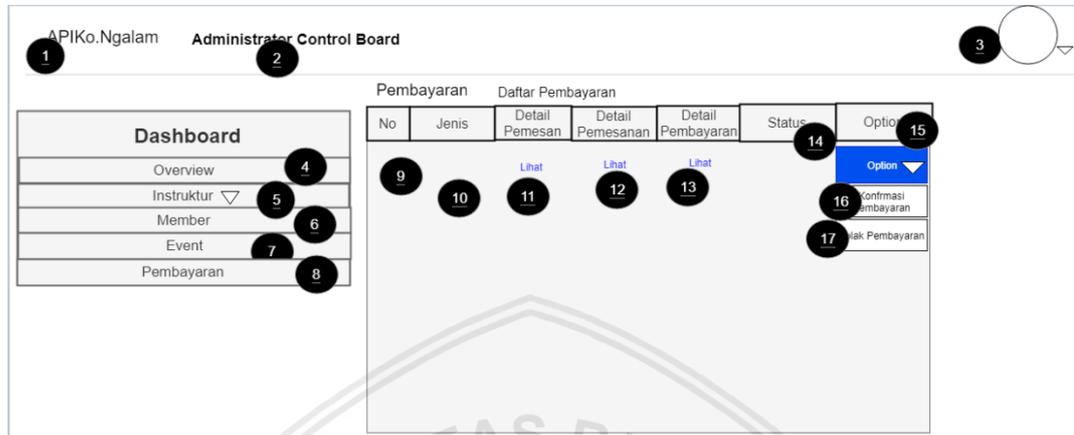
Tabel 5.66 Keterangan Perancangan *Interface* Konfirmasi Pembayaran

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Label Menu Logo Member/Instruktur
6	Field untuk menampilkan profil member
7	Field untuk menampilkan jadwal latihan
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menghapus instruktur
10	Field untuk menampilkan nama bank
11	Field untuk menampilkan kode bayar
12	Field untuk menampilkan jumlah pembayaran
13	Tombol untuk mengupload bukti pembayaran
14	Tombol untuk mengkonfirmasi pembayaran
15	Tombol untuk membatalkan pembayaran



### 5.1.4.29 Perancangan *Interface* Lihat Daftar Pembayaran *Member*

Perancangan *interface* dapat dilihat pada Gambar 5.62 Perancangan *Interface* Lihat Daftar Pembayaran *Member* dan deskripsinya pada Tabel 5.67 Keterangan Perancangan *Interface* Lihat Daftar Pembayaran *Member*.



**Gambar 5.63** Perancangan *Interface* Lihat Daftar Pembayaran *Member*

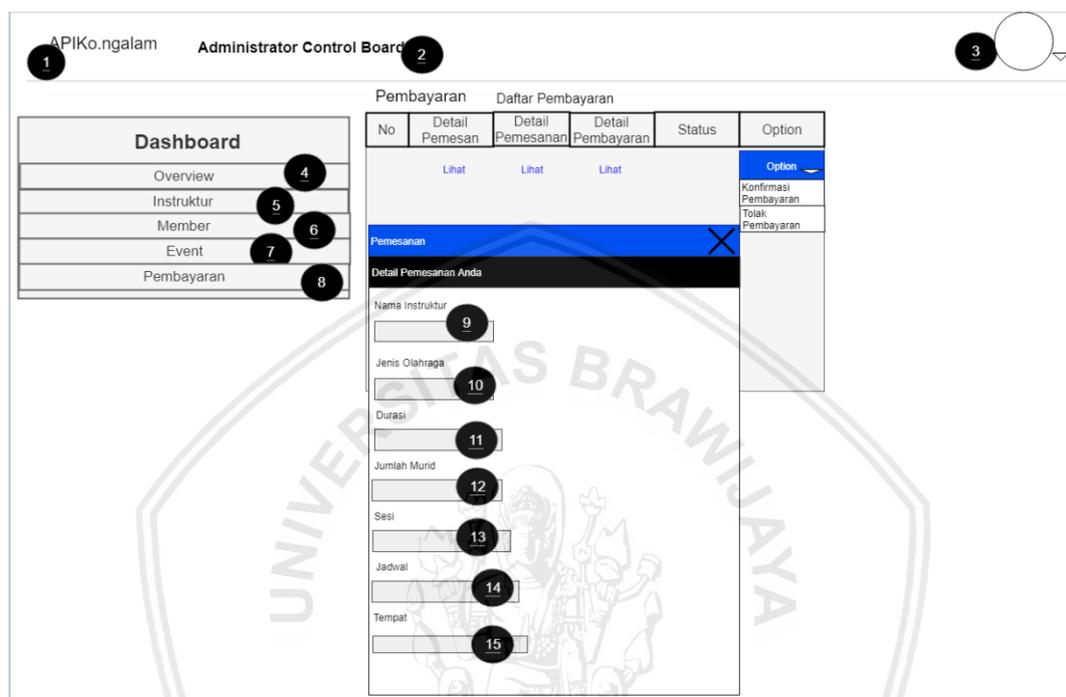
**Tabel 5.68** Keterangan Perancangan *Interface* Lihat Daftar Pembayaran *Member*

No	Deskripsi
1	Label Judul Sistem
2	Label untuk admin
3	Label Menu Logo Member/Instruktur
4	Field untuk menampilkan dashboard
5	Field untuk menampilkan daftar instruktur
6	Field untuk menampilkan daftar member
7	Field untuk menampilkan daftar event
8	Field untuk menampilkan daftar pembayaran
9	Tabel untuk menampilkan no daftar pembayaran
10	Tabel untuk menampilkan jenis kelas yang dipesan
11	Tabel untuk menampilkan detail pemesan
12	Tabel untuk menampilkan detail pemesanan
13	Tabel untuk menampilkan detail pembayaran
14	Tabel untuk menampilkan status pembayaran
15	Tabel untuk menampilkan option
16	Tombol untuk mengkonfirmasi pembayaran

17	Tombol untuk menolak pembayaran
----	---------------------------------

### 5.1.4.30 Perancangan *Interface* Lihat Detail Pemesanan *Member*

Perancangan *interface* dapat dilihat pada Gambar 5.64 Perancangan *Interface* Lihat Detail Pemesanan *Member* dan deskripsinya pada Tabel 5.69 Keterangan Perancangan *Interface* Lihat Detail Pemesanan *Member*.



Gambar 5.65 Perancangan *Interface* Lihat Detail Pemesanan *Member*

Tabel 5.70 Keterangan Perancangan *Interface* Lihat Detail Pemesanan *Member*

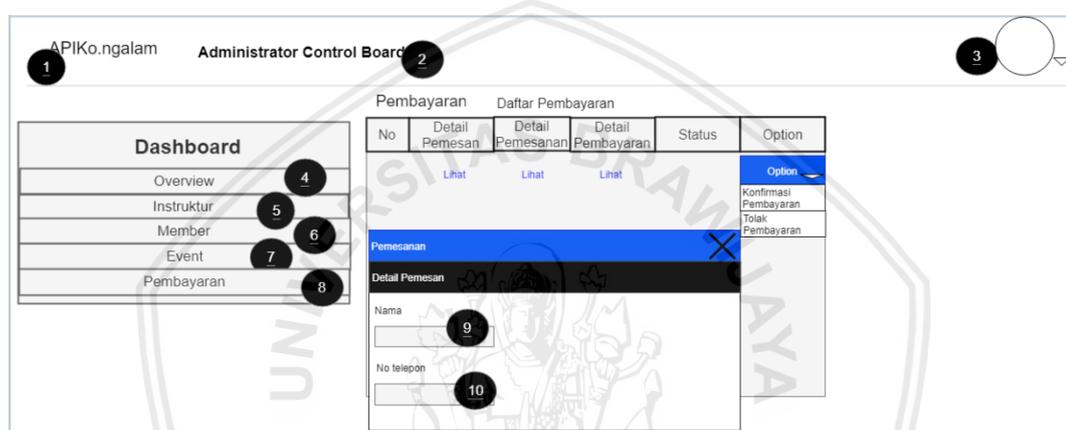
No	Deskripsi
1	Label Judul Sistem
2	Label untuk admin
3	Label Menu Logo Member/Instruktur
4	Field untuk menampilkan dashboard
5	Field untuk menampilkan daftar instruktur
6	Field untuk menampilkan daftar member
7	Field untuk menampilkan daftar event
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menampilkan nama instruktur
10	Field untuk menampilkan jenis olahraga



11	Field untuk menampilkan durasi
12	Field untuk menampilkan jumlah murid
13	Field untuk menampilkan sesi
14	Field untuk menampilkan jadwal
15	Field untuk menampilkan tempat

### 5.1.4.31 Perancangan *Interface* Lihat Detail Pemesan

Perancangan *interface* dapat dilihat Gambar 5.66 Perancangan *Interface* Lihat Detail Pemesan dan deskripsinya pada Tabel 5.71 Keterangan Perancangan *Interface* Lihat Detail Pemesan.



Gambar 5.67 Perancangan *Interface* Lihat Detail Pemesan

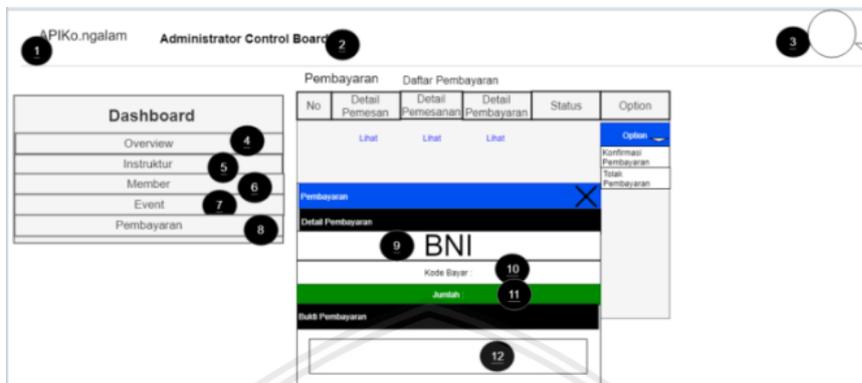
Tabel 5.72 Keterangan Perancangan *Interface* Lihat Detail Pemesan

No	Deskripsi
1	Label Judul Sistem
2	Label untuk admin
3	Label Menu Logo Member/Instruktur
4	Field untuk menampilkan dashboard
5	Field untuk menampilkan daftar instruktur
6	Field untuk menampilkan daftar member
7	Field untuk menampilkan daftar event
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menampilkan nama pemesan
10	Field untuk menampilkan no telepon



### 5.1.4.32 Perancangan *Interface* Lihat Detail Pembayaran

Perancangan *interface* dapat dilihat pada Gambar 5.68 Perancangan *Interface* Lihat Detail Pembayaran dan deskripsinya pada Tabel 5.73 Keterangan Perancangan *Interface* Lihat Detail Pembayaran.



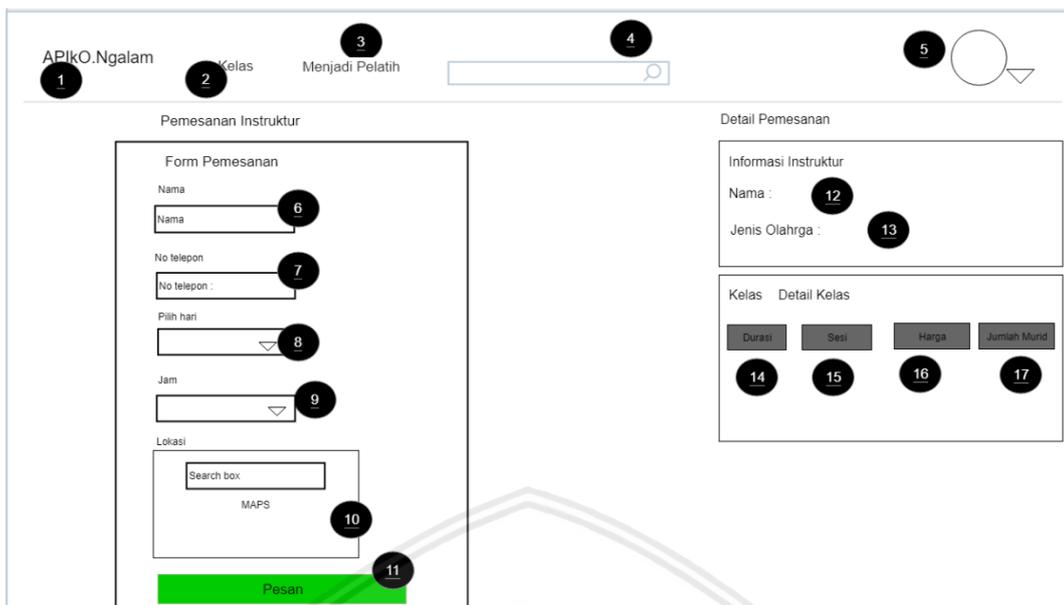
Gambar 5.69 Perancangan *Interface* Lihat Detail Pembayaran

Tabel 5.74 Keterangan Perancangan *Interface* Lihat Detail Pembayaran

No	Deskripsi
1	Label Judul Sistem
2	Label untuk admin
3	Label Menu Logo Member/Instruktur
4	Field untuk menampilkan dashboard
5	Field untuk menampilkan daftar instruktur
6	Field untuk menampilkan daftar member
7	Field untuk menampilkan daftar event
8	Field untuk menampilkan daftar pembayaran
9	Field untuk menampilkan nama bank
10	Field untuk menampilkan kode bayar
11	Field untuk menampilkan jumlah pembayaran
12	Field untuk menampilkan bukti pembayaran

### 5.1.4.33 Perancangan *Interface* Halaman Pemesanan Kelas Paket

Perancangan *interface* dapat dilihat pada Gambar 5.70 Perancangan *Interface* Halaman Pemesanan Kelas Paket dan deskripsinya pada Tabel 5.75 Keterangan Perancangan *Interface* Halaman Pemesanan Kelas Paket.



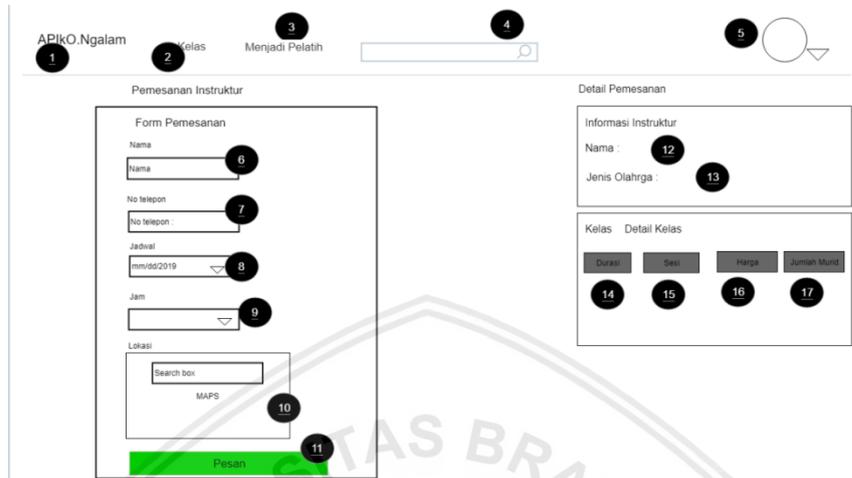
**Gambar 5.71** Perancangan *Interface* Halaman Pemesanan Kelas Paket

**Tabel 5.76** Keterangan Perancangan *Interface* Halaman Pemesanan Kelas Paket

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Field untuk isian nama
7	Field untuk isian no telepon
8	Field untuk isian lokasi
9	Tombol pesan untuk melakukan pemesanan
10	Tabel untuk menampilkan nama instruktur
11	Tabel untuk menampilkan jenis olahraga
12	Tabel untuk menampilkan durasi
13	Tabel untuk menampilkan jumlah murid
14	Tabel untuk menampilkan jumlah sesi
15	Tabel untuk menampilkan harga
16	Tabel untuk menampilkan jadwal
17	Tabel untuk menampilkan jumlah murid

### 5.1.4.34 Perancangan *Interface* Halaman Pemesanan Kelas Reguler

Perancangan *interface* dapat dilihat pada Gambar 5.72 Perancangan *Interface* Halaman Pemesanan Kelas Reguler dan deskripsinya pada Tabel 5.77 Keterangan Perancangan *Interface* Halaman Pemesanan Kelas Reguler.



**Gambar 5.73** Perancangan *Interface* Halaman Pemesanan Kelas Reguler

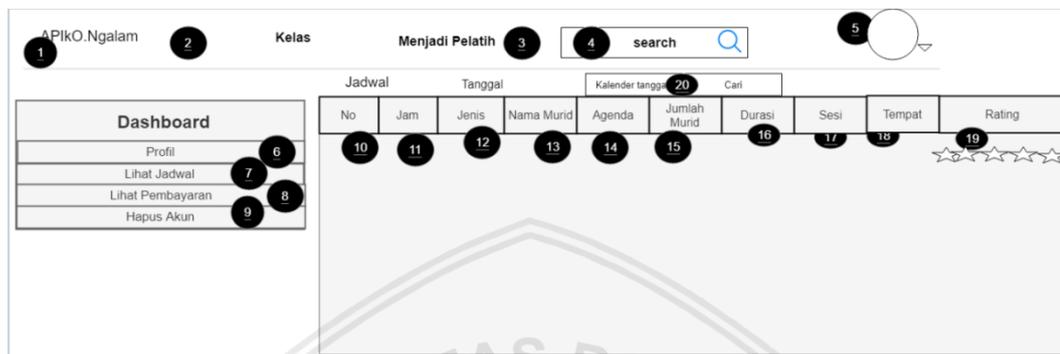
**Tabel 5.78** Keterangan Perancangan *Interface* Halaman Pemesanan Kelas Reguler

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down pilihan kelas
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Field untuk isian nama
7	Field untuk isian no telepon
8	Field untuk isian lokasi
9	Tombol pesan untuk melakukan pemesanan
10	Tabel untuk menampilkan nama instruktur
11	Tabel untuk menampilkan jenis olahraga
12	Tabel untuk menampilkan durasi
13	Tabel untuk menampilkan jumlah murid
14	Tabel untuk menampilkan jumlah sesi
15	Tabel untuk menampilkan harga
16	Tabel untuk menampilkan jadwal

17	Tabel untuk menampilkan jumlah murid
----	--------------------------------------

**5.1.4.35 Perancangan *Interface* Halaman Lihat Jadwal *Member***

Perancangan *interface* dapat dilihat pada Gambar 5.74 Perancangan *Interface* Halaman Lihat Jadwal *Member* dan deskripsinya pada Tabel 5.79 Keterangan Perancangan *Interface* Halaman Lihat Jadwal *Member*.



**Gambar 5.75 Perancangan *Interface* Halaman Lihat Jadwal *Member***

**Tabel 5.80 Keterangan Perancangan *Interface* Halaman Lihat Jadwal *Member***

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Label profil untuk melihat profil
7	Label Lihat Jadwal untuk menampilkan jadwal
8	Label pembayaran untuk menampilkan list pembayaran
9	Label hapus akun berfungsi menghapus akun
10	Field untuk menampilkan no
11	Field untuk menampilkan jam
12	Field untuk menampilkan jenis olahraga
13	Field untuk menampilkan nama instruktur
14	Field untuk menampilkan agenda
15	Field untuk menampilkan jumlah murid
16	Field untuk menampilkan durasi



17	Field untuk menampilkan sesi
18	Field untuk menampilkan tempat
19	Field untuk menampilkan rating dan memberi rating
20	Field untuk mencari jadwal dengan tanggal

### 5.1.4.36 Perancangan *Interface* Halaman Lihat Jadwal Instruktur Olahraga

Perancangan *interface* dapat dilihat pada Gambar 5.76 Perancangan *Interface* Halaman Lihat Jadwal Instruktur Olahraga dan deskripsinya pada Tabel 5.81 Keterangan Perancangan *Interface* Halaman Lihat Jadwal Instruktur Olahraga.



**Gambar 5.77 Perancangan *Interface* Halaman Lihat Jadwal Instruktur Olahraga**  
**Tabel 5.82 Keterangan Perancangan *Interface* Halaman Lihat Jadwal Instruktur Olahraga**

No	Deskripsi
1	Label Judul Sistem
2	Label Kelas untuk menampilkan drop down
3	Label Menjadi Pelatih untuk menampilkan form pendaftaran pelatih
4	Field pencarian instruktur
5	Menu logo instruktur
6	Label profil untuk melihat profil
7	Label Lihat Jadwal untuk menampilkan jadwal
8	Label pembayaran untuk menampilkan list pembayaran
9	Label hapus akun berfungsi menghapus akun
10	Field untuk menampilkan no
11	Field untuk menampilkan jam
12	Field untuk menampilkan jenis olahraga
13	Field untuk menampilkan nama murid



14	Field untuk menampilkan agenda
15	Field untuk menampilkan jumlah murid
16	Field untuk menampilkan durasi
17	Field untuk menampilkan sesi
18	Field untuk menampilkan tempat
19	Field untuk menampilkan rating dan memberi rating
20	Field untuk mencari jadwal dengan tanggal

#### 5.1.4.37 Perancangan *Interface* Halaman *Dashboard Admin*

Perancangan *interface* dapat dilihat pada Gambar 5.78 Perancangan *Interface* Halaman *Dashboard Admin* dan deskripsinya pada Tabel 5.83 Keterangan Perancangan *Interface* Halaman *Dashboard Admin*.



**Gambar 5.79** Perancangan *Interface* Halaman *Dashboard Admin*

**Tabel 5.84** Keterangan Perancangan *Interface* Halaman *Dashboard Admin*

No	Deskripsi
1	Label Judul Sistem
2	Label untuk admin
3	Label Menu Logo Member/Instruktur
4	Field untuk menampilkan dashboard
5	Field untuk menampilkan daftar instruktur
7	Field untuk menampilkan daftar instruktur terverifikasi
8	Field untuk menampilkan daftar pendaftar instruktur
9	Field untuk menampilkan daftar member
10	Field untuk menampilkan daftar event
11	Field untuk menampilkan daftar pembayaran

12	Field untuk jumlah member
13	Field untuk jumlah instruktur
14	Field untuk melihat pendaftar instruktur
15	Field untuk menampilkan total kelas regular
16	Field untuk menampilkan total kelas paket
17	Field untuk menampilkan total kelas event
18	Field untuk melihat event yang perlu diverifikasi
19	Field untuk menampilkan pembayaran yang menunggu dikonfirmasi
20	Field untuk menampilkan transaksi yang sukses dikonfirmasi
21	Field untuk menampilkan pembayaran yang menunggu untuk dibayar

## 5.2 Implementasi Sistem

*Implementasi* sistem menjelaskan tentang proses pembangunan sistem dalam bentuk *coding*. Tahap *implementasi* sistem didasarkan pada perancangan yang dibuat. Tahapan *implementasi* terdiri dari *spesifikasi* perangkat keras dan perangkat lunak untuk membangun sistem, *implementasi source code* yang didasarkan pada rancangan *algoritma* dan implementasi *interface* yang didasarkan pada perancangan *interface* sebelumnya. Pada penelitian ini untuk implementasi interface nya dapat dilihat pada domain website <http://apiko.4pps.web.id> dengan username untuk login sebagai member yaitu `dias@gmail.com` dengan password 123456, username untuk login sebagai instruktur yaitu `tyaradiassananda@gmail.com` dengan password 123456, dan username untuk login admin yaitu `tyara@gmail.com` dengan password 123456.

### 5.2.1 Spesifikasi Sistem

*Spesifikasi* sistem adalah segala bentuk sumber daya yang digunakan untuk membangun rekayasa aplikasi, yang mana pada penelitian ini terdiri dari *spesifikasi* perangkat keras dan *spesifikasi* perangkat lunak.

#### 5.2.1.1 Spesifikasi Perangkat Keras

*Spesifikasi* Perangkat Keras yang digunakan untuk membangun aplikasi pemesanan jasa instruktur olahraga kota Malang berbasis web dijelaskan pada tabel 5.43 *Spesifikasi* Perangkat Keras dibawah ini.

**Tabel 5.85 Spesifikasi Perangkat Keras**

Nama Komponen	Spesifikasi
<i>System Model</i>	<i>Acer Aspire 4752Z</i>
<i>Processor</i>	<i>Intel(R)Pentium(R)CPU B960 @2.20GHz 2.20GHz</i>

Memory	2.00 Gb
Display	Intel(R) HD Graphics

### 5.2.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak digunakan untuk membangun aplikasi pemesanan jasa instruktur kota Malang berbasis web pada tabel 5.44 Spesifikasi Perangkat Lunak dibawah ini.

**Tabel 5.86 Spesifikasi Perangkat Lunak**

Nama Komponen	Spesifikasi
Operating System	Windows 10 Pro 64-bit
Programming Language	PHP
Programming Environment	Sublime Text 3
Support Device	Xampp 3.2.2
Support Browser	Mozilla Firefox & Google Chrome

### 5.2.2 Implementasi Source Code

Implementasi Source Code berisi source code yang digunakan dengan bahasa php yang mana source code ini didasarkan dari rancangan algoritma yang sebelumnya .Algoritma-algoritma yang berbentuk pseudocode dimana berisi alur logika ini yang diubah menjadi source code.

#### 5.2.2.1 Implementasi Source Code Konfirmasi Pembayaran

Implementasi source code dapat dilihat pada Tabel 5.87 Implementasi Source Code Konfirmasi Pembayaran dibawah ini.

Nama Kelas : Pembayaran

Nama Method : konfirmasi\_pembayaran()

Deskripsi : Source Code ini digunakan untuk menkonfirmasi pembayaran. Operasi ini dipanggil ketika member menekan tombol konfirmasi pembayaran pada halaman pembayaran.php. Source Code ini berisi fungsi method konfirmasi\_pembayaran() yang akan digunakan untuk mengkonfirmasi file foto bukti pembayaran. Jika file foto bukti pembayaran yang diinputkan salah maka proses konfirmasi pembayaran tidak berhasil namun jika yang diinputkan benar maka proses konfirmasi pembayaran berhasil.

**Tabel 5.88 Implementasi Source Code Konfirmasi Pembayaran**

1	<code>public function konfirmasi_pembayaran()</code>
2	<code>{</code>
3	<code>{</code>
4	<code>{</code>
5	<code>{</code>

```

6      $config['upload_path']           = './resources/img/bukti/';
7
8      $config['allowed_types']        = 'jpg|png';
9
10     $config['max_size']              = 3000;
11
12     $config['max_width']             = 5000;
13
14     $config['max_height']           = 5000;
15
16     $config['encrypt_name']         = TRUE;
17
18     $this->load->library('upload', $config);
19
20     if (!$this->upload->do_upload('foto')) {
21         $i = $this->input;
22         // $this->session->set_flashdata('success', 'Update foto
23         alumni gagal. ');
24         // redirect('member/profile/'. $i->post('edit_photo_id'));
25         echo $this->upload->display_errors();
26         exit();
27     } else {
28         $i = $this->input;
29         $data = array(
30             'bukti_transfer' => $this->upload-
31             >data('file_name'),
32             'id_pesan'       => $i->post('id_pesan'),
33             'status'        => '1'
34         );
35         $this->Order_Model->konfirmasi_pembayaran($data);
36         $this->session->set_flashdata('success', 'Update foto
37         berhasil. ');
38         echo "<script type='text/javascript'>
39         alert('Anda berhasil mengonfirmasi pembayaran. Silahkan tunggu
40         pembayaran anda dikonfirmasi oleh admin. ');
41         window.location.href = '" . site_url() . "/order/pembayaran/" .
42         $i->post('id_member') . "'";
43     </script>";
44     }
45 }

```

### 5.2.2.2 Implementasi Source Code Mengkonfirmasi Pendaftaran Instruktur

Implementasi source code dapat dilihat pada Tabel 5.89 Implementasi Source Code Mengkonfirmasi Pendaftaran Instruktur dibawah ini.

Nama Kelas : Instrukturs

Nama Method : confirm\_instruktur(id)

Deskripsi : *Source Code* ini digunakan untuk mengkonfirmasi pendaftaran instruktur. Operasi ini dipanggil saat admin menekan tombol terima pendaftaran pada halaman view-instruktur.php. *Source Code* ini berisi fungsi method confirm\_instruktur() yang akan digunakan untuk mengkonfirmasi pendaftaran instruktur. Ketika admin proses berhasil maka akan mengirim notifikasi email pada instruktur dan jika terjadi eror maka email tidak terkirim.

**Tabel 5.90 Implementasi Source Code Mengkonfirmasi Pendaftaran Instruktur**

```
1 function confirm_instruktur($id){
2
3     $data = array(
4         'id'                => $id,
5         'verified'         => 1
6     );
7
8     $requestor_instruktur = $this->Instruktur_Model-
9 >get_detail_instruktur($id);
10
11     $this->Instruktur_Model->edit_instruktur($data);
12
13     $this->load->library('email');
14     $config = array();
15     $config['charset'] = 'utf-8';
16     $config['useragent'] = 'Codeigniter';
17     $config['protocol'] = "smtp";
18     $config['mailtype'] = "html";
19     $config['smtp_host'] = "ssl://smtp.gmail.com";//pengaturan
20 smtp
21     $config['smtp_port'] = "465";
22     $config['smtp_timeout'] = "400";
23
24     $config['smtp_user'] = "apikomalang@gmail.com"; // isi dengan
25 email kamu
26
27     $config['smtp_pass'] = "dummyemailfordev"; // isi dengan
28 password kamu
29
30     $config['crlf'] = "\r\n";
31
32     $config['newline'] = "\r\n";
33
34     $config['wordwrap'] = true;
35
36     //memanggil library email dan set konfigurasi untuk
37 pengiriman email
38
39     $this->email->initialize($config);
40
41     //konfigurasi pengiriman
42
43
44
```

```

45     $this->email->from($config['smtp_user']);
46     $this->email->to($requestor_instruktur->email);
47     $this->email->subject("Pendaftaran anda diterima!");
48     $message  ="Hi ".$requestor_instruktur->nama."<br>";
49     $message  .="Pendaftaran anda sebagai Instruktur di APIKO Ngalam
50     telah diterima<br>";
51     $message  .="Silahkan login menggunakan email dan password yang
52     sudah terdaftar untuk melakukan perubahan profil, pembuatan kelas
53     dan event<br>";
54     $message  .="<br>";
55     $message  .="<br>";
56     $message  .="=====<br>";
57     $message  .="APIKO Ngalam<br>";
58     $message  .="<a href='localhost/apikomalang'>www.apiko-
59     ngalam.com</a>";
60     $this->email->message($message);
61     if ($this->email->send()) {
62         $this->session->set_flashdata('notification', 'Email berhasil
63         dikirim. ');
64         redirect('administrator/instrukturs/requested');
65     } else {
66         $this->session->set_flashdata('notification', 'Email gagal
67         dikirim. ');
68         redirect('administrator/instrukturs/requested');
69     }
70     $this->session->set_flashdata('notification', 'Berhasil menerima
71     pendaftar instruktur. ');
72     redirect('administrator/instrukturs/requested');
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }

```

### 5.2.2.3 Implementasi Source Code Melakukan Pemesanan Kelas Reguler

*Implementasi source code* dapat dilihat pada Tabel 5.91 *Implementasi Source Code* Melakukan Pemesanan Reguler dibawah ini.

Nama Kelas : Order

Nama Method : pesan\_reguler()

Deskripsi : *Source Code* ini digunakan untuk memesan reguler. Operasi ini dipanggil ketika member menekan tombol pesan pada halaman pemesanan-reguler. *Source Code* ini berisi fungsi method pesan\_reguler() yang akan digunakan untuk proses pemesanan reguler.

Ketika memilih jadwal yang sudah dipesan member lain maka akan menampilkan pesan “Tanggal yyyy-mm-dd Jam X penuh”. Ketika jadwal yang dipesan kosong maka pemesanan reguler berhasil.

**Tabel 5.92 Implementasi Source Code Melakukan Pemesanan Reguler**

```

1 function pesan_reguler() {
2
3     $i = $this->input;
4
5     $cek = array(
6         'id_instruktur' => $i->post('id_instruktur'),
7         'jadwal'         => date("d-m-Y", strtotime($this-
8 >input->post('jadwal'))),
9         'jam'           => $i->post('jam')
10    );
11
12    $num = $this->Order_Model->cek_order_reguler($cek);
13
14    if ($num > 0) {
15        echo "<script type='text/javascript'>
16            alert('Tanggal " . $i->post('jadwal') . " Jam " . $i-
17 >post('jam') . " penuh');
18            window.location.href = " . site_url() .
19            "/order/detail/" . $i->post('id_instruktur') . "/" . $i-
20 >post('id_reguler') . "/reguler" . "'";
21        </script>";
22    } else {
23        $i = $this->input;
24        $data = array(
25            'id_instruktur' => $i->post('id_instruktur'),
26            'id_member'     => $i->post('id_member'),
27            'id_reguler'    => $i->post('id_reguler'),
28            'status'       => 0,
29            'kodebayar'    => uniqid(),
30            'jadwal'       => date("d-m-Y", strtotime($this-
31 >input->post('jadwal'))),
32            'jam'          => $i->post('jam'),
33            'lokasi'       => $i->post('lokasi'),
34            'tempat'      => $i->post('tempat')
35        );
36
37        $id_pemesanan = $this->Order_Model-
38 >pesan_reguler($data);
39
40        redirect('order/pembayaran/' . $i->post('id_member') .
41

```

52	'/' . \$id_pemesanan . '/reguler');
53	}
54	}

#### 5.2.2.4 Implementasi Source Code Menambahkan Member

Implementasi source code dapat dilihat pada Tabel 5.93 Implementasi Source Code Membuat Member dibawah ini.

Nama Kelas : Members

Nama Method : add\_member()

Deskripsi : *Source Code* ini digunakan untuk membuat member. Operasi ini dipanggil saat admin menekan tombol tambah member pada halaman view-member.php. *Source Code* ini berisi fungsi method add\_member() yang akan digunakan untuk menambahkan member. Ketika data yang diinputkan salah maka proses menambahkan member tidak berhasil dibuat, namun jika data yang diinputkan benar maka proses menambahkan member berhasil.

**Tabel 5.94 Implementasi Source Code Membuat Member**

1	function add_member() {
2	\$valid = \$this->form_validation;
3	\$valid->set_rules(
4	'nama',
5	'nama',
6	'required',
7	array( 'required' => 'Anda belum mengisi Nama.')
8	);
9	\$valid->set_rules(
10	'jenis_kelamin',
11	'jenis_kelamin',
12	'required',
13	array( 'required' => 'Anda belum memilih Jenis
14	Kelamin.')
15	);
16	\$valid->set_rules(
17	'alamat',
18	'alamat',
19	'required',
20	array( 'required' => 'Anda belum mengisi Alamat.')
21	);
22	}
23	}

```
34     $valid->set_rules(
35         'email',
36         'email',
37         'required',
38         array( 'required' => 'Anda belum mengisikan Email.')
39     );
40     $valid->set_rules(
41         'no_telepon',
42         'no_telepon',
43         'required|is_natural',
44         array( 'required' => 'Anda belum mengisikan Nomor
45 Telepon.',
46             'is_natural' => 'Nomor telepon harus berupa
47 angka')
48     );
49     $valid->set_rules(
50         'password',
51         'password',
52         'required',
53         array( 'required' => 'Anda belum mengisikan
54 Password.')
55     );
56     if ($valid->run()===false)
57     {
58         $members = $this->Member_Model->get_all_member();
59         $data     = array('isi'      => 'administrator/view-
60 member',
61             'members' => $members
62         );
63         $this->load->view("layouts/wrapper", $data, false);
64     }
65     else {
66         $where = array(
67             'email' => $i->post('email')
68         );
69         $num = $this->Instruktur_Model->check_email($where);
70         $num2 = $this->Member_Model->check_email($where);
71     }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
```

```

92         if ($num > 0 || $num2 > 0) {
93             echo "<script type='text/javascript'>
94                 alert('Email yang anda isi telah terdaftar di sistem. Silahkan
95                 isi dengan email yang belum terdaftar!');
96                 window.location.href = '" . site_url()
97                 ."/administrator/members';
98             </script>";
99         } else {
100             $data = array(
101                 'nama' => $i->post('nama'),
102                 'jenis_kelamin' => $i->post('jenis_kelamin'),
103                 'alamat' => $i->post('alamat'),
104                 'email' => $i->post('email'),
105                 'no_telepon' => $i->post('no_telepon'),
106                 'password' => md5($i->post('password'))
107             );
108             $this->Member_Model->add_member($data);
109             $this->session->set_flashdata('sukses', 'Berhasil menambah
110             member. ');
111             redirect('administrator/members');
112         }
113     }
114 }
115 }
116 }
117 }
118 }
119 }
120 }
121 }
122 }
123 }
124 }
125 }
126 }

```

### 5.2.2.5 Implementasi Source Code Membuat Kelas Event

*Implementasi source code* dapat dilihat pada Tabel 5.95 *Implementasi Source Code* Membuat Jadwal Paket Event dibawah ini.

Nama Kelas : Events

Nama Method : create\_event()

Deskripsi : *Source Code* ini digunakan untuk membuat kelas event. Operasi ini dipanggil saat instruktur menekan tombol tambah event pada halaman view-instruktur.php. *Source Code* ini berisi fungsi method create\_event() yang akan digunakan untuk membuat kelas event. Ketika data yang diinputkan salah maka proses pembuatan kelas event tidak berhasil dibuat, namun jika data yang diinputkan benar maka proses pembuatan kelas event berhasil dan menampilkan alert "Sukses mendaftar Event". Silahkan tunggu konfirmasi admin".

Tabel 5.96 Implementasi Source Code Membuat Jadwal Paket Event

1	function create_event()
2	{
3	
4	
5	\$instruktur_details = \$this->Instruktur_Model->
6	get_detail_instruktur(\$this->session->userdata('id'));
7	\$valid = \$this->form_validation;
8	
9	\$valid->set_rules(
10	
11	'jumlah_sesi',
12	'jumlah_sesi',
13	'required',
14	array( 'required' => 'Anda belum memilih Jumlah
15	Sesi.')
16	);
17	
18	\$valid->set_rules(
19	
20	'jumlah_murid',
21	'jumlah_murid',
22	'required',
23	array( 'required' => 'Anda belum mengisi Jumlah
24	Murid.')
25	);
26	
27	\$valid->set_rules(
28	
29	'durasi',
30	'durasi',
31	'required',
32	array( 'required' => 'Anda belum mengisi Durasi.')
33	);
34	
35	\$valid->set_rules(
36	
37	'harga',
38	'harga',
39	'required is_natural',
40	array( 'required' => 'Anda belum mengisi Harga.')
41	);
42	
43	\$valid->set_rules(
44	
45	'jadwal',
46	'jadwal',
47	'required',
48	
49	
50	
51	
52	
53	
54	
55	
56	

```
57         array( 'required' => 'Anda belum mengisikan Jadwal.')
58     );
59
60     $valid->set_rules(
61         'tempat',
62         'tempat',
63         'required',
64         array( 'required' => 'Anda belum mengisikan Tempat.')
65     );
66
67     $valid->set_rules(
68         'agenda',
69         'agenda',
70         'required',
71         array( 'required' => 'Anda belum mengisikan Agenda.')
72     );
73
74     if ($valid->run()===false)
75     {
76         $data = array('isi' => 'add-event',
77                     'instruktur_details' =>
78 $instruktur_details
79                     );
80
81         $this->load->view("layouts/setting-wrapper", $data,
82 false);
83     }
84     else
85     {
86         $i = $this->input;
87         $data = array(
88             'jumlah_sesi' => $i->
89 >post('jumlah_sesi'),
90             'jumlah_murid' => $i->
91 >post('jumlah_murid'),
92             'durasi' => $i->post('durasi'),
93             'harga' => $i->post('harga'),
94             'jadwal' => $i->post('jadwal'),
95             'tempat' => $i->post('tempat'),
96             'agenda' => $i->post('agenda'),
97             'verified' => 0);
98     }
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
```

```

115         $this->Event_Model->add_event($data);
116
117         $this->session->set_flashdata('sukses', 'Sukses
118 mendaftar Event. Silahkan tunggu konfirmasi admin ');
119         redirect('events/create_event');
120
121     }
122 }
}

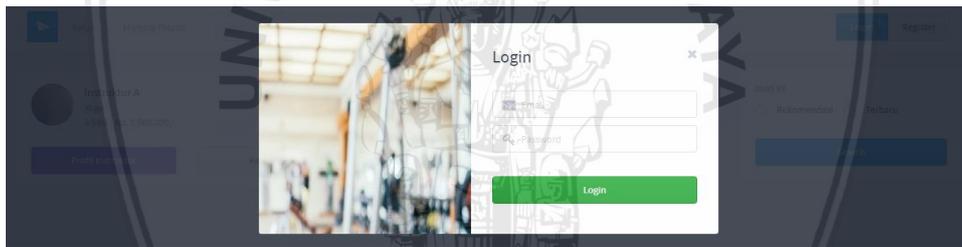
```

### 5.2.3 Implementasi Interface

*Implementasi Interface* berisi *user interface* sistem yang mana dalam pembuatannya mengacu pada perancangan *Interface* sebelumnya.

#### 5.2.3.1 Implementasi Interface Halaman Login Member dan Instruktur

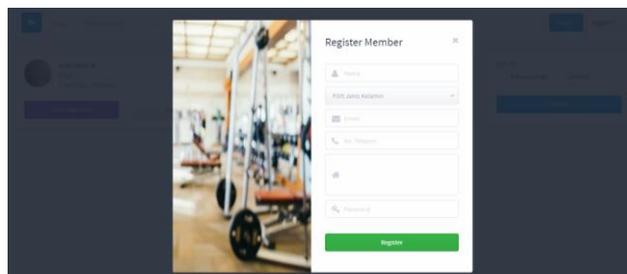
*Implementasi Interface* dapat dilihat pada Gambar 5.80 *Implementasi Interface* Halaman *Login Member* dan Instruktur dibawah ini mengacu pada perancangan *interface* halaman *login member* dan instruktur. Pada halaman ini *Member* dan Instruktur dapat melakukan *login* dengan menekan tombol *login* pada menu utama dahulu lalu mengisi *email* dan *password* lalu klik tombol *login*.



**Gambar 5.81 Implementasi Interface Halaman Login Member dan Instruktur**

#### 5.2.3.2 Implementasi Interface Halaman Register Member

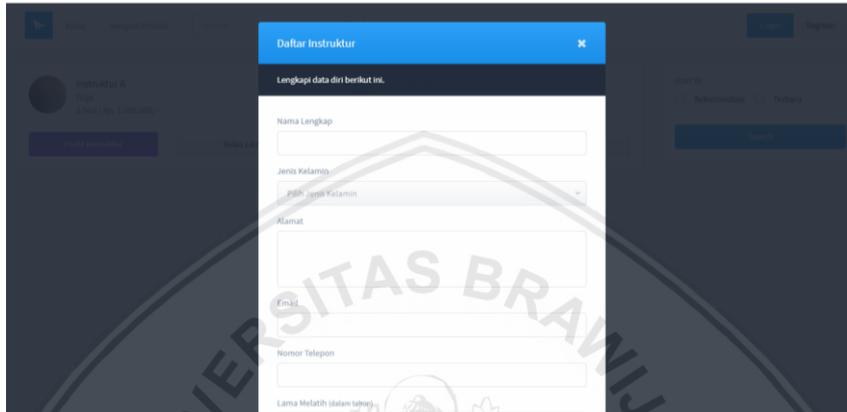
*Implementasi Interface* dapat dilihat pada Gambar 5.82 *Implementasi Interface* Halaman *Register Member* dibawah ini mengacu pada perancangan *interface register member*. Pada halaman ini *Member* dapat registrasi dengan menekan tombol *register* pada menu utama lalu mengisi nama, jenis kelamin, *email*, no telepon, alamat, *password* kemudian klik tombol *register*.



**Gambar 5.83 Implementasi Interface Halaman Register Member**

### 5.2.3.3 Implementasi Interface Halaman Register Instruktur

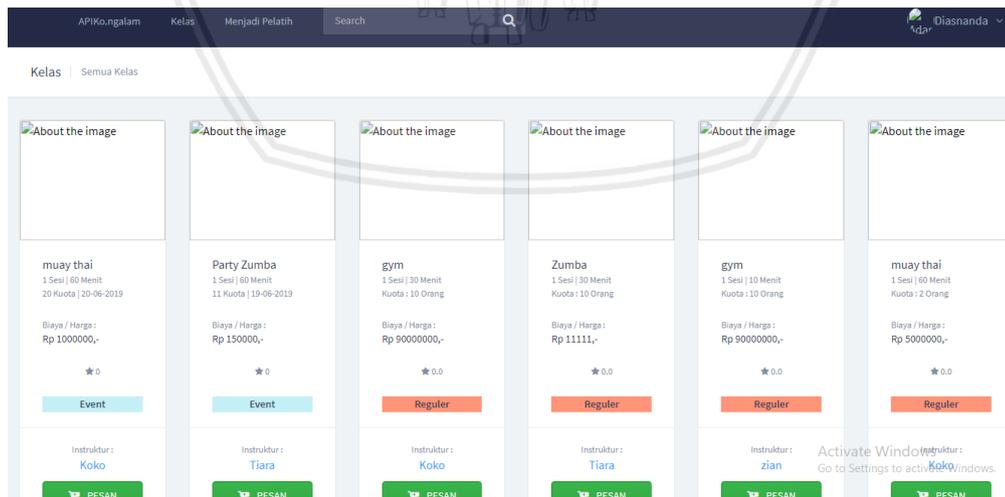
Implementasi Interface dapat dilihat pada Gambar 5.84 Implementasi Interface Halaman Register Instruktur dibawah ini mengacu pada perancangan interface halaman register instruktur. Pada halaman ini Instruktur dapat melakukan register dengan menekan label menjadi pelatih pada menu utama lalu mengisi nama, jenis kelamin, alamat, email, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat dan password lalu klik tombol daftar instruktur.



Gambar 5.85 Implementasi Interface Halaman Register Instruktur

### 5.2.3.4 Implementasi Interface Halaman Utama

Implementasi Interface dapat dilihat pada Gambar 5.86 Implementasi Interface Halaman Utama dibawah ini mengacu pada rancangan interface halaman utama dimana pada halaman utama nanti dapat melihat daftar kelas yang ditawarkan instruktur.



Gambar 5.87 Implementasi Interface Halaman Utama

### 5.2.3.5 Implementasi Interface Halaman Pemesanan Kelas Event

Implementasi Interface dapat dilihat pada Gambar 5.88 Implementasi Interface Halaman Pemesanan Kelas Event dibawah ini mengacu pada perancangan interface halaman pemesanan event yang mana pada halaman pemesanan event ini terdapat detail pemesanan yang berisi informasi instruktur nama dan jenis olahraga instruktur, serta kelas yang berisi durasi jumlah murid, sesi dan harga yang sudah dipilih sebelumnya lalu mengisi form pemesanan berisi nama, no telepon, lokasi lalu klik tombol pesan.

Durasi	Sisa Kuota	Sesi	Harga	Jadwal	Jam
60 Menit	11	1	Rp 150000,-	19-06-2019	09:00

Gambar 5.89 Implementasi Interface Halaman Pemesanan Kelas Event

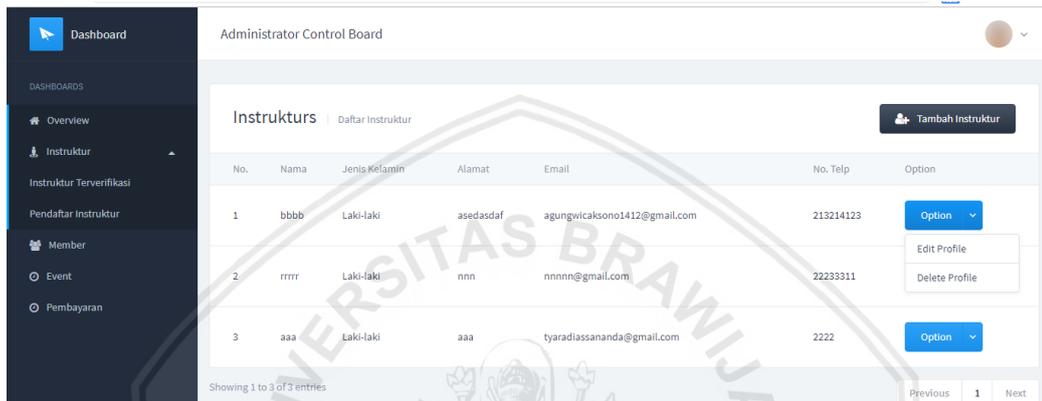
### 5.2.3.6 Implementasi Interface Halaman Ganti Foto

Implementasi Interface dapat dilihat pada Gambar 5.90 Implementasi Interface Halaman Ganti Foto dibawah ini mengacu pada rancangan interface halaman ganti foto yang mana menekan tombol ganti foto lalu akan muncul form ganti foto lalu memilih file foto dan klik upload foto.

Gambar 5.91 Implementasi Interface Halaman Ganti Foto

### 5.2.3.7 Implementasi Interface Halaman Lihat Daftar Instruktur Terverifikasi

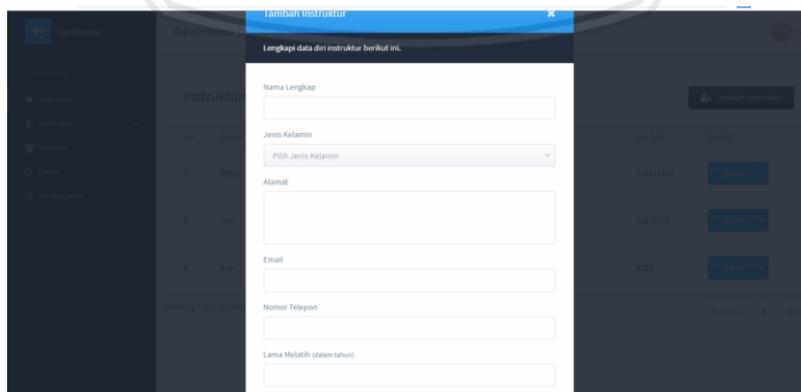
*Implementasi Interface* dapat dilihat pada Gambar 5.92 *Implementasi Interface* Halaman Lihat Daftar Instruktur Terverifikasi dibawah ini mengacu pada rancangan *interface* halaman lihat daftar instruktur terverifikasi yang mana setelah kita menekan menu instruktur dan memilih menu instruktur terverifikasi maka akan menampilkan daftar instruktur yang berisi table no, nama, jenis kelamin, alamat, no telepon dan table option yang terdapat tombol dropdown option yang isinya edit profile untuk mengedit instruktur dan delete profile untuk menghapus instruktur.



Gambar 5.93 Implementasi Interface Halaman Lihat Daftar Instruktur Terverifikasi

### 5.2.3.8 Implementasi Interface Halaman Membuat Instruktur

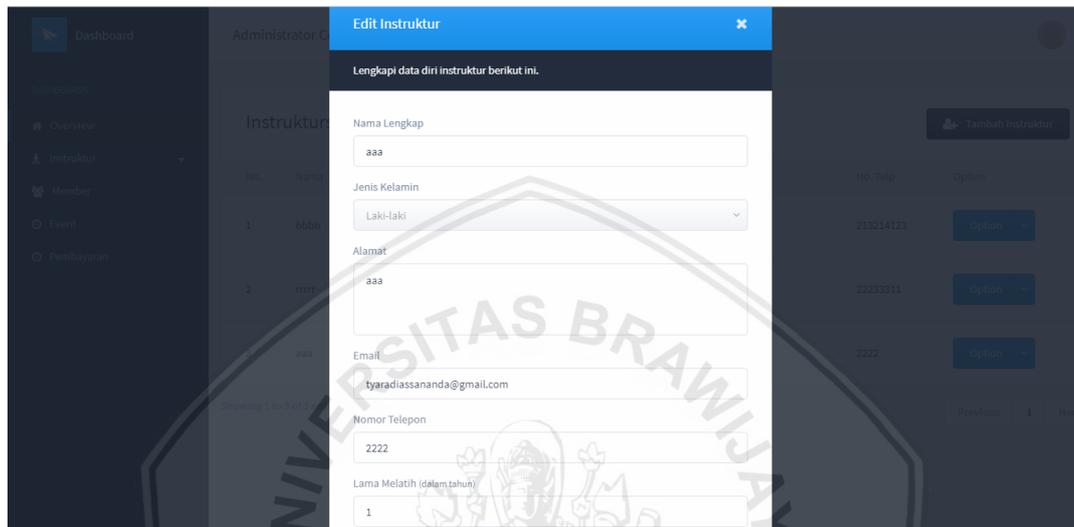
*Implementasi Interface* dapat dilihat pada Gambar 5.94 *Implementasi Interface* Halaman Membuat Instruktur dibawah ini mengacu pada rancangan halaman *interface* membuat instruktur dimana sebelumnya harus menekan tombol tambah instruktur lalu mengisi form tambah instruktur yang berisi nama, jenis kelamin, alamat, email, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat dan password lalu klik tombol tambah instruktur.



Gambar 5.95 Implementasi Interface Halaman Membuat Instruktur

### 5.2.3.9 Implementasi Interface Halaman Edit Instruktur

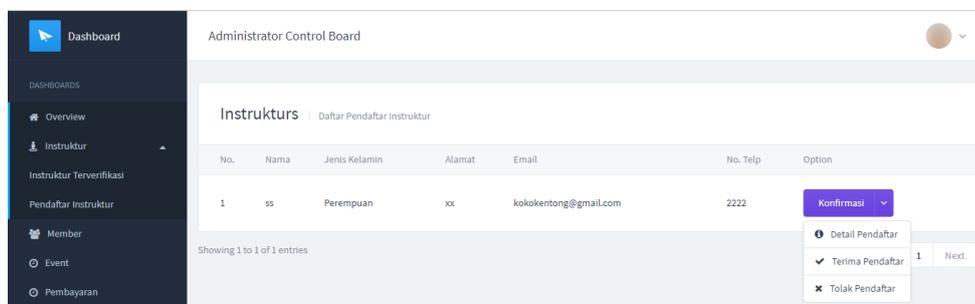
Implementasi Interface pada Gambar 5.96 Implementasi Interface Halaman Edit Instruktur dibawah ini mengacu pada rancangan *interface* halaman edit instruktur dimana sebelumnya harus menekan tombol edit profile lalu mengisi form edit instruktur yang berisi nama, jenis kelamin, alamat, *email*, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat dan *password* lalu klik tombol edit instruktur.



Gambar 5.97 Implementasi Interface Halaman Edit Instruktur

### 5.2.3.10 Implementasi Interface Halaman Lihat Daftar Instruktur Belum Terverifikasi

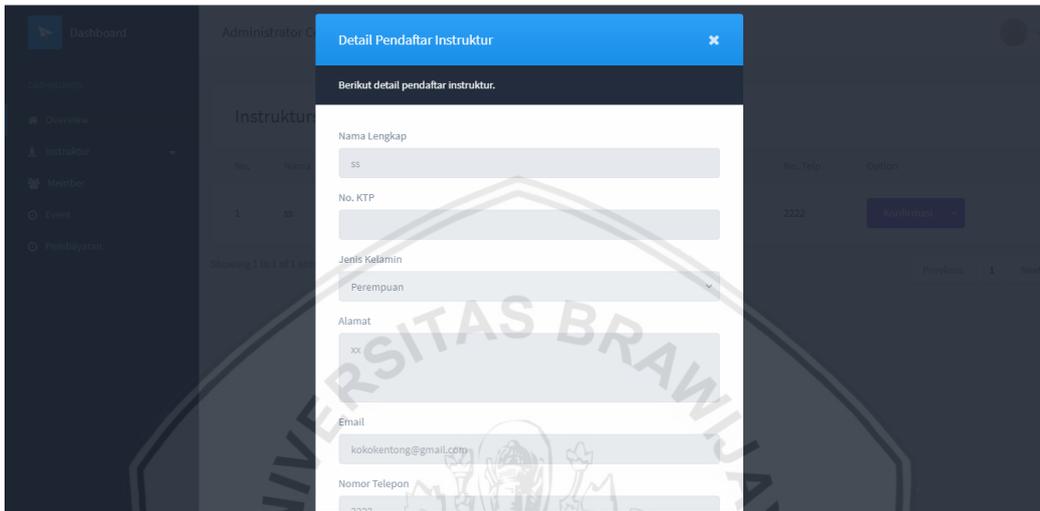
Implementasi Interface dapat dilihat pada Gambar 5.98 Implementasi Interface Halaman Lihat Daftar Pendaftar Instruktur dibawah ini mengacu pada rancangan *interface* halaman lihat daftar instruktur belum terverifikasi yang mana setelah kita menekan menu instruktur dan memilih menu pendaftar instruktur maka akan menampilkan daftar instruktur yang berisi table no, nama, jenis kelamin, alamat, no telepon dan table option yang terdapat tombol dropdown konfirmasi yang isinya detail pendaftar untuk melihat calon instruktur, terima pendaftaran untuk menerima instruktur, tolak instruktur untuk menolak.



Gambar 5.99 Implementasi Interface Halaman Lihat Daftar Pendaftar Instruktur

### 5.2.3.11 Implementasi Interface Halaman Lihat Detail Pendaftar

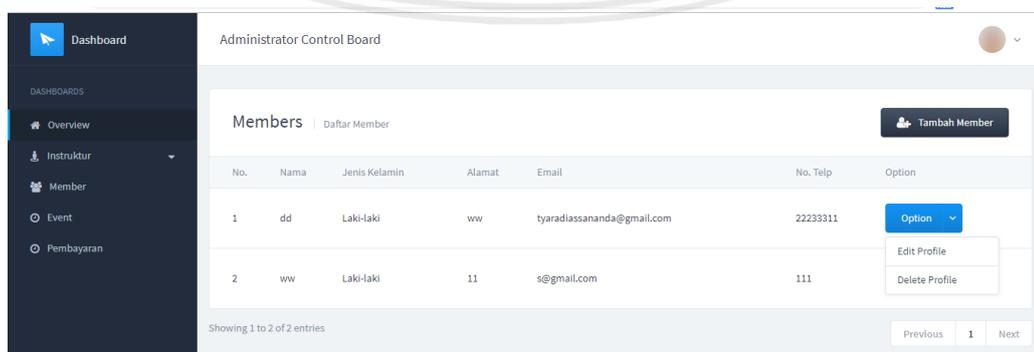
*Implementasi Interface* dapat dilihat pada Gambar 5.100 *Implementasi Interface* Halaman Lihat Detail Pendaftar dibawah ini mengacu pada rancangan *interface* halaman lihat detail pendaftar yang mana setelah kita menekan tombol detail pendaftar maka akan menampilkan detail pendaftar instruktur yang berisi nama, jenis kelamin, alamat, *email*, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat dan *password*.



Gambar 5.101 *Implementasi Interface* Halaman Lihat Detail Pendaftar

### 5.2.3.12 Implementasi Interface Halaman Lihat Daftar Member

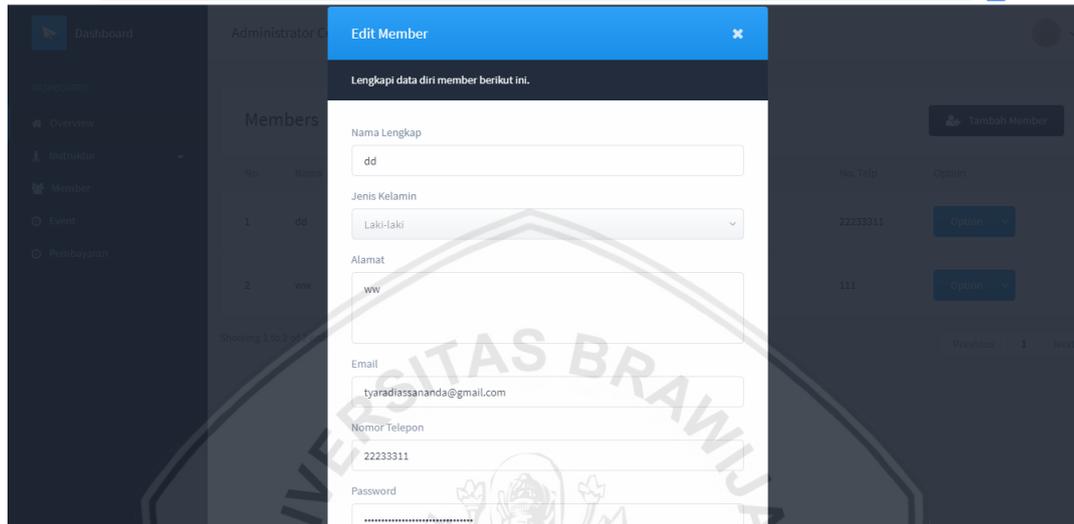
*Implementasi Interface* pada Gambar 5.102 *Implementasi Interface* Halaman Lihat Daftar *Member* dibawah ini mengacu pada rancangan *interface* halaman lihat daftar *member* yang mana setelah kita menekan menu *member* maka akan menampilkan daftar *member* yang berisi table no, nama, jenis kelamin, alamat, no telepon dan table option yang terdapat tombol dropdown option yang isinya edit profile untuk mengedit *member* dan delete profile untuk menghapus *member*.



Gambar 5.103 *Implementasi Interface* Halaman Lihat Daftar *Member*

### 5.2.3.13 Implementasi Interface Halaman Edit Member

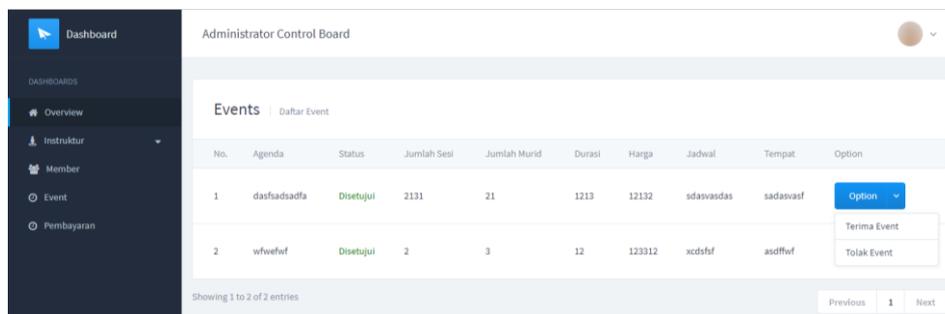
Implementasi Interface dapat dilihat pada Gambar 5.104 Implementasi Interface Halaman Edit Member dibawah ini mengacu pada rancangan interface halaman edit member dimana sebelumnya harus menekan tombol edit profile lalu mengisi form edit member yang berisi nama, jenis kelamin, alamat, email, no telepon dan password lalu klik tombol edit member.



Gambar 5.105 Implementasi Interface Halaman Edit Member

### 5.2.3.14 Implementasi Interface Halaman Lihat Daftar Event

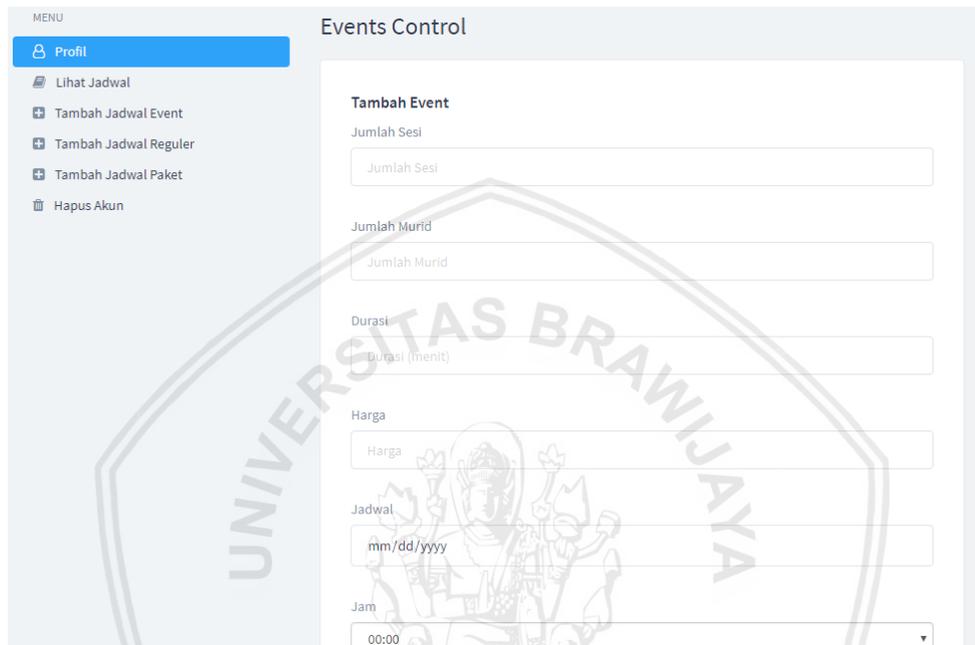
Implementasi Interface dapat dilihat pada Gambar 5.106 Implementasi Interface Halaman Lihat Daftar Event dibawah ini mengacu pada rancangan interface halaman lihat daftar event yang mana ketika menekan menu member akan menampilkan list daftar table event yang berisi no, agenda, status, jumlah sesi, jumlah murid, durasi, harga, jadwal, tempat dan option yang terdapat tombol option yang berisi dropdown terima event untuk menerima event dan tolak event untuk menolak event kemudian jika menekan tombol terima event kolom status pada baris event yang dipilih akan berubah menjadi disetujui, jika belum disetujui statusnya belum disetujui.



Gambar 5.107 Implementasi Interface Halaman Lihat Daftar Event

### 5.2.3.15 Implementasi Interface Halaman Buat Kelas Event

*Implementasi Interface* dapat dilihat pada Gambar 5.108 *Implementasi Interface* Halaman Buat Event dibawah ini mengacu pada rancangan *interface* halaman buat kelas event yang mana ketika menekan menu tambah jadwal event akan menampilkan form agenda, jumlah sesi, jumlah murid, durasi, jam , harga, jadwal, tempat jika menekan tombol tambah event akan memproses pembuatan daftar kelas event.

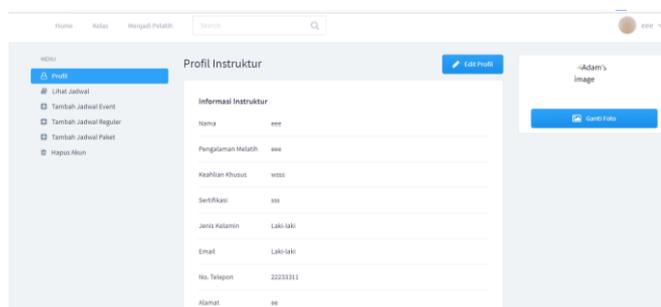


The screenshot shows a web interface titled "Events Control". On the left is a sidebar menu with the following items: "Profil", "Lihat Jadwal", "Tambah Jadwal Event", "Tambah Jadwal Reguler", "Tambah Jadwal Paket", and "Hapus Akun". The main content area is titled "Tambah Event" and contains several input fields: "Jumlah Sesi", "Jumlah Murid", "Durasi (menit)", "Harga", "Jadwal" (with a date format "mm/dd/yyyy"), and "Jam" (with a dropdown menu showing "00:00"). A large watermark of the Universitas Brawijaya logo is overlaid on the form.

**Gambar 5.109 Implementasi Interface Halaman Buat Event**

### 5.2.3.16 Implementasi Interface Halaman Lihat Profil Instruktur

*Implementasi Interface* dapat dilihat pada Gambar 5.110 *Implementasi Interface* Halaman Lihat Profil Instruktur dibawah ini mengacu pada rancangan *interface* halaman lihat profil instruktur yang mana ketika menekan menu profil akan menampilkan nama, jenis kelamin, alamat, *email*, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat. Dan terdapat tombol edit instruktur untuk menampilkan form edit profil instruktur.

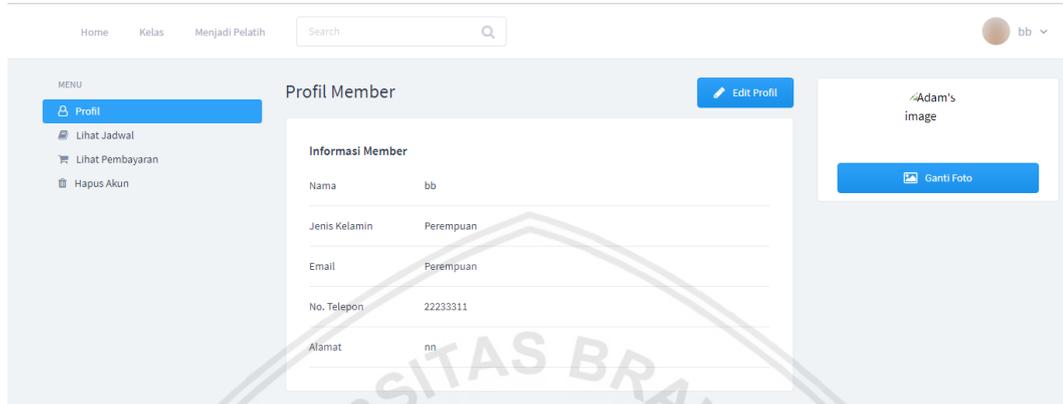


The screenshot shows a web interface titled "Profil Instruktur". On the left is a sidebar menu with the following items: "Profil", "Lihat Jadwal", "Tambah Jadwal Event", "Tambah Jadwal Reguler", "Tambah Jadwal Paket", and "Hapus Akun". The main content area is titled "Informasi Instruktur" and contains several input fields: "Nama", "Pengalaman Melatih", "Keahlian Khusus", "Sertifikasi", "Jenis Kelamin", "Email", "No. Telepon", and "Alamat". There is an "Edit Profil" button in the top right corner. A large watermark of the Universitas Brawijaya logo is overlaid on the form.

**Gambar 5.111 Implementasi Interface Halaman Lihat Profil Instruktur**

### 5.2.3.17 Implementasi Interface Halaman Lihat Profil Member

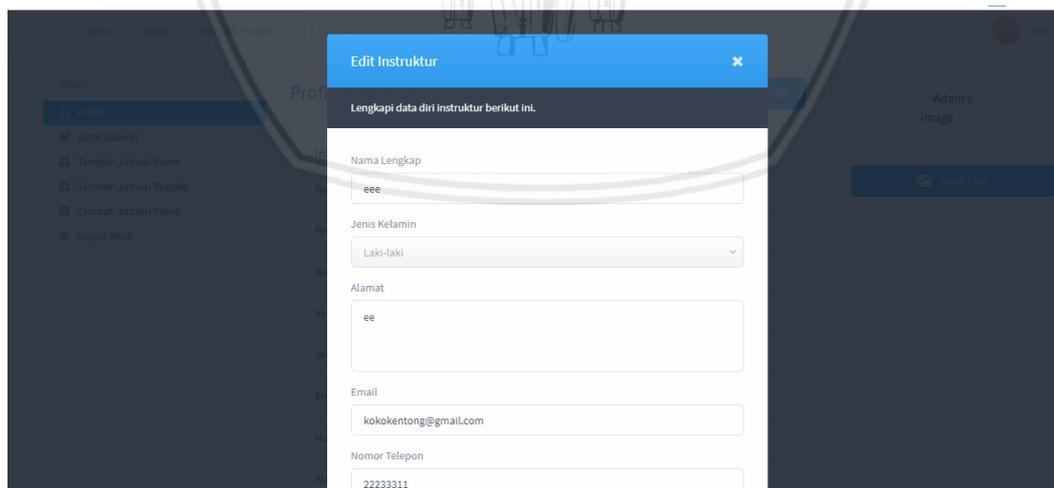
Implementasi Interface dapat dilihat pada Gambar 5.112 Implementasi Interface Halaman Lihat Profil Member dibawah ini mengacu pada rancangan interface halaman lihat profil instruktur yang mana ketika menekan menu profil akan menampilkan nama ,jenis kelamin, alamat, email, no telepon. Dan terdapat tombol edit member untuk menampilkan form edit profil member.



Gambar 5.113 Implementasi Interface Halaman Lihat Profil Member

### 5.2.3.18 Implementasi Interface Halaman Edit Profil Instruktur

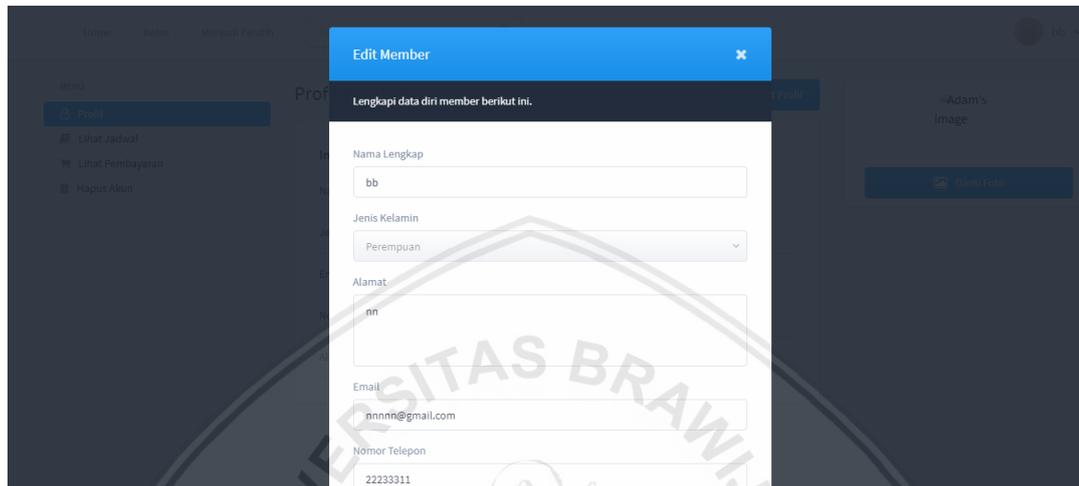
Implementasi Interface dapat dilihat pada Gambar 5.114 Implementasi Interface Halaman Edit Profil Instruktur dibawah ini mengacu pada rancangan interface halaman lihat profil instruktur yang mana ketika menekan tombol edit instruktur akan menampilkan nama, jenis kelamin, alamat, email, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat. Dan terdapat tombol edit instruktur untuk memproses perubahan profil instruktur.



Gambar 5.115 Implementasi Interface Halaman Edit Profil Instruktur

### 5.2.3.19 Implementasi Interface Halaman Edit Profil Member

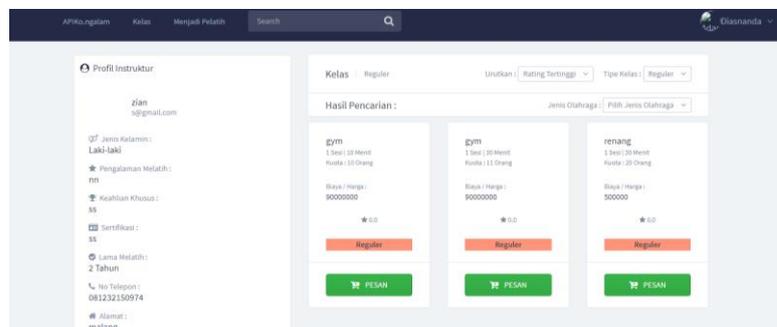
Implementasi Interface dapat dilihat pada Gambar 5.116 Implementasi Interface Halaman Edit Profil Member dibawah ini mengacu pada rancangan interface halaman lihat profil instruktur yang mana ketika menekan tombol edit instruktur akan menampilkan nama, jenis kelamin, alamat, email, no telepon. Dan terdapat tombol edit member untuk memproses perubahan profil member.



Gambar 5.117 Implementasi Interface Halaman Edit Profil Member

### 5.2.3.20 Implementasi Interface Halaman Lihat Detail Profil Instruktur

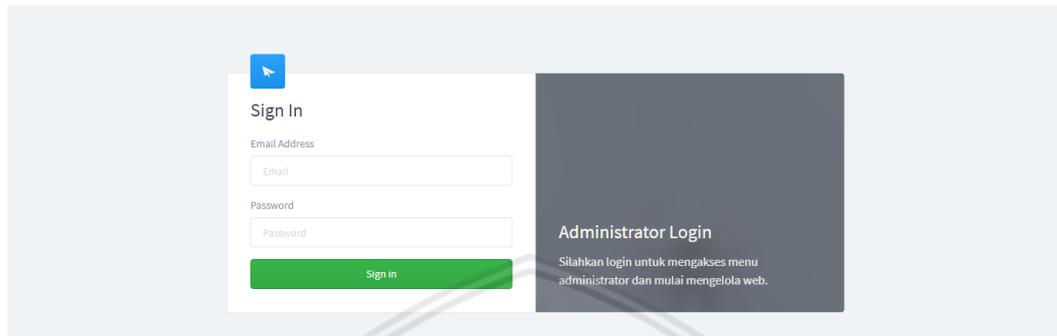
Implementasi Interface dapat dilihat pada Gambar 5.118 Implementasi Interface Halaman Lihat Detail Profil Instruktur dibawah ini mengacu pada rancangan interface halaman lihat detail profil instruktur yang menampilkan nama, jenis kelamin, alamat, email, no telepon, lama melatih, pengalaman melatih, keahlian khusus, sertifikat. Terdapat tombol pesan instruktur untuk menampilkan form pemesanan. Terdapat filter urutan untuk mencari kelas berdasarkan kriteria rating tertinggi, rating terendah, harga tertinggi, harga terendah, sesi terbanyak, sesi tersedikit juga filter tipe kelas yang berisi kelas regular, event, paket dan filter jenis olahraga yang berisi zumba, yoga, muaythai, aerobic, gym, renang.



Gambar 5.119 Implementasi Interface Halaman Lihat Detail Profil Instruktur

### 5.2.3.21 Implementasi Interface Halaman Login Admin

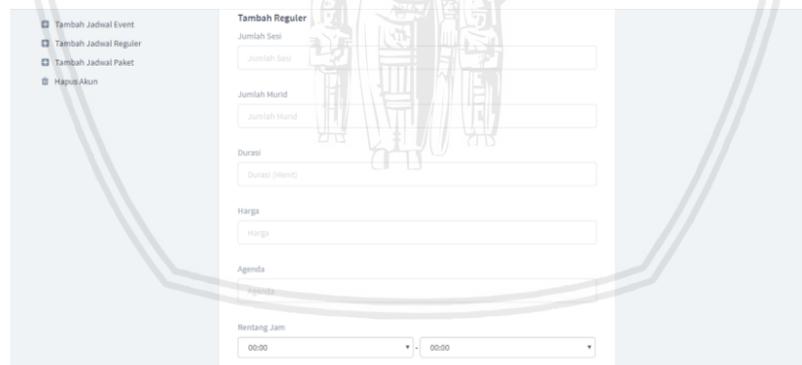
Implementasi Interface dapat dilihat pada Gambar 5.120 Implementasi Interface Halaman Login Admin dibawah ini mengacu pada perancangan interface halaman login admin .Pada halaman ini admin dapat melakukan login dengan mengisi email dan password lalu klik tombol sign in.



Gambar 5.121 Implementasi Interface Halaman Login Admin

### 5.2.3.22 Implementasi Interface Halaman Buat Kelas Reguler

Implementasi Interface dapat dilihat pada Gambar 5.122 Implementasi Interface Halaman Buat Kelas Reguler dibawah ini mengacu pada rancangan interface halaman buat kelas reguler yang mana ketika menekan menu tambah jadwal reguler akan menampilkan form jumlah sesi, jumlah murid, durasi, harga jika menekan tombol tambah reguler akan memproses pembuatan daftar kelas reguler.



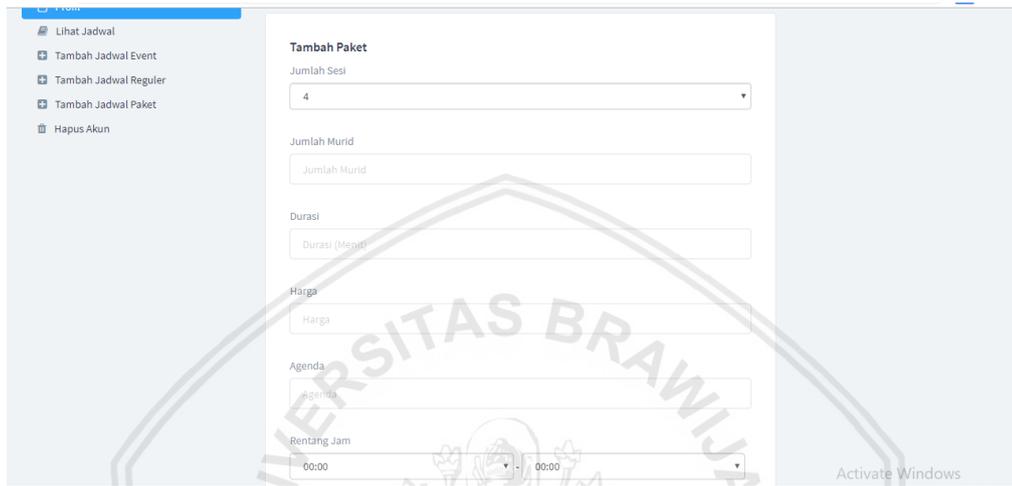
Gambar 5.123 Implementasi Interface Halaman Buat Kelas Reguler

### 5.2.3.23 Implementasi Interface Halaman Buat Kelas Paket

Implementasi Interface dapat dilihat pada Gambar 5.124 Implementasi Interface Halaman Buat Kelas Paket dibawah ini mengacu pada rancangan interface halaman buat kelas paket yang mana ketika menekan menu tambah jadwal paket akan menampilkan form jumlah sesi, jumlah murid, durasi , harga jika menekan tombol tambah paket akan memproses pembuatan daftar kelas paket.

### 5.2.3.23 Implementasi Interface Halaman Buat Kelas Paket

Implementasi Interface dapat dilihat pada Gambar 5.125 Implementasi Interface Halaman Buat Kelas Paket dibawah ini mengacu pada rancangan interface halaman buat kelas paket yang mana ketika menekan menu tambah jadwal paket akan menampilkan form jumlah sesi, jumlah murid, durasi , harga jika menekan tombol tambah paket akan memproses pembuatan daftar kelas paket.



Gambar 5.126 Implementasi Interface Halaman Buat Kelas Paket

### 5.2.3.24 Implementasi Interface Lihat Detail Pembayaran

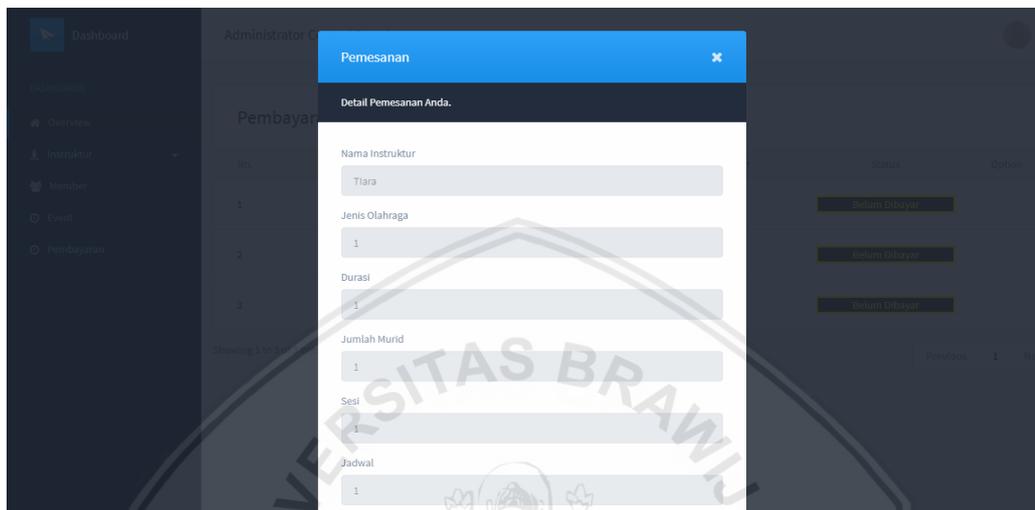
Implementasi Interface pada Gambar 5.127 Implementasi Interface Lihat Detail Pembayaran dibawah ini mengacu pada rancangan interface lihat detail pembayaran yang mana ketika menekan field lihat pada table detail pembayaran akan menampilkan form yang berisi nama bank, kode bayar, jumlah pembayaran dan bukti pembayaran.



Gambar 5.128 Implementasi Interface Lihat Detail Pembayaran

### 5.2.3.25 Implementasi Interface Lihat Detail Pemesanan Member

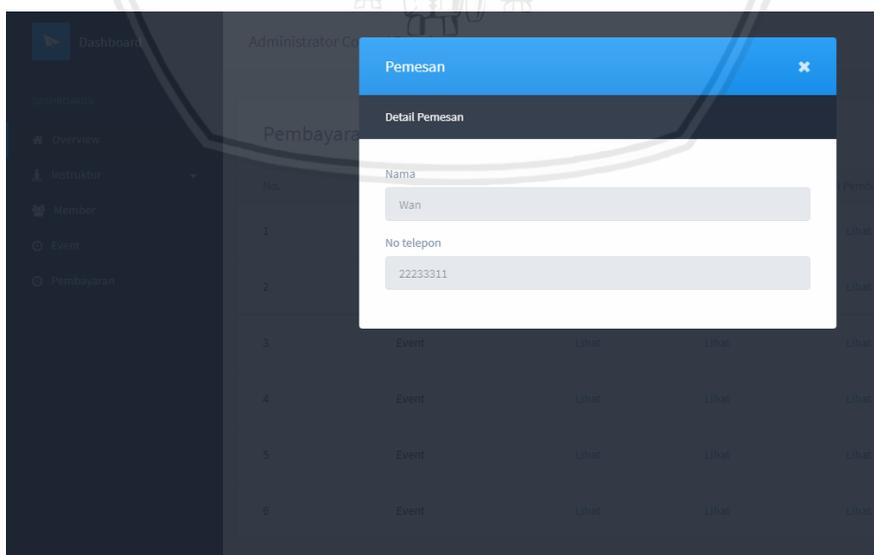
*Implementasi Interface* dapat dilihat pada Gambar 5.129 *Implementasi Interface* Lihat Detail Pemesanan Member dibawah ini mengacu pada rancangan *interface* lihat detail pemesanan member yang mana ketika menekan field lihat pada table detail pemesanan akan menampilkan form yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi, jadwal dan tempat.



**Gambar 5.130 Implementasi Interface Lihat Detail Pemesanan Member**

### 5.2.3.26 Implementasi Interface Lihat Detail Pemesan

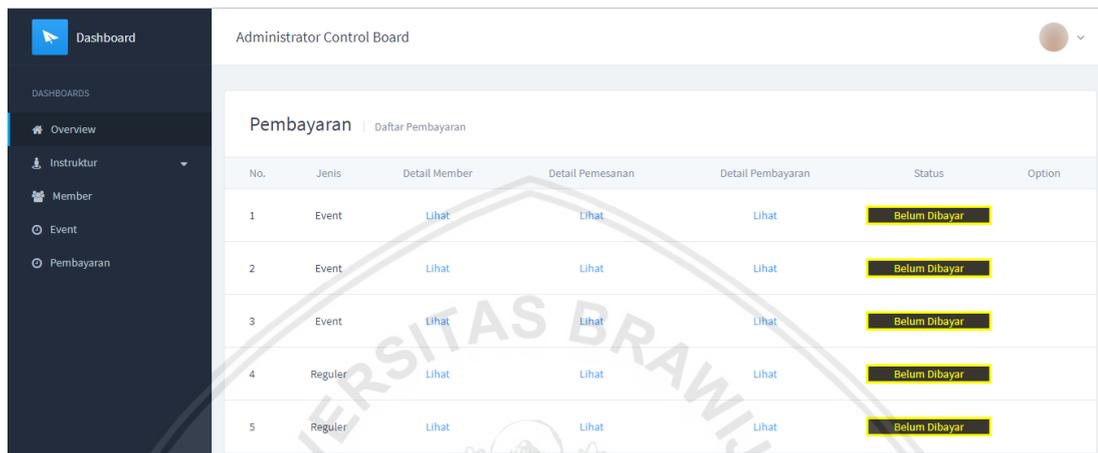
*Implementasi Interface* dapat dilihat pada Gambar 5.131 *Implementasi Interface* Lihat Detail Pemesan dibawah ini mengacu pada rancangan *interface* lihat detail pemesanan yang mana ketika menekan field lihat pada table detail pemesanan akan menampilkan form yang berisi nama pemesan dan no telepon.



**Gambar 5.132 Implementasi Interface Lihat Detail Pemesan**

### 5.2.3.27 Implementasi Interface Lihat Daftar Pembayaran Member

Implementasi Interface dapat dilihat pada Gambar 5.133 Implementasi Interface Lihat Daftar Pembayaran Member dibawah ini mengacu pada rancangan interface lihat daftar pembayaran member yang mana ketika menekan menu pembayaran akan menampilkan table no, detail pemesan, detail pembayaran, status, dan option yang akan menampilkan tombol konfrimasi dan tolak pembayaran.



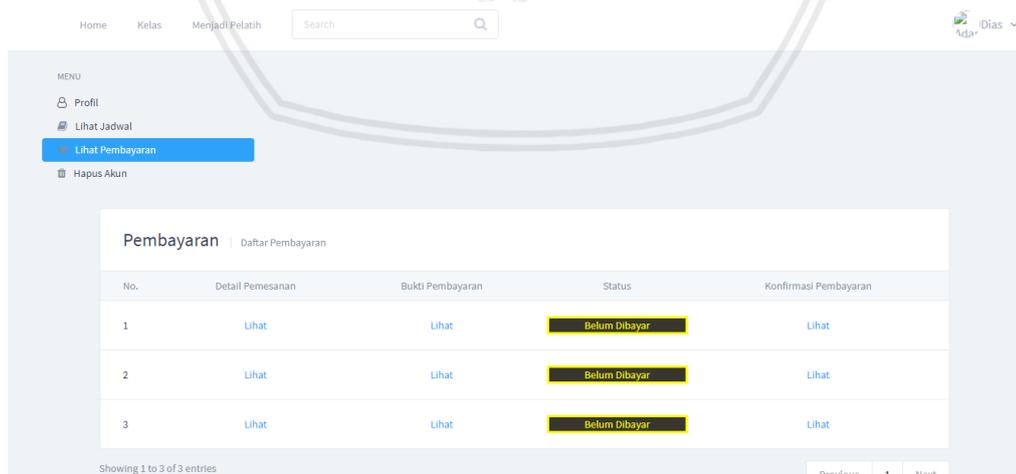
The screenshot shows the 'Administrator Control Board' with a sidebar menu on the left. The main content area is titled 'Pembayaran | Daftar Pembayaran' and contains a table with the following data:

No.	Jenis	Detail Member	Detail Pemesanan	Detail Pembayaran	Status	Option
1	Event	Lihat	Lihat	Lihat	Belum Dibayar	
2	Event	Lihat	Lihat	Lihat	Belum Dibayar	
3	Event	Lihat	Lihat	Lihat	Belum Dibayar	
4	Reguler	Lihat	Lihat	Lihat	Belum Dibayar	
5	Reguler	Lihat	Lihat	Lihat	Belum Dibayar	

Gambar 5.134 Implementasi Interface Lihat Daftar Pembayaran Member

### 5.2.3.28 Implementasi Interface Lihat Daftar Pembayaran

Implementasi Interface dapat dilihat pada Gambar 5.135 Implementasi Interface Lihat Daftar Pembayaran dibawah ini mengacu pada rancangan interface lihat daftar pembayaran yang mana ketika menekan menu pembayaran akan menampilkan table no, detail pemesanan, bukti pembayaran, status, konfirmasi pembayaran.



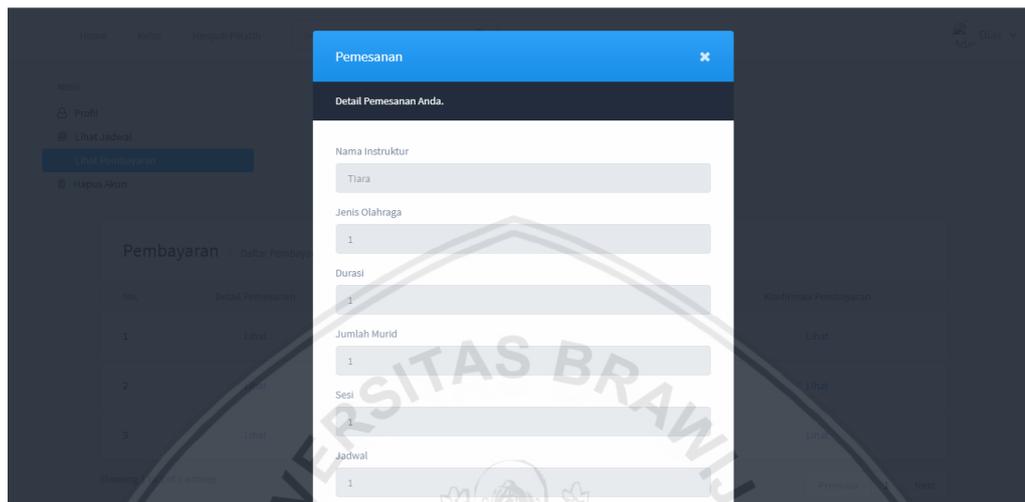
The screenshot shows a user interface with a top navigation bar and a sidebar menu. The main content area is titled 'Pembayaran | Daftar Pembayaran' and contains a table with the following data:

No.	Detail Pemesanan	Bukti Pembayaran	Status	Konfirmasi Pembayaran
1	Lihat	Lihat	Belum Dibayar	Lihat
2	Lihat	Lihat	Belum Dibayar	Lihat
3	Lihat	Lihat	Belum Dibayar	Lihat

Gambar 5.136 Implementasi Interface Lihat Daftar Pembayaran

### 5.2.3.29 Implementasi Interface Lihat Detail Pemesanan

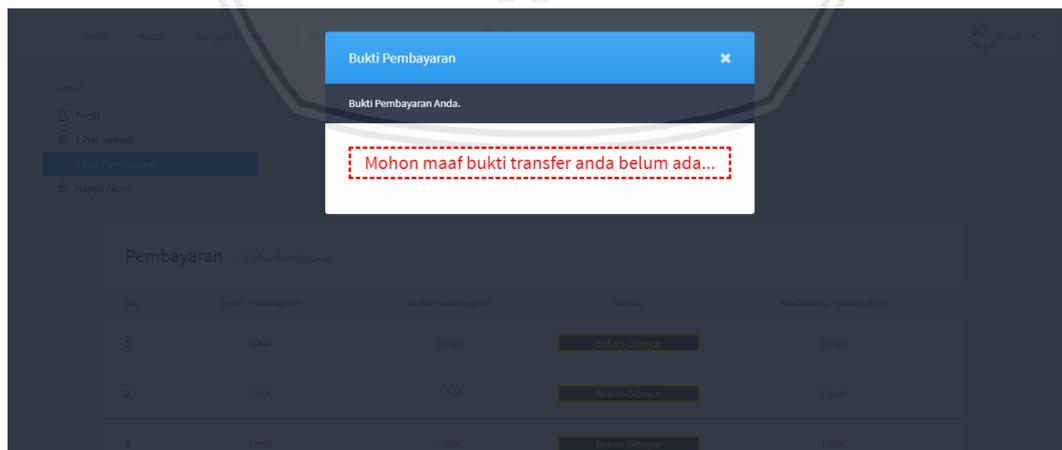
*Implementasi Interface* dapat dilihat pada Gambar 5.137 *Implementasi Interface* Lihat Detail Pemesanan dibawah ini mengacu pada rancangan *interface* lihat detail pemesanan yang mana ketika menekan field lihat pada table detail pemesanan akan menampilkan form yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi, jadwal dan tempat.



**Gambar 5.138 Implementasi Interface** Lihat Detail Pemesanan

### 5.2.3.30 Implementasi Interface Lihat Bukti Pembayaran

*Implementasi Interface* dapat dilihat pada Gambar 5.139 *Implementasi Interface* Lihat Bukti Pembayaran dibawah ini mengacu pada rancangan *interface* lihat bukti pembayaran yang mana ketika menekan field lihat pada table bukti pembayaran akan menampilkan bukti pembayaran jika sudah mengupload bukti pembayarannya jika belum maka bukti transfer belum ditampilkan.



**Gambar 5.140 Implementasi Interface** Lihat Bukti Pembayaran

### 5.2.3.31 Implementasi Interface Konfirmasi Pembayaran

Implementasi Interface dapat dilihat pada Gambar 5.141 Implementasi Interface Konfirmasi Pembayaran dibawah ini mengacu pada rancangan interface konfirmasi pembayaran yang mana ketika menekan field lihat pada table konfirmasi pembayaran akan menampilkan form yang berisi nama bank, kode bayar, jumlah pembayaran, tombol choose file untuk mengupload bukti pembayaran, tombol konfirmasi pembayaran untuk mengkonfirmasi pembayaran dan tombol batalkan pemesanan untuk membatalkan pemesanan.



Gambar 5.142 Implementasi Interface Konfirmasi Pembayaran

### 5.2.3.32 Implementasi Interface Cek Kode Pembayaran

Implementasi Interface dapat dilihat pada Gambar 5.143 Implementasi Interface Cek Kode Pembayaran dibawah ini mengacu pada rancangan interface lihat cek kode pembayaran yang mana ketika menekan tombol pesan maka akan menampilkan form yang berisi nama bank, kode bayar dan jumlah pembayaran.



Gambar 5.144 Implementasi Interface Cek Kode Pembayaran

### 5.2.3.33 Implementasi Interface Halaman Pemesanan Kelas Paket

*Implementasi Interface* dapat dilihat pada Gambar 5.145 *Implementasi Interface* Halaman Pemesanan Kelas Paket dibawah ini mengacu pada perancangan *interface* halaman pemesanan paket yang mana pada halaman pemesanan paket ini terdapat detail pemesanan yang berisi informasi instruktur nama dan jenis olahraga instruktur, serta kelas yang berisi durasi jumlah murid, sesi dan harga yang sudah dipilih sebelumnya lalu mengisi form pemesanan berisi nama, no telepon, lokasi lalu klik tombol pesan.

The screenshot shows a web interface for booking an instructor class. It is divided into two main sections: 'Form Pemesanan' and 'Detail Pemesanan'.

**Form Pemesanan:**

- Nama:** Diasnanda
- No. Telpn:** 22233311
- Pilih Hari:** Dropdown menu showing 'Senin'.
- Jam:** Time selection dropdowns showing '09:00'.
- Lokasi:** A map area with a search box labeled 'Search Box (Wajib diisi)' and a 'Bandit' button.

**Detail Pemesanan:**

**Informasi Instruktur:**

- Nama:** Tiara
- Jenis Olahraga:** aerobik

**Kelas:**

Durasi	Sesi	Harga	Jumlah Murid
Menit	8	Rp 800000,-	4 Orang

**Gambar 5.146 Implementasi Interface Halaman Pemesanan Kelas Paket**

### 5.2.3.34 Implementasi Interface Halaman Pemesanan Kelas Reguler

*Implementasi Interface* dapat dilihat pada Gambar 5.147 *Implementasi Interface* Halaman Pemesanan Kelas Reguler dibawah ini mengacu pada perancangan *interface* halaman pemesanan reguler yang mana pada halaman pemesanan reguler ini terdapat detail pemesanan yang berisi informasi instruktur nama dan jenis olahraga instruktur, serta kelas yang berisi durasi jumlah murid, sesi dan harga yang sudah dipilih sebelumnya lalu mengisi form pemesanan berisi nama, no telepon, lokasi lalu klik tombol pesan.

The screenshot shows a web interface for booking a regular class. It is divided into two main sections: 'Form Pemesanan' and 'Detail Pemesanan'.

**Form Pemesanan:**

- Nama:** Diasnanda
- No. Telpn:** 22233311
- Jadwal:** Input field with 'mm/dd/yyyy' placeholder.
- Jam:** Time selection dropdown showing '08:00'.
- Lokasi:** A map area with a search box labeled 'Search Box'.

**Detail Pemesanan:**

**Informasi Instruktur:**

- Nama:** Dian
- Jenis Olahraga:** gym

**Kelas:**

Durasi	Sesi	Jumlah Murid	Harga
30 Menit	1	11 Orang	Rp 900000,-

**Gambar 5.148 Implementasi Interface Halaman Pemesanan Kelas Reguler**

### 5.2.3.35 Implementasi Interface Halaman Lihat Jadwal Member

Implementasi Interface dapat dilihat pada Gambar 5.149 Implementasi Interface Halaman Lihat Jadwal Member dibawah ini mengacu pada rancangan interface halaman lihat jadwal member dimana sebelumnya harus menekan menu lihat jadwal kemudian akan menampilkan table no, jenis, nama pelatih, sesi, durasi, jam, tempat, rating, agenda, jumlah murid dan terdapat kalender untuk mencari jadwal.

No.	Jam	Jenis	Nama Instruktur	Agenda	Jumlah Murid	Durasi	Sesi	Tempat	Rating
1	08:00	Event	Tiara	Party Zumba	11	60	1	Lihat	★★★★★

Gambar 5.150 Implementasi Interface Halaman Lihat Jadwal Member

### 5.2.3.36 Implementasi Interface Halaman Lihat Jadwal Instruktur Olahraga

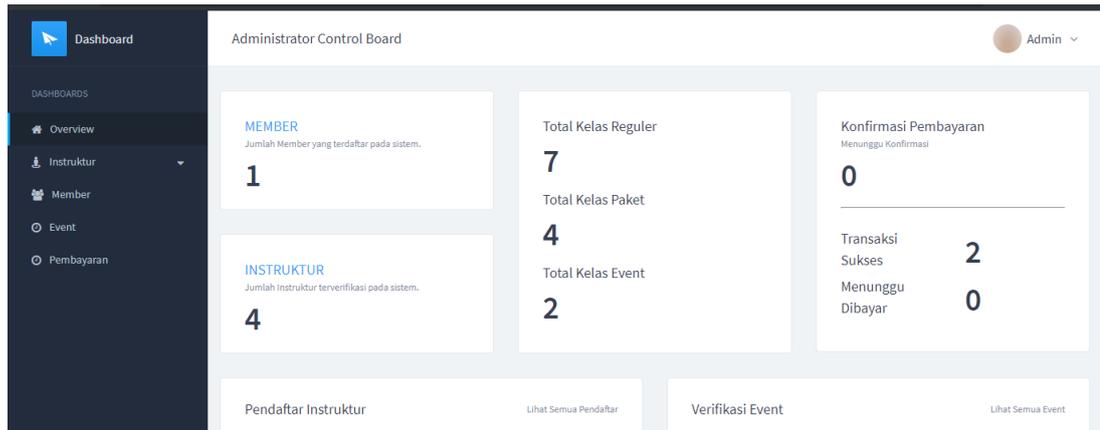
Implementasi Interface dapat dilihat pada Gambar 5.151 Implementasi Interface Halaman Lihat Jadwal Instruktur Olahraga dibawah ini mengacu pada rancangan interface halaman lihat jadwal member dimana sebelumnya harus menekan menu lihat jadwal kemudian akan menampilkan table no, jenis, nama member, sesi, durasi, jam, tempat, rating, agenda, jumlah murid dan terdapat kalender untuk mencari jadwal.

No.	Jam	Jenis	Nama Member	Agenda	Jumlah Murid	Durasi	Sesi	Tempat	Rating
1	09:00	Event	Diasnanda	Party Zumba	11	60	1	Lihat	★★★★★

Gambar 5.152 Implementasi Interface Halaman Lihat Jadwal Instruktur Olahraga

### 5.2.37 Implementasi Interface Halaman Dashboard Admin

Implementasi Interface dapat dilihat pada Gambar 5.153 Implementasi Interface Halaman Dashboard Admin dibawah ini mengacu pada rancangan interface halaman dashboard admin yang mana menampilkan jumlah member, instruktur, total kelas regular, paket, event, total yang menunggu dikonfirmasi pembayarannya, total transaksi yang sukses dikonfirmasi, total transaksi yang belum dibayar, melihat pendaftar instruktur dan melihat event yang perlu diverifikasi.

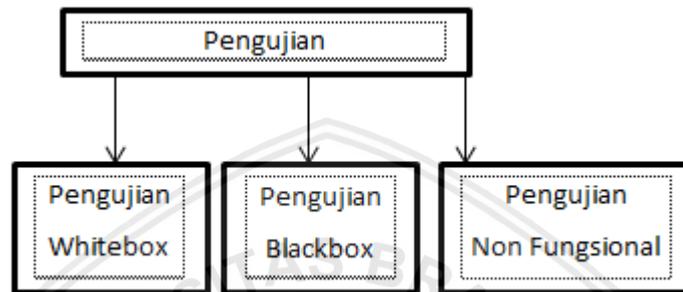


**Gambar 5.154 Implementasi Interface Halaman Dashboard Admin**



## BAB 6 PENGUJIAN

Pengujian dilakukan untuk memperoleh hasil validasi fungsionalitas sistem berjalan sesuai kebutuhan serta untuk memastikan tidak adanya kegagalan pada sistem. Pada penelitian ini pengujian yang dilakukan menggunakan pengujian *blackbox* dan *whitebox testing*. Struktur diagram pengujian dapat dilihat pada Gambar 6.1 Struktur Diagram Pengujian dibawah ini



Gambar 6.2 Struktur Diagram Pengujian

### 6.1 Pengujian *Whitebox Testing*

Pengujian *whitebox testing* dilakukan untuk memastikan komponen *source code* yang sudah diimplementasi berjalan sesuai dengan fungsionalitasnya. Pengujian ini dilakukan dengan melihat unit komponen terkecil dari sebuah program yaitu, kelas yang didalamnya terdapat operasi *methodnya*. Pengujian ini akan dilakukan menggunakan jenis pengujian basis path yang mana akan memilih kode operasi program untuk dicari jalur *independennya* yang mana jalur *independennya* dapat diketahui dari rancangan *flow graphnya*.

#### 6.1.1 Pengujian *Whitebox* Kelas *Order Controller* untuk *method konfirmasi\_pembayaran()*

Pengujian ini didasarkan pada perancangan *algoritma* konfirmasi pembayaran dengan operasi *method* konfirmasi\_pembayaran() pada kelas *Order Controller*.

##### 1. *Pseudocode*

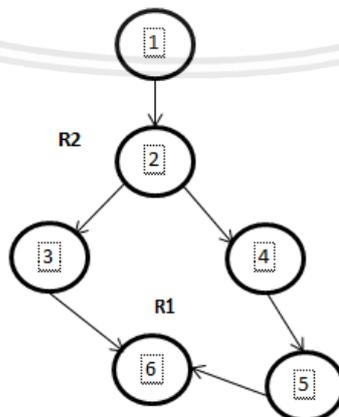
*Pseudocode method* konfirmasi\_pembayaran() dapat dilihat pada Tabel 6.1 *Pseudocode* Pengujian *Whitebox Method* konfirmasi\_pembayaran dan hasil rancangan *flowgraphnya* pada Gambar 6.3 *Flowgraph* method konfirmasi\_pembayaran.

**Tabel 6.2 Pseudocode Pengujian Whitebox Method konfirmasi\_pembayaran**

1	START	
2	Initiate konfigurasi resource image	1
3	Initiate type image png or jpg	
4	Initiate max size 3000	
5	Initiate max width 5000	
6	Initiate max height 5000	
7	Initiate encrypt name true	
8	Set data image	
9	If set data image false	2
10	Set data fail	3
11	Show alert "please select a file"	
12	Else set data image true	4
13	Get array data	5
14	Call compile method konfirmasi pembayaran on order model	
15	Set data succes	
16	Show alert "anda berhasil mengonfirmasi pembayaran. Silahkan tunggu pembayaran anda dikonfirmasi oleh admin"	
17	END	

**2. Basic Path Testing**

**a. Flow Graph**



**Gambar 6.4 Flowgraph method konfirmasi\_pembayaran**

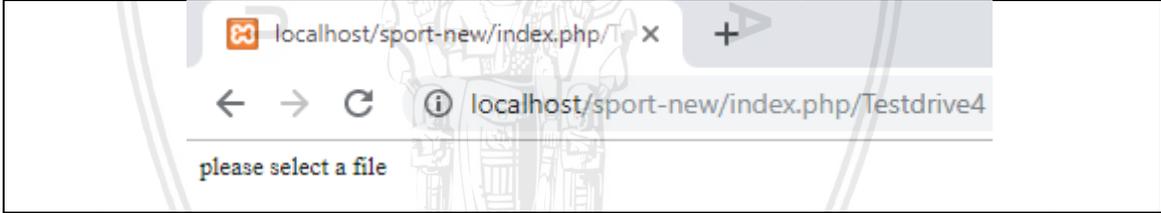
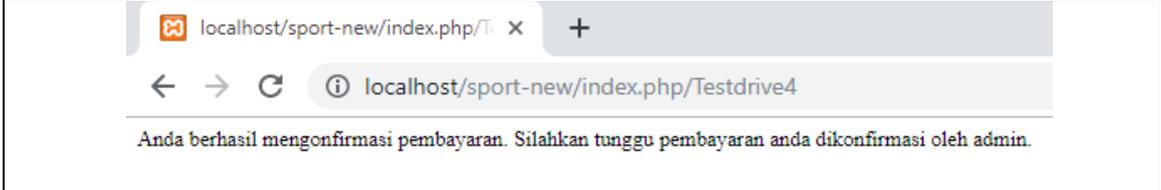
**b. Cyclomatic Complexity**

1.  $V(G) = 2$  region (R1, R2)
2.  $V(G) = 6 \text{ edges} - 6 \text{ nodes} + 2 = 2$
3.  $V(G) = 1 \text{ predicate nodes} + 1 = 2$

**c. Independen Path**

1. Jalur1=1-2-3-6
2. Jalur2=1-2-4-5-6

**3. Test Case**

No	Jalur Uji	Test Case	Expected Result	Result	Status
1	1-2-3-6	Menggunakan kelas driver Testdriver4 dengan menginputkan nilai variable upload = false pada method konfirmasi_pembayaran()	Menampilkan pesan notifikasi "please select a file"	Menampilkan pesan notifikasi "please select a file"	Valid
					
2	1-2-4-5-6	Menggunakan kelas driver Testdriver4 dengan menginputkan nilai variable upload = true pada method konfirmasi_pembayaran()	Menampilkan pesan notifikasi "Anda berhasil mengonfirmasi pembayaran. Silahkan tunggu pembayaran anda dikonfirmasi oleh admin"	Menampilkan pesan notifikasi "anda berhasil mengonfirmasi pembayaran. Silahkan tunggu pembayaran anda dikonfirmasi oleh admin"	Valid
					



### 6.1.2 Pengujian *Whitebox* Kelas *Members Controller* untuk *method add\_member()*

Pengujian ini didasarkan pada perancangan *algoritma* menambahkan *member* dengan operasi *method add\_member()* pada kelas *Members Controller*.

#### 1. *Pseudocode*

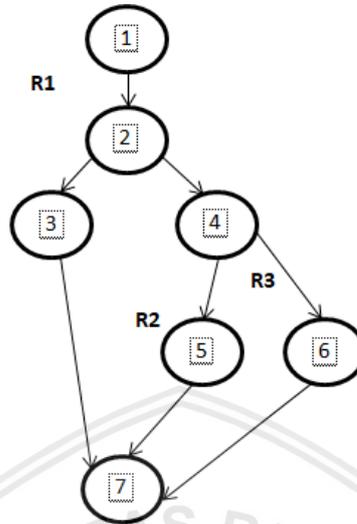
*Pseudocode* dapat dilihat pada Tabel 6.3 *Pseudocode* Pengujian *Whitebox Method add\_member()* dan rancangan *flowgraphnya* pada Gambar 6.5 *Flowgraph method add\_member*.

**Tabel 6.4 *Pseudocode* Pengujian *Whitebox Method add\_member()***

1	START	
2	Get form validation	1
3	Set data nama,jenis kelamin,alamat,email,no telepon,password	
4	If set data false	2
5	Call compile method get all member on member model	
6	Set data fail	3
7	Show alert "lengkapi data"	
8	Else set data email true	
9	Inisialisasi variable num call compile method check email with parameter where on Instruktur Model	4
10	Inisialisasi variable num2 call compile method check email with parameter where on Member Model	
11	If variable num > 0    variable num2 > 0	
12	Show alert "mail yang anda isi telah terdaftar di sistem Silahkan isi dengan email yang belum terdaftar!"	5
13	Else	
14	Set data array nama,jenis kelamin,password,alamat,email,no telepon	6
15	Show alert "Berhasil menambah member"	
16	END	7

## 2. Basic Path Testing

### a. Flow Graph



Gambar 6.6 Flowgraph method add\_member

### b. Cyclomatic Complexity

- $V(G) = 3$  region (R1, R2, R3)
- $V(G) = 8 \text{ edges} - 7 \text{ nodes} + 2 = 3$
- $V(G) = 2 \text{ predicate nodes} + 1 = 3$

### c. Independen Path

1. Jalur1=1-2-3-7
2. Jalur2=1-2-4-5-7
3. Jalur2=1-2-4-6-7

## 3. Test Case

No	Jalur Uji	Test Case	Expected Result	Result	Status
1	1-2-3-6	Menggunakan kelas driver Testdriver2 dengan menginputkan variable run = false pada method add_member()	Menampilkan pesan notifikasi "lengkapi data"	Menampilkan pesan notifikasi "lengkapi data"	valid

2	1-2-4-5-7	Menggunakan kelas driver Testdriver2 dengan menginputkan variable num = 1 dan variable num2 = 1 pada method add_member()	Menampilkan pesan notifikasi "Email yang anda isi telah terdaftar di sistem. Silahkan isi dengan email yang belum terdaftar!"	Menampilkan pesan notifikasi ""Email yang anda isi telah terdaftar di sistem. Silahkan isi dengan email yang belum terdaftar!"	valid
3	1-2-4-6-7	Menggunakan kelas driver Testdriver2 dengan menginputkan variable run = true pada method add_member()	Menampilkan pesan notifikasi "sukses membuat member"	Menampilkan pesan notifikasi "sukses membuat member"	

### 6.1.3 Pengujian *Whitebox* Kelas *Events Controller* untuk *method* `add_event()`

Pengujian ini didasarkan pada perancangan *algoritma* membuat kelas event dengan operasi *method* `add_event()` pada kelas *Events Controller*.

#### 1. *Pseudocode*

*Pseudocode* dapat dilihat pada Tabel 6.5 *Pseudocode* Pengujian *Whitebox Method* `add_event()` dan rancangan *flowgraphnya* pada Gambar 6.7 *Flowgraph method* `add_event`.

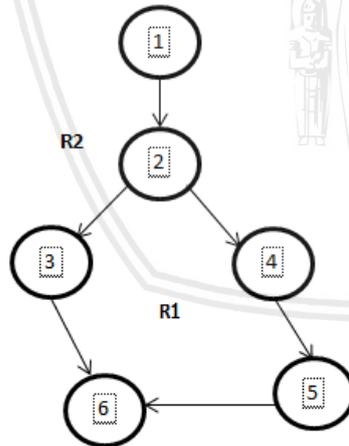


**Tabel 6.6 Pseudocode Pengujian Whitebox Method add\_event()**

1	START
2	Get form validation
3	Set data nama event, nama instruktur, sesi, durasi, jumlah murid, harga <b>1</b>
4	If set data false <b>2</b>
5	Call compile method get all event on event model
6	Set data fail
7	Show message "lengkapi data" <b>3</b>
8	Else set data true <b>4</b>
9	Get array data
10	Call compile method add event on event model
11	Set data succes
12	Show message "sukses mendaftar event" <b>5</b>
13	END <b>6</b>

**2. Basic Path Testing**

**a. Flow Graph**



**Gambar 6.8 Flowgraph method add\_event**

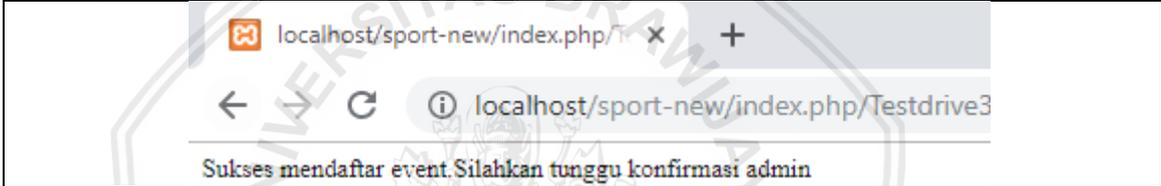
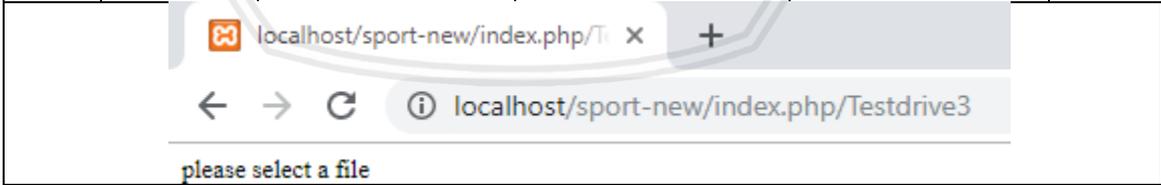
**b. Cyclomatic Complexity**

- $V(G) = 2$  region (R1, R2)
- $V(G) = 6 \text{ edges} - 6 \text{ nodes} + 2 = 2$
- $V(G) = 1 \text{ predicate nodes} + 1 = 2$

**c. Independen Path**

1. Jalur1=1-2-3-6
2. Jalur2=1-2-4-5-6

**3. Test Case**

No	Jalur Uji	Test Case	Expected Result	Result	Status
1	1-2-3-6	Menggunakan kelas driver Testdrive3 dengan menginputkan nilai kembalian false pada method add_event()	Menampilkan pesan notifikasi "please select a file"	Menampilkan pesan notifikasi "please select a file"	valid
					
2	1-2-4-5-6	Menggunakan kelas driver Testdrive3 dengan menginputkan nilai kembalian true pada method add_event()	Menampilkan pesan notifikasi "sukses membuat event"	Menampilkan pesan notifikasi "sukses membuat event"	valid
					

**6.1.4 Pengujian Whitebox Kelas Order Controller untuk method confirm\_instruktur()**

Pengujian ini didasarkan pada perancangan *algoritma* melakukan konfirmasi pendaftaran instruktur dengan operasi *method* confirm\_instruktur() pada kelas *Order Controller*.



### 1.Pseudocode

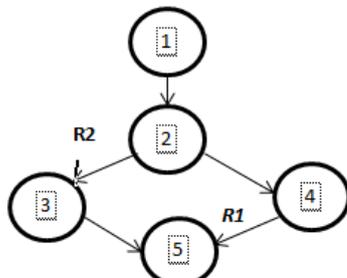
Pseudocode dapat dilihat pada Tabel 6.7 Pseudocode Pengujian Whitebox Method confirm\_instruktur() dan rancangan flowgraphnya pada Gambar 6.9 Flowgraph method confirm\_instruktur.

**Tabel 6.8 Pseudocode Pengujian Whitebox Method confirm\_instruktur()**

1	STRAT	
2	Get data array	
3	Id = id	
4	Verified = 1	
5	Call compile method get detail instruktur id on instruktur model	
6	Call compile method edit instruktur data on instruktur model	
7	Set configuration email	1
8	Call library email	
9	Set email respons	
10	If email send succes	2
11	Notification success	
12	Show message "Hi 'X' , Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima,Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima, Silahkan login menggunakan email dan password yang sudah terdaftar untuk melakukan perubahan profil, pembuatan kelas dan event"	3
13	Else email send fail	
14	Notification failed	4
15	END	5

### 2.Basic Path Testing

#### a. Flow Graph



**Gambar 6.10 Flowgraph method confirm\_instruktur**

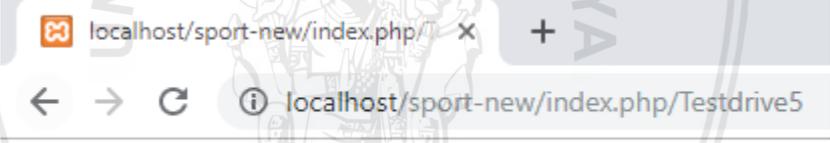
**b.Cyclomatic Complexity**

1.  $V(G) = 2$  region (R1, R2)
2.  $V(G) = 5 \text{ edges} - 5 \text{ nodes} + 2 = 2$
3.  $V(G) = 1 \text{ predicate nodes} + 1 = 2$

**c.Independen Path**

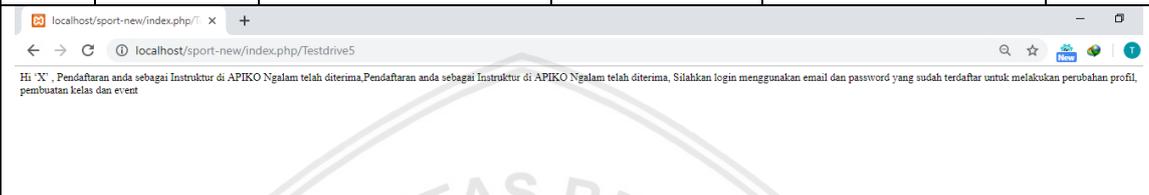
1. Jalur1=1-2-3-5
2. Jalur2=1-2-4-5

**3.Test Case**

No	Jalur Uji	Test Case	Expected Result	Result	Status
1	1-2-3-5	Menggunakan kelas driver Testdrive5 dengan menginputkan nilai variable send = false	Notifikasi tidak terkirim pada email	Notifikasi tidak terkirim pada email	Valid
					
2	1-2-4-5	Menggunakan kelas driver Testdrive5 dengan menginputkan nilai variable send =true	Notifikasi terkirim pada email dan menampilkan pesan "Hi 'X' , Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima,Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima,Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima,Silahkan login menggunakan	Notifikasi terkirim pada email dan menampilkan pesan "Hi 'X' , Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima,Pendaftaran anda sebagai Instruktur di APIKO Ngalam telah diterima, Silahkan login menggunakan email dan password yang sudah terdaftar untuk melakukan perubahan profil,	Valid



			email dan password yang sudah terdaftar untuk melakukan perubahan profil, pembuatan kelas dan event”	pembuatan kelas dan event”	
--	--	--	--	----------------------------	--



### 6.1.5 Pengujian *Whitebox* Kelas *Order Controller* untuk *method* *pesen\_reguler()*

Pengujian ini didasarkan pada perancangan *algoritma* melakukan pemesanan reguler dengan operasi *method* *pesen\_reguler()* pada kelas *Order Controller*.

#### 1. *Pseudocode*

*Pseudocode* dapat dilihat pada Tabel 6.9 *Pseudocode* Pengujian *Whitebox Method* *pesan\_reguler ()* dan *flowgraphnya* pada Gambar 6.11 *Flowgraph method* *pesan\_reguler*.

**Tabel 6.10 *Pseudocode* Pengujian *Whitebox Method* *pesan\_reguler ()***

1	STRAT	
2	i=inputan	
3	variable cek = array	
4	Inisialisasi id instruktur get form field	1
5	Id-instruktur	
6	Inisialisasi jadwal get from field jadwal	
7	Inisialisai jam get from field jam	
8	Inisialisasi varibael num get form method cek-order-reguler in class Order-Model	
9	If variable num > 0	2
10	Set data fail	
11	Show message "Tanggal-jadwal-jam- Penuh"	3
12	Else	4



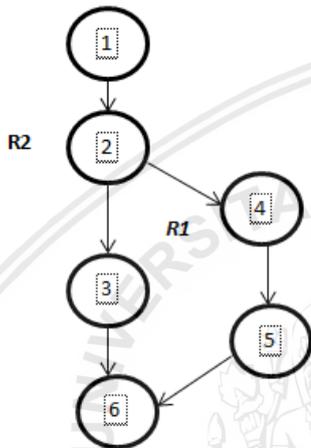
```

13
14   Get data array
    Insialisasi id-pemesanan get form field method pesan-
15 regular parameter data in method Order-Model
16   Set data success
    Show message "Sukses memesan regular"
17
END

```

**2. Basic Path Testing**

**a. Flow Graph**



Gambar 6.12 Flowgraph method pesan\_reguler

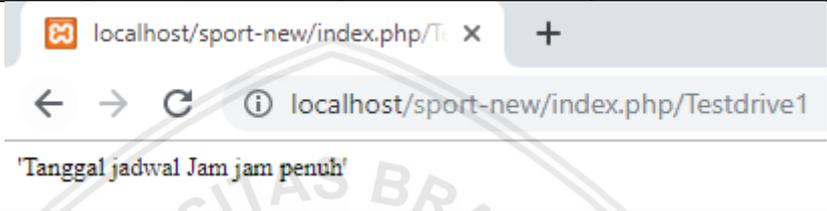
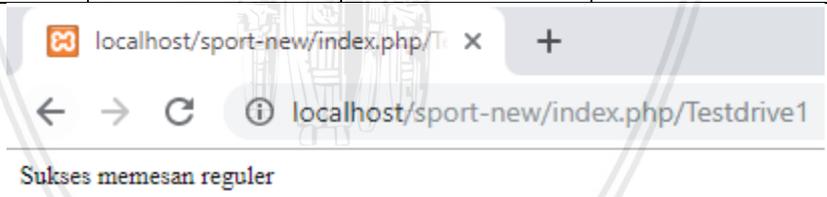
**b. Cyclomatic Complexity**

1.  $V(G) = 2$  region (R1, R2)
2.  $V(G) = 6 \text{ edges} - 6 \text{ nodes} + 2 = 2$
3.  $V(G) = 1 \text{ predicate nodes} + 1 = 2$

**c. Independen Path**

1. Jalur1=1-2-3-6
2. Jalur2=1-2-4-5-6

### 3. Test Case

No	Jalur Uji	Test Case	Expected Result	Result	Status
1	1-2-3-6	Menggunakan kelas driver Testdriver1 dengan menginputkan variable num = 1 pada method pesan_reguler()	Menampilkan pesan notifikasi "Tanggal-jadwal-jam Penuh"	Menampilkan pesan notifikasi "Tanggal-jadwal-jam Penuh"	valid
					
2	1-2-4-5-6	Menggunakan kelas driver Testdriver1 dengan menginputkan variable num = 0 pada method pesan_reguler()	Menampilkan pesan notifikasi "sukses memesan reguler"	Menampilkan pesan notifikasi "sukses memesan reguler"	valid
					

#### 6.1.6 Hasil Analisis Pengujian Whitebox Testing

Dari Pengujian 5 sampel method yang sudah diuji dengan menggunakan metode *cyclomatic complexity* maka didapatkan hasil 2,3,2,2,2 sehingga dapat disimpulkan bahwa program memiliki probabilitas yang rendah terhadap *error*.

#### 6.2 Pengujian Blackbox Testing

Pengujian *blackbox testing* dilakukan untuk memvalidasi keseluruhan fungsi pada sistem yang berjalan sudah sesuai dengan kebutuhan yang sudah didefinisikan pada analisis kebutuhan sebelumnya sebelumnya. Pengujian *blackbox testing* dilakukan dengan cara melihat output dari sistem berdasarkan inputan tertentu. Pengujian *blackbox* didasarkan pada *use case scenario* yang telah dibuat sebelumnya.

### 6.2.1 Pengujian *Blackbox* Fungsi Register Member

Pengujian *blackbox* testing fungsi register member ini didasarkan pada use case scenario Guest Register sebagai Member. Hasil pengujiannya dapat dilihat pada Tabel 6.11 Pengujian *Blackbox* Fungsi Register Member dan hasil pengujian alternatifnya pada Tabel 6.12 Pengujian *Blackbox* Fungsi Register Member dan Tabel 6.13 Pengujian *Blackbox* Fungsi Register Member.

**Tabel 6.14 Pengujian *Blackbox* Fungsi Register Member**

<b>Nama Kasus Uji</b>	Register Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “Daftar”</li> <li>3. Mengisi seluruh field pada form yang tersedia berisi nama lengkap, alamat email, jenis kelamin, no telepon, password.</li> <li>4. Menekan tombol “daftar”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Sukses Mendaftar Member”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Sukses Mendaftar Member”
<b>Status</b>	Valid

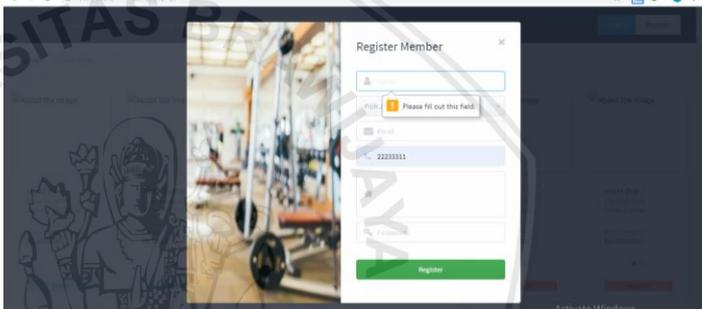
#### a. Kasus Uji Alternatif 1

**Tabel 6.15 Pengujian *Blackbox* Fungsi Register Member**

<b>Nama Kasus Uji</b>	Register Member Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “Daftar”</li> <li>3. Mengisi seluruh field pada form yang tersedia berisi nama lengkap, alamat email, jenis kelamin, no telepon, password.</li> <li>4. Mengisi alamat email yang sudah terdaftar</li> <li>5. Menekan tombol “daftar”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Status</b>	Valid
<b>Screenshoot</b>	

**b.Kasus Uji Alternatif2**

**Tabel 6.16 Pengujian *Blackbox* Fungsi *Register Member***

<b>Nama Kasus Uji</b>	Register Member Alternatif 2
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1.Membuka aplikasi apiko</li> <li>2.Menekan tombol “Daftar”</li> <li>3.Tidak mengisi seluruh field pada form yang tersedia berisi nama lengkap,alamat email,jenis kelamin,no telepon,password.</li> <li>4.Menekan tombol “daftar”</li> </ol>
<b>Hasil yang Diharapkan</b>	1.Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.2 Pengujian *Blackbox* Fungsi *Register Instruktur Olahraga***

*Pengujian blackbox testing fungsi register instruktur olahraga ini didasarkan pada use case scenario Guest Register sebagai Instruktur Olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.17 Pengujian Blackbox Fungsi Register Instruktur Olahraga dan hasil pengujian alternatifnya pada Tabel 6.18 Pengujian Blackbox Fungsi Register Instruktur Olahraga dan Tabel 6.19 Pengujian Blackbox Fungsi Register Instruktur Olahraga*

**Tabel 6.20 Pengujian *Blackbox* Fungsi *Register Instruktur Olahraga***

<b>Nama Kasus Uji</b>	Register Instruktur Olahraga
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “Menjadi Pelatih”</li> <li>3. Mengisi seluruh field pada form yang tersedia berisi nama lengkap,alamat email,jenis kelamin,no telepon,alamat,lamamelatih,pengalaman melatih,keahlian lain,sertifikasi,password,</li> <li>4. Menekan tombol “daftar instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan “Sukses Mendaftar



	Instruktur,Tunggu Konfirmasi dari Admin”
<b>Hasil</b>	1. Aplikasi menampilkan “Sukses Mendaftar Instruktur,Tunggu Konfirmasi dari Admin”
<b>Status</b>	Valid

**a.Kasus Uji Alternatif1**

**Tabel 6.21 Pengujian *Blackbox* Fungsi *Register* Instruktur Olahraga**

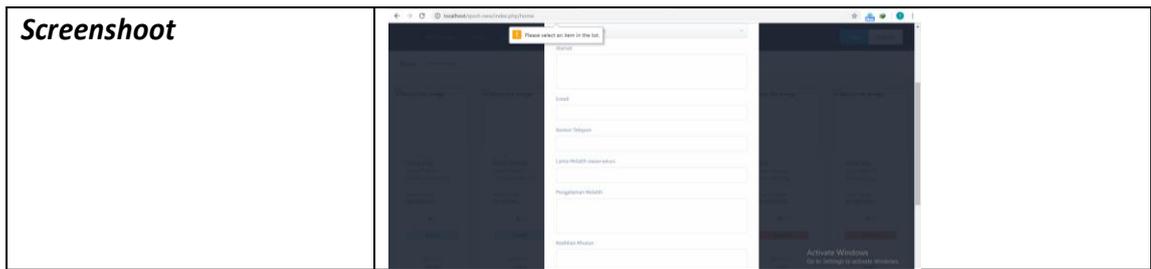
<b>Nama Kasus Uji</b>	Register Instruktur Olahraga Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “Menjadi Pelatih”</li> <li>3. Mengisi seluruh field pada form yang tersedia berisi nama lengkap,alamat email,jenis kelamin,no telepon,alamat,lamamelatih,pengalaman melatih,keahlian lain,sertifikasi,password,</li> <li>4. Mengisi email yang sudah terdaftar</li> <li>5. Menekan tombol “daftar instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Status</b>	Valid
<b>Screenshot</b>	

**b.Kasus Uji Alternatif2**

**Tabel 6.22 Pengujian *Blackbox* Fungsi *Register* Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Register Instruktur Olahraga Alternatif 2
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1.Membuka aplikasi apiko</li> <li>2.Menekan menu “Menjadi Pelatih”</li> <li>3.Tidak mengisi seluruh field pada form yang tersedia berisi nama lengkap,alamat email,jenis kelamin,no telepon,alamat,lamamelatih,pengalaman melatih,keahlian lain,sertifikasi,password,</li> <li>4.Menekan tombol “daftar instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid





### 6.2.3 Pengujian *Blackbox* Fungsi Login

Pengujian *blackbox* testing fungsi login ini didasarkan pada use case scenario Login. Hasil pengujiannya dapat dilihat pada Tabel 6.23 Pengujian *Blackbox* Fungsi Login dan hasil pengujian alternatifnya pada Tabel 6.24 Pengujian *Blackbox* Fungsi Login dan Tabel 6.25 Pengujian *Blackbox* Fungsi Login.

**Tabel 6.26 Pengujian *Blackbox* Fungsi Login**

<b>Nama Kasus Uji</b>	Login
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “Login”</li> <li>3. Mengisi seluruh field pada form yang tersedia berisi alamat email dan password</li> <li>4. Menekan tombol “login”</li> </ol>
<b>Hasil yang Diharapkan</b>	Aplikasi menampilkan halaman utama
<b>Hasil</b>	Aplikasi menampilkan halaman utama
<b>Status</b>	Valid

**a. Kasus Uji Alternatif 1**

**Tabel 6.27 Pengujian *Blackbox* Fungsi Login**

<b>Nama Kasus Uji</b>	Login Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “Login”</li> <li>3. Tidak Mengisi seluruh field pada form yang tersedia berisi alamat email dan password</li> <li>4. Menekan tombol “login”</li> </ol>
<b>Hasil yang Diharapkan</b>	Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	



**b. Kasus Uji Alternatif2**

**Tabel 6.28 Pengujian *Blackbox* Fungsi Login**

<b>Nama Kasus Uji</b>	Login Alternatif 1
<b>Prosedur</b>	1.Membuka aplikasi apiko 2.Menekan tombol “Login” 3.Memasukkan password dan email salah atau belum terdaftar 4.Menekan tombol “login”
<b>Hasil yang Diharapkan</b>	Menampilkan alert “Akun Tidak Terdaftar / Password Salah Pastikan Email & Password sudah benar”
<b>Hasil</b>	Menampilkan alert “Akun Tidak Terdaftar / Password Salah Pastikan Email & Password sudah benar”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.4 Pengujian *Blackbox* Fungsi Mencari Instruktur Olahraga**

*Pengujian blackbox* testing fungsi mencari instruktur olahraga ini didasarkan pada use case scenario mencari instruktur olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.29 Pengujian *Blackbox* Fungsi Mencari Instruktur Olahraga dan hasil pengujian alternatifnya pada Tabel 6.30 Pengujian *Blackbox* Fungsi Mencari Instruktur Olahraga.

**Tabel 6.31 Pengujian *Blackbox* Fungsi Mencari Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Mencari Instruktur Olahraga
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Memasukkan jenis olahraga pada field pencarian
<b>Hasil yang Diharapkan</b>	Aplikasi menampilkan daftar instruktur olahraga yang melatih jenis olahraga dan nama instruktur olahraga yang diinputkan
<b>Hasil</b>	Aplikasi menampilkan daftar instruktur olahraga yang melatih jenis olahraga dan nama instruktur olahraga yang diinputkan
<b>Status</b>	Valid

**a.Kasus Uji Alternatif1**

**Tabel 6.32 Pengujian *Blackbox* Fungsi Mencari Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Mencari Instruktur Olahraga Alternatif 1
<b>Prosedur</b>	1. Membuka aplikasi apiko



	2. Memasukkan jenis olahraga pada field pencarian yang tidak terdaftar jenis olahraganya
<b>Hasil yang Diharapkan</b>	Menampilkan “Tidak Ditemukan”
<b>Hasil</b>	Menampilkan “Tidak Ditemukan”
<b>Status</b>	Valid
<b>Screenshot</b>	

### 6.2.5 Pengujian *Blackbox* Fungsi Mencari Kelas Event

Pengujian *blackbox testing* fungsi mencari kelas event ini didasarkan pada use case scenario mencari kelas event.

**Tabel 6.33 Pengujian *Blackbox* Fungsi Mencari Kelas Event**

<b>Nama Kasus Uji</b>	Mencari Kelas Event
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “kelas”</li> <li>3. Memilih menu “event”</li> </ol>
<b>Hasil yang Diharapkan</b>	Aplikasi menampilkan daftar instruktur olahraga yang membuat kelas event
<b>Hasil</b>	Aplikasi menampilkan daftar instruktur olahraga yang membuat kelas event
<b>Status</b>	Valid

### 6.2.6 Pengujian *Blackbox* Fungsi Melihat Detail Profil Instruktur Olahraga

Pengujian *blackbox testing* fungsi melihat detail profil instruktur olahraga ini didasarkan pada *use case scenario* melihat detail profil instruktur olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.34 Pengujian *Blackbox* Fungsi Melihat Detail Profil Instruktur Olahraga.

**Tabel 6.35 Pengujian *Blackbox* Fungsi Melihat Detail Profil Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Melihat Detail Profil Instruktur Olahraga
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih daftar instruktur yang ingin dilihat profilnya</li> <li>3. Menekan tombol “lihat detail”</li> </ol>
<b>Hasil yang Diharapkan</b>	Aplikasi menampilkan seluruh informasi profil instruktur olahraga yang berisi foto,nama,jenis kelamin,alamat,no telepon,alamat email,lama melatih,pengalaman

	melatih,keahlian lain,sertifikasi,dan daftar kelas yang ditawarkan
<b>Hasil</b>	Aplikasi menampilkan seluruh informasi profil instruktur olahraga yang berisi foto,nama,jenis kelamin,alamat,no telepon,alamat email,lama melatih,pengalaman melatih,keahlian lain,sertifikasi dan daftar semua kelas yang ditawarkan
<b>Status</b>	Valid

### 6.2.7 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Event

Pengujian *blackbox testing* fungsi melakukan pemesanan kelas event ini didasarkan pada *use case scenario* melakukan pemesanan kelas event.Hasil pengujiannya dapat dilihat pada Tabel 6.36 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Event dan hasil pengujian alternatifnya pada Tabel 6.37 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Event.

**Tabel 6.38 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Event**

<b>Nama Kasus Uji</b>	Pesan Kelas Event
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih instruktur yang ingin dipesan</li> <li>3. Menekan tombol “pesan instruktur”</li> <li>4. Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga</li> <li>5. Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,lokasi latihan</li> <li>6. Menekan tombol “pesan”</li> </ol>
<b>Hasil yang Diharapkan</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menyimpan data pemesanan</li> <li>2. Aplikasi menampilkan kode pembayaran</li> </ol>
<b>Hasil</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menyimpan data pemesanan</li> <li>2. Aplikasi menampilkan kode pembayaran</li> </ol>
<b>Status</b>	Valid

#### a. Kasus Uji Alternatif 1

**Tabel 6.39 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Event**

<b>Nama Kasus Uji</b>	Pesan Kelas Event Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih instruktur yang ingin dipesan dengan kouta yang sudah habis</li> <li>3. Menekan tombol “pesan instruktur”</li> </ol>



<b>Hasil yang Diharapkan</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga,jadwal latihan,jam</li> <li>2. Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,lokasi latihan</li> <li>3. Aplikasi menampilkan tulisan “Kouta Penuh”</li> </ol>
<b>Hasil</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga</li> <li>2. Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,lokasi latihan</li> <li>3. Aplikasi menampilkan tulisan “Kouta Penuh”</li> </ol>
<b>Status</b>	Valid
<b>Screenshot</b>	

### 6.2.8 Pengujian *Blackbox* Fungsi Melihat Kode Pembayaran

Pengujian *blackbox testing* fungsi melihat kode pembayaran ini didasarkan pada *use case scenario* melihat kode pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.40 Pengujian *Blackbox* Fungsi Melihat Kode Pembayaran.

**Tabel 6.41 Pengujian *Blackbox* Fungsi Melihat Kode Pembayaran**

<b>Nama Kasus Uji</b>	Cek Kode Bayar
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “pesan”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan form yang berisi informasi nama bank,kode bayar,jumlah pembayaran
<b>Hasil</b>	1. Aplikasi menampilkan form yang berisi informasi nama bank,kode bayar,jumlah pembayaran
<b>Status</b>	Valid

### 6.2.9 Pengujian *Blackbox* Fungsi Melihat Profil Member

Pengujian *blackbox testing* fungsi melihat profil *member* ini didasarkan pada *use case scenario* melihat profil *member*. Hasil pengujiannya dapat dilihat pada Tabel 6.42 Pengujian *Blackbox* Fungsi Melihat Profil *Member*.

**Tabel 6.43 Pengujian *Blackbox* Fungsi Melihat Profil Member**

<b>Nama Kasus Uji</b>	Lihat Profil Member
-----------------------	---------------------



<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “akun saya”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan profil member yang berisi nama lengkap,no telepon,alamat,jenis kelamin,alamat email,password
<b>Hasil</b>	1. Aplikasi menampilkan profil member yang berisi nama lengkap,no telepon,alamat,jenis kelamin,alamat email,password
<b>Status</b>	Valid

### 6.2.10 Pengujian *Blackbox* Fungsi Edit Profil *Member*

Pengujian blackbox testing fungsi edit profil *member* ini didasarkan pada *use case scenario* edit profil *member*. Hasil pengujian dapat dilihat pada Tabel 6.44 Pengujian *Blackbox* Fungsi Edit Profil *Member* dan hasil pengujian alternatifnya pada Tabel 6.45 Pengujian *Blackbox* Fungsi Edit Profil *Member*.

**Tabel 6.46 Pengujian *Blackbox* Fungsi Edit Profil *Member***

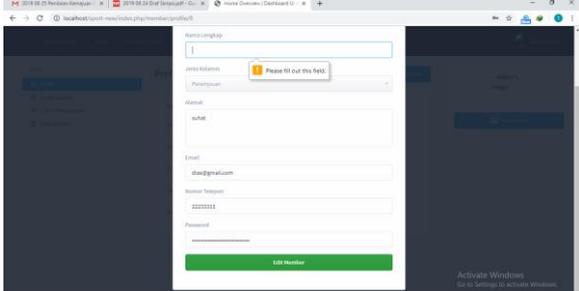
<b>Nama Kasus Uji</b>	Edit Profil Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “edit profil”</li> <li>3. Memasukkan data baru pada field yang ingin dilakukan perubahan</li> <li>4. Menekan tombol “Edit Member”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Status</b>	Valid

#### a. Kasus Uji Alternatif 1

**Tabel 6.47 Pengujian *Blackbox* Fungsi Edit Profil *Member***

<b>Nama Kasus Uji</b>	Edit Profil Member Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “edit profil”</li> <li>3. Tidak mengisi seluruh field pada form edit profil member yang ingin dilakukan perubahan</li> <li>4. Menekan tombol “Edit Member”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”



<b>Status</b>	Valid
<b>Screenshot</b>	

### 6.2.11 Pengujian *Blackbox* Fungsi Memberi Rating Instruktur

Pengujian *blackbox testing* fungsi memberi rating instruktur ini didasarkan pada *use case scenario* memberi rating instruktur. Hasil pengujiannya dapat dilihat pada Tabel 6.48 Pengujian *Blackbox* Fungsi Memberi Rating Instruktur

**Tabel 6.49 Pengujian *Blackbox* Fungsi Memberi Rating Instruktur**

<b>Nama Kasus Uji</b>	Memberi Rating Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih menu Lihat Jadwal</li> <li>3. Memasukkan bintang</li> <li>4. Menekan tombol “ok” pada alert</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil menyimpan data rating instruktur
<b>Hasil</b>	1. Aplikasi berhasil menyimpan data rating instruktur
<b>Status</b>	Valid

### 6.2.12 Pengujian *Blackbox* Fungsi Konfirmasi Pembayaran

Pengujian *blackbox testing* fungsi konfirmasi pembayaran ini didasarkan pada *use case scenario* konfirmasi pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.50 Pengujian *Blackbox* Fungsi Konfirmasi Pembayaran dan hasil pengujian alternatifnya pada Tabel 6.51 Pengujian *Blackbox* Fungsi Konfirmasi Pembayaran.

**Tabel 6.52 Pengujian *Blackbox* Fungsi Konfirmasi Pembayaran**

<b>Nama Kasus Uji</b>	Konfirmasi Pembayaran
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “lihat pembayaran”</li> <li>3. Menekan field lihat pada table konfirmasi pembayaran</li> <li>4. Memasukkan file bukti pembayaran</li> <li>5. Menekan tombol “konfirmasi pembayaran”</li> </ol>



<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Anda berhasil mengonfirmasi pembayaran. Silahkan tunggu pembayaran anda dikonfirmasi oleh admin”
<b>Hasil</b>	1. Aplikasi Aplikasi menampilkan pesan “Anda berhasil mengonfirmasi pembayaran. Silahkan tunggu pembayaran anda dikonfirmasi oleh admin”
<b>Status</b>	Valid

**a. Kasus Uji Alternatif 1**

**Tabel 6.53 Pengujian *Blackbox* Fungsi Konfirmasi Pembayaran**

<b>Nama Kasus Uji</b>	Konfirmasi Pembayaran Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “lihat pembayaran”</li> <li>3. Menekan field lihat pada table konfirmasi pembayaran</li> <li>4. Tidak Memasukkan file bukti pembayaran</li> <li>5. Menekan tombol “konfirmasi pembayaran”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.13 Pengujian *Blackbox* Fungsi Melakukan Pembatalan Pemesanan**

Pengujian *blackbox testing* fungsi melakukan pembatalan pemesanan ini didasarkan pada *use case scenario* melakukan pembatalan pemesanan. Hasil Pengujiannya dapat dilihat pada Tabel 6.54 Pengujian *Blackbox* Melakukan Pembatalan Pemesanan.

**Tabel 6.55 Pengujian *Blackbox* Melakukan Pembatalan Pemesanan**

<b>Nama Kasus Uji</b>	Batal Pesan
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “batalkan pemesanan”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Berhasil membatalkan pemesanan”



	2. Aplikasi menampilkan status pembayaran menjadi dibatalkan
<b>Hasil</b>	1. Aplikasi menampilkan pesan “Berhasil membatalkan pemesanan” 2. Aplikasi menampilkan status pembayaran menjadi dibatalkan
<b>Status</b>	Valid

#### 6.2.14 Pengujian *Blackbox* Fungsi Hapus Akun *Member*

Pengujian *blackbox testing* fungsi hapus akun member ini didasarkan pada *use case scenario* melakukan hapus akun member. Hasil pengujiannya dapat dilihat pada Tabel 6.56 Pengujian *Blackbox* Fungsi Hapus Akun *Member*.

**Tabel 6.57 Pengujian *Blackbox* Fungsi Hapus Akun *Member***

<b>Nama Kasus Uji</b>	Hapus Akun Member
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Masuk pada halaman profil member 3. Menekan menu “hapus akun”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Akun anda berhasil dihapus”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “Akun anda berhasil dihapus”
<b>Status</b>	Valid

#### 6.2.15 Pengujian *Blackbox* Fungsi Melihat Jadwal Latihan *Member*

Pengujian *blackbox testing* fungsi melihat jadwal latihan member ini didasarkan pada *use case scenario* melihat jadwal latihan *member*. Hasil Pengujiannya dapat dilihat pada Tabel 6.58 Pengujian *Blackbox* Fungsi Melihat Jadwal Latihan *Member*.

**Tabel 6.59 Pengujian *Blackbox* Fungsi Melihat Jadwal Latihan *Member***

<b>Nama Kasus Uji</b>	Lihat Jadwal Member
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Masuk pada halaman profil member 3. Menekan menu “Lihat Jadwal”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan jadwal latihan member yang berisi sesi, jenis, nama pelatih, jam, tempat, rating, agenda, durasi, jumlah murid, kalender pencarian jadwal
<b>Hasil</b>	1. Aplikasi menampilkan jadwal latihan member yang berisi sesi, kelas, pelatih, waktu, lokasi
<b>Status</b>	Valid



### 6.2.16 Pengujian *Blackbox* Fungsi Melihat Profil Instruktur Olahraga

Pengujian *blackbox testing* fungsi melihat profil instruktur olahraga ini didasarkan pada *use case scenario* melihat profil instruktur olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.60 Pengujian *Blackbox* Fungsi Melihat Profil Instruktur Olahraga.

**Tabel 6.61 Pengujian *Blackbox* Fungsi Melihat Profil Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Lihat Profil Instruktur Olahraga
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “akun saya”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan profil instruktur olahraga yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no ktp, nama bank, rekening bank, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password
<b>Hasil</b>	1. Aplikasi menampilkan profil instruktur olahraga yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no ktp, nama bank, rekening bank, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password
<b>Status</b>	Valid

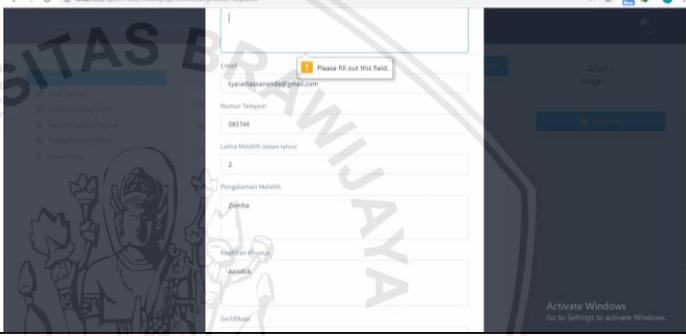
### 6.2.17 Pengujian *Blackbox* Fungsi Edit Profil Instruktur Olahraga

Pengujian *blackbox testing* fungsi edit profil instruktur olahraga ini didasarkan pada *use case scenario* edit profil instruktur olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.62 Pengujian *Blackbox* Fungsi Edit Profil Instruktur Olahraga dan hasil pengujian alternatifnya pada Tabel 6.63 Pengujian *Blackbox* Fungsi Edit Profil Instruktur Olahraga.

**Tabel 6.64 Pengujian *Blackbox* Fungsi Edit Profil Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Edit Profil Instruktur Olahraga
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “edit profil”</li> <li>3. Memasukkan data baru pada field yang ingin dilakukan perubahan</li> <li>4. Menekan tombol “Edit Instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Status</b>	Valid

a.Kasus Uji Alternatif1  
Tabel 6.65 Pengujian *Blackbox* Fungsi Edit Profil Instruktur Olahraga

<b>Nama Kasus Uji</b>	Edit Profil Instruktur Olahraga Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan tombol “edit profil”</li> <li>3. Tidak mengisi seluruh field pada form edit profil instruktur yang ingin dilakukan perubahan</li> <li>4. Menekan tombol “Edit Instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

### 6.2.18 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Reguler

Pengujian *blackbox testing* fungsi melakukan pembuatan kelas reguler ini didasarkan pada *use case scenario* melakukan pembuatan kelas reguler. Hasil pengujiannya dapat dilihat pada Tabel 6.66 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Reguler dan hasil pengujian alternatifnya pada Tabel 6.67 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Reguler.

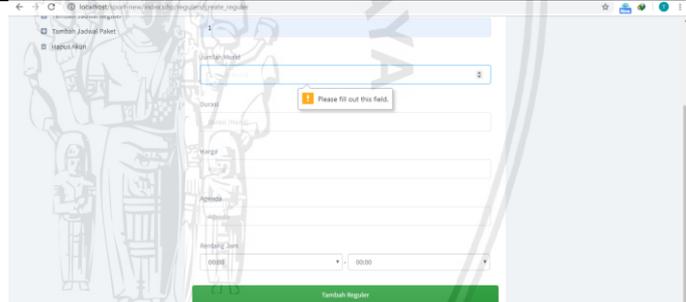
Tabel 6.68 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Reguler

<b>Nama Kasus Uji</b>	Buat Reguler
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “Tambah Jadwal Reguler”</li> <li>3. Memasukan data pada form berisi sesi, jumlah murid, durasi, harga, agenda</li> <li>4. Memilih rentang jam kelas</li> <li>5. Menekan tombol “Tambah Reguler”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil menyimpan data pembuatan kelas reguler

<b>Hasil</b>	1. Aplikasi berhasil menyimpan data pembuatan kelas reguler
<b>Status</b>	Valid

**a. Kasus Uji Alternatif1**

**Tabel 6.69 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Reguler**

<b>Nama Kasus Uji</b>	Buat Reguler Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “Tambah Jadwal Reguler”</li> <li>3. Tidak mengisi seluruh filed pada form yang berisi sesi,jumlah murid,durasi,harga,agenda</li> <li>4. Memilih rentang jam kelas</li> <li>5. Menekan tombol “Tambah Reguler”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.19 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Event**

Pengujian *blackbox testing* fungsi melakukan pembuatan kelas event ini didasarkan pada *use case scenario* melakukan pembuatan kelas event. Hasil pengujiannya dapat dilihat pada Tabel 6.70 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Event dan hasil pengujian alternatifnya pada Tabel 6.71 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Event.

**Tabel 6.72 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Event**

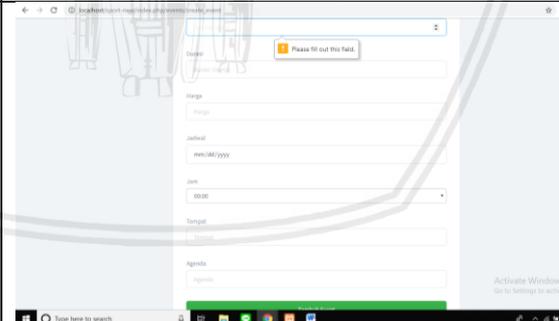
<b>Nama Kasus Uji</b>	Buat Event
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “Tambah Jadwal Event”</li> <li>3. Memasukan data pada form berisi sesi,jumlah murid,durasi,harga,agenda,jadwal,jam,tempat</li> </ol>



	4. Menekan tombol “Tambah Event”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Sukses mendaftar event. Sukses mendaftar event.Silahkan tunggu konfirmasi admin”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “Sukses mendaftar event. Sukses mendaftar event.Silahkan tunggu konfirmasi admin”
<b>Status</b>	Valid

**a.Kasus Uji Alternatif 1**

**Tabel 6.73 Pengujian Blackbox Fungsi Melakukan Pembuatan Kelas Event**

<b>Nama Kasus Uji</b>	Buat Event Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>6. Membuka aplikasi apiko</li> <li>7. Menekan menu “Tambah Jadwal Event”</li> <li>8. Tidak mengisi seluruh field pada form yang berisi sesi, jumlah murid, durasi, harga, agenda</li> <li>9. Memilih rentang jam kelas</li> <li>10. Menekan tombol “Tambah Event”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.20 Pengujian Blackbox Fungsi Melihat Jadwal Latihan Instruktur Olahraga**

Pengujian *blackbox testing* fungsi melihat jadwal latihan instruktur olahraga ini didasarkan pada *use case scenario* melihat jadwal latihan instruktur olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.74 Pengujian *Blackbox* Fungsi Melihat Jadwal Latihan Instruktur Olahraga.

**Tabel 6.75 Pengujian *Blackbox* Fungsi Melihat Jadwal Latihan Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Lihat Jadwal Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk pada halaman profil instruktur olahraga</li> <li>3. Menekan menu "Lihat Jadwal"</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan jadwal latihan member yang berisi sesi, kelas, member, waktu, lokasi
<b>Hasil</b>	1. Aplikasi menampilkan jadwal latihan member yang berisi sesi, jenis, jam, tempat, nama member, rating, agenda, jumlah murid, kalender pencarian jadwal
<b>Status</b>	Valid

### 6.2.21 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Paket

Pengujian *blackbox testing* fungsi melakukan pembuatan kelas paket ini didasarkan pada *use case scenario* melakukan pembuatan kelas paket. Hasil pengujiannya dapat dilihat pada Tabel 6.76 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Paket dan hasil pengujian alternatifnya pada Tabel 6.77 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Paket.

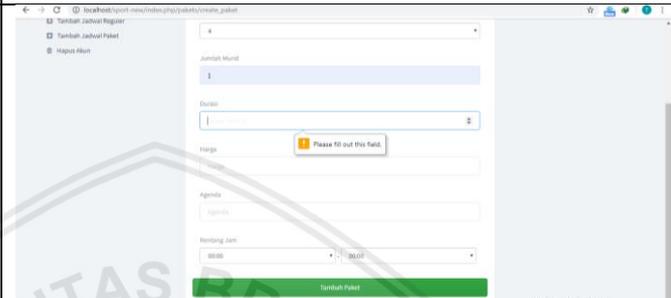
**Tabel 6.78 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Paket**

<b>Nama Kasus Uji</b>	Buat Paket
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu "Tambah Jadwal Paket"</li> <li>3. Memasukan data pada form berisi sesi, jumlah murid, durasi, harga, agenda</li> <li>4. Memilih rentang jam kelas</li> <li>5. Menekan tombol "Tambah Paket"</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil menyimpan data pembuatan kelas paket
<b>Hasil</b>	1. Aplikasi berhasil menyimpan data pembuatan kelas paket
<b>Status</b>	Valid

#### a. Kasus Uji Alternatif 1

**Tabel 6.79 Pengujian *Blackbox* Fungsi Melakukan Pembuatan Kelas Paket**

<b>Nama Kasus Uji</b>	Buat Paket Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>11. Membuka aplikasi apiko</li> <li>12. Menekan menu "Tambah Jadwal Paket"</li> <li>13. Tidak mengisi seluruh field pada form yang berisi</li> </ol>

	<p>sesi, jumlah murid, durasi, harga, agenda</p> <p>14. Memilih rentang jam kelas</p> <p>15. Menekan tombol “Tambah Paket”</p>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	 <p>The screenshot shows a web browser window with a form titled 'Tambah Paket'. The form includes fields for 'Jumlah Murid', 'Durasi', 'Harga', 'Agenda', and 'Rentang Jam'. A red error message 'Please fill out this field' is displayed above the 'Durasi' field. A green 'Tambah Paket' button is at the bottom of the form.</p>

### 6.2.22 Pengujian *Blackbox* Fungsi Hapus Akun Instruktur Olahraga

Pengujian *blackbox testing* fungsi hapus akun instruktur olahraga ini didasarkan pada *use case scenario* melakukan hapus akun instruktur olahraga. Hasil pengujiannya dapat dilihat pada Tabel 6.80 Pengujian *Blackbox* Fungsi Hapus Akun Instruktur Olahraga.

**Tabel 6.81 Pengujian *Blackbox* Fungsi Hapus Akun Instruktur Olahraga**

<b>Nama Kasus Uji</b>	Hapus Akun Instruktur Olahraga
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk pada halaman profil instruktur olahraga</li> <li>3. Menekan menu “hapus akun”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Akun anda berhasil dihapus”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “Akun anda berhasil dihapus”
<b>Status</b>	Valid

### 6.2.23 Pengujian *Blackbox* Fungsi Melihat Daftar Instruktur Terverifikasi

Pengujian *blackbox testing* fungsi melihat daftar instruktur terverifikasi ini didasarkan pada *use case scenario* melihat daftar instruktur terverifikasi. Hasil pengujiannya dapat dilihat pada Tabel 6.82 Pengujian *Blackbox* Fungsi Melihat Daftar Instruktur Terverifikasi.

**Tabel 6.83 Pengujian *Blackbox* Fungsi Melihat Daftar Instruktur Terverifikasi**

<b>Nama Kasus Uji</b>	Lihat Daftar Instruktur Terverifikasi
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Sudah login sebagai admin</li> </ol>



	<ol style="list-style-type: none"> <li>3. Menekan menu “Instruktur”</li> <li>4. Memilih menu “Instruktur Terverifikasi”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan nama,jenis kelamin,alamat email, no telepon
<b>Hasil</b>	1. Aplikasi menampilkan nama,jenis kelamin,alamat email, no telepon
<b>Status</b>	Valid

#### 6.2.24 Pengujian *Blackbox* Fungsi Melihat Detail Profil Pendaftaran Instruktur

Pengujian *blackbox testing* fungsi melihat detail profil pendaftaran instruktur ini didasarkan pada *use case scenario* melihat detail profil pendaftaran instruktur.Hasil pengujiannya dapat dilihat pada Tabel 6.84 Pengujian *Blackbox* Fungsi Melihat Detail Profil Pendaftaran Instruktur.

**Tabel 6.85 Pengujian *Blackbox* Fungsi Melihat Detail Profil Pendaftaran Instruktur**

<b>Nama Kasus Uji</b>	Lihat Profil Instruktur Pendaftar
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Sudah login sebagai admin</li> <li>3. Menekan tombol“detail instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1.Aplikasi menampilkan nama lengkap,alamat email,jenis kelamin ,alamat,no telepon,lama melatih,pengalaman melatih,keahlian lain,sertifikasi
<b>Hasil</b>	1. Aplikasi menampilkan nama lengkap,alamat email,jenis kelamin ,alamat,no telepon,lama melatih,pengalaman melatih,keahlian lain,sertifikasi
<b>Status</b>	Valid

#### 6.2.25 Pengujian *Blackbox* Fungsi Melakukan Validasi Pendaftaran Instruktur

Pengujian *blackbox testing* fungsi melakukan validasi pendaftaran instruktur ini didasarkan pada *use case scenario* melakukan validasi pendaftaran instruktur.Hasil pengujiannya dapat dilihat pada Tabel 6.86 Pengujian *Blackbox* Fungsi Melakukan Validasi Pendaftaran Instruktur.

**Tabel 6.87 Pengujian *Blackbox* Fungsi Melakukan Validasi Pendaftaran Instruktur**

<b>Nama Kasus Uji</b>	Validasi Pendaftaran Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Sudah login sebagai admin</li> <li>3. Menekan tombol “konfirmasi”</li> <li>4. Memilih tombol “terima pendaftaran”</li> </ol>
<b>Hasil yang Diharapkan</b>	1.Aplikasi berhasil menyimpan data validasi pendaftaran instruktur olahraga
<b>Hasil</b>	1. Aplikasi berhasil menyimpan data validasi pendaftaran instruktur olahraga

Status	Valid
--------	-------

### 6.2.26 Pengujian *Blackbox* Fungsi Menolak Pendaftaran Instruktur

Pengujian *blackbox testing* fungsi menolak pendaftaran instruktur ini didasarkan pada *use case scenario* menolak pendaftaran instruktur. Hasil pengujiannya dapat dilihat pada Tabel 6.88 Pengujian *Blackbox* Fungsi Menolak Pendaftaran Instruktur.

**Tabel 6.89 Pengujian *Blackbox* Fungsi Menolak Pendaftaran Instruktur**

<b>Nama Kasus Uji</b>	Tolak Pendaftaran Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Sudah login sebagai admin</li> <li>3. Menekan tombol “konfirmasi”</li> <li>4. Memilih tombol “tolak pendaftaran”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil menghapus data pendaftaran instruktur olahraga
<b>Hasil</b>	1. Aplikasi berhasil menghapus data pendaftaran instruktur olahraga
<b>Status</b>	Valid

### 6.2.27 Pengujian *Blackbox* Fungsi Melihat Daftar Pembayaran Member

Pengujian *blackbox testing* fungsi melihat daftar pembayaran member ini didasarkan pada *use case scenario* melihat daftar pembayaran member. Hasil pengujiannya dapat dilihat pada Tabel 6.90 Pengujian *Blackbox* Fungsi Melihat Daftar Pembayaran Member.

**Tabel 6.91 Pengujian *Blackbox* Fungsi Melihat Daftar Pembayaran Member**

<b>Nama Kasus Uji</b>	Lihat Daftar Pembayaran Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Sudah login sebagai admin</li> <li>3. Menekan menu “pembayaran”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan daftar tabel detail pemesanan, detail pemesanan, detail pembayaran, status pembayaran dan option yang berisi tombol konfirmasi pembayaran dan tolak pembayaran
<b>Hasil</b>	1. Aplikasi menampilkan daftar tabel detail pemesanan, detail pemesanan, detail pembayaran, status pembayaran dan option yang berisi tombol konfirmasi pembayaran dan tolak pembayaran
<b>Status</b>	Valid

### 6.2.28 Pengujian Blackbox Fungsi Melakukan Validasi Pembayaran

Pengujian *blackbox testing* fungsi melakukan validasi pembayaran ini didasarkan pada *use case scenario* melakukan validasi pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.92 Pengujian *Blackbox* Fungsi Melakukan Validasi Pembayaran.

**Tabel 6.93 Pengujian *Blackbox* Fungsi Melakukan Validasi Pembayaran**

Nama Kasus Uji	Validasi Pembayaran
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk pada halaman admin</li> <li>3. Menekan tombol "option"</li> <li>4. Memilih tombol "konfirmasi pembayaran"</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan "berhasil mengkonfirmasi pembayaran dan mengirimkan jadwal ke email."
<b>Hasil</b>	1. Aplikasi menampilkan pesan "berhasil mengkonfirmasi pembayaran dan mengirimkan jadwal ke email."
<b>Status</b>	Valid

### 6.2.29 Pengujian Blackbox Fungsi Membatalkan Pembayaran

Pengujian *blackbox testing* fungsi membatalkan pembayaran ini didasarkan pada *use case scenario* membatalkan pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.94 Pengujian *Blackbox* Fungsi Membatalkan Pembayaran.

**Tabel 6.95 Pengujian *Blackbox* Fungsi Membatalkan Pembayaran**

Nama Kasus Uji	Validasi Pembayaran
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk pada halaman admin</li> <li>3. Menekan tombol "option"</li> <li>4. Memilih tombol "tolak pembayaran"</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi Aplikasi menampilkan pesan "berhasil menolak pembayaran"
<b>Hasil</b>	1. Aplikasi Aplikasi menampilkan pesan "berhasil menolak pembayaran"
<b>Status</b>	Valid

### 6.2.30 Pengujian Blackbox Fungsi Melihat Daftar Member

Pengujian *blackbox testing* fungsi melihat daftar member ini didasarkan pada *use case scenario* melihat daftar member. Hasil pengujiannya dapat dilihat pada Tabel 6.96 Pengujian *Blackbox* Fungsi Melihat Daftar Member.

**Tabel 6.97 Pengujian *Blackbox* Fungsi Melihat Daftar Member**

Nama Kasus Uji	Lihat Daftar Member

<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Masuk pada halaman admin 3. Menekan menu “member”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan berisi nama lengkap,no telepon,jenis kelamin,alamat email
<b>Hasil</b>	1. Aplikasi menampilkan berisi nama lengkap,no telepon,jenis kelamin,alamat email
<b>Status</b>	Valid

### 6.2.31 Pengujian *Blackbox* Fungsi Melihat Daftar Kelas Event

Pengujian *blackbox testing* fungsi melihat daftar kelas event ini didasarkan pada *use case scenario* melihat daftar kelas event. Hasil pengujiannya dapat dilihat pada Tabel 6.98 Pengujian *Blackbox* Fungsi Melihat Daftar Kelas Event.

**Tabel 6.99 Pengujian *Blackbox* Fungsi Melihat Daftar Kelas Event**

<b>Nama Kasus Uji</b>	Lihat Daftar Event
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Masuk pada halaman admin 3. Menekan menu “event”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan berisi agenda,status,sesi,durasi,jumlah murid,harga,jadwal,lokasi
<b>Hasil</b>	1. Aplikasi menampilkan berisi agenda,status,sesi,durasi,jumlah murid,harga,jadwal,lokasi
<b>Status</b>	Valid

### 6.2.32 Pengujian *Blackbox* Fungsi Melakukan Validasi Kelas Event

Pengujian *blackbox testing* fungsi melakukan validasi kelas event ini didasarkan pada *use case scenario* melakukan validasi kelas event. Hasil pengujiannya dapat dilihat pada Tabel 6.100 Pengujian *Blackbox* Fungsi Melakukan Validasi Kelas Event.

**Tabel 6.101 Pengujian *Blackbox* Fungsi Melakukan Validasi Kelas Event**

<b>Nama Kasus Uji</b>	Validasi Event
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Masuk pada halaman admin 3. Menekan tombol “option” 4. Memilih tombol “terima event”
<b>Hasil yang Diharapkan</b>	1. Aplikasi Aplikasi menampilkan pesan “berhasil menerima event”
<b>Hasil</b>	1. Aplikasi Aplikasi menampilkan pesan “berhasil menerima event”
<b>Status</b>	Valid

### 6.2.33 Pengujian *Blackbox* Fungsi Menolak Kelas Event

Pengujian *blackbox testing* fungsi menolak kelas event ini didasarkan pada *use case scenario* menolak kelas event. Hasil pengujiannya dapat dilihat pada Tabel 6.102 Pengujian *Blackbox* Fungsi Menolak Kelas Event.

**Tabel 6.103 Pengujian *Blackbox* Fungsi Menolak Kelas Event**

Nama Kasus Uji	Tolak Event
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk pada halaman admin</li> <li>3. Menekan tombol “option”</li> <li>4. Memilih tombol “tolak event”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi Aplikasi Aplikasi menampilkan pesan “berhasil menghapus event”
<b>Hasil</b>	1. Aplikasi Aplikasi Aplikasi menampilkan pesan “berhasil menghapus event”
<b>Status</b>	Valid

### 6.2.34 Pengujian *Blackbox* Fungsi Logout

Pengujian *blackbox testing* fungsi logout ini didasarkan pada *use case scenario logout*. Hasil pengujiannya dapat dilihat pada Tabel 6.104 Pengujian *Blackbox* Fungsi Logout.

**Tabel 6.105 Pengujian *Blackbox* Fungsi Logout**

Nama Kasus Uji	Logout
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Melakukan login</li> <li>3. Menekan tombol “logout”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan halaman utama
<b>Hasil</b>	1. Aplikasi menampilkan halaman utama
<b>Status</b>	Valid

### 6.2.35 Pengujian *Blackbox* Fungsi Melihat Daftar Instruktur Belum Terverifikasi

Pengujian *blackbox testing* fungsi melihat daftar instruktur belum terverifikasi ini didasarkan pada *use case scenario* melihat daftar instruktur belum terverifikasi. Hasil pengujiannya dapat dilihat pada Tabel 6.106 Pengujian *Blackbox* Fungsi Melihat Daftar Instruktur Belum Terverifikasi.

**Tabel 6.107 Pengujian *Blackbox* Fungsi Melihat Daftar Instruktur Belum Terverifikasi**

Nama Kasus Uji	Lihat Daftar Instruktur Terverifikasi
----------------	---------------------------------------

<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Menekan menu “Instruktur”</li> <li>3. Memilih menu “Instruktur Belum Terverifikasi”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan nama, jenis kelamin, alamat email, no telepon, option yang berisi tombol terima dan tolak instruktur
<b>Hasil</b>	1. Aplikasi menampilkan nama, jenis kelamin, alamat email, no telepon, option yang berisi tombol terima dan tolak instruktur
<b>Status</b>	Valid

### 6.2.36 Pengujian *Blackbox* Fungsi Menambahkan Instruktur

Pengujian *blackbox testing* fungsi menambahkan instruktur ini didasarkan pada *use case scenario* menambahkan instruktur. Hasil pengujiannya dapat dilihat pada Tabel 6.108 Pengujian *Blackbox* Fungsi Menambahkan Instruktur dan hasil pengujian alternatifnya pada Tabel 6.109 Pengujian *Blackbox* Fungsi Menambahkan Instruktur dan Tabel 6.110 Pengujian *Blackbox* Fungsi Menambahkan Instruktur.

**Tabel 6.111 Pengujian *Blackbox* Fungsi Menambahkan Instruktur**

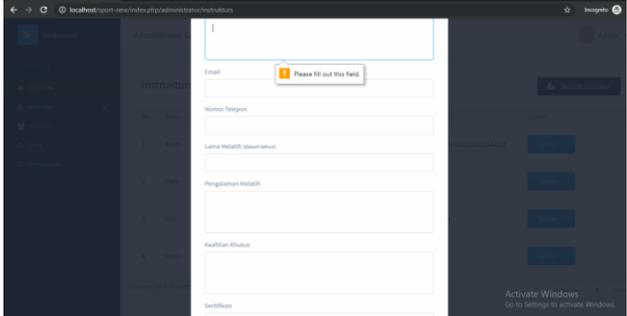
<b>Nama Kasus Uji</b>	Tambah Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Memilih tombol “tambah instruktur”</li> <li>4. Mengisi form yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data penambahan instruktur baru
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data penambahan instruktur baru
<b>Status</b>	Valid

#### a. Kasus Uji Alternatif 1

**Tabel 6.112 Pengujian *Blackbox* Fungsi Menambahkan Instruktur**

<b>Nama Kasus Uji</b>	Tambah Instruktur Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Memilih tombol “tambah instruktur”</li> <li>4. Tidak mengisi seluruh form yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no telepon, lama melatih, pengalaman melatih, keahlian lain, sertifikasi, password</li> </ol>



<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**b.Kasus Uji Alternatif2**

**Tabel 6.113 Pengujian *Blackbox* Fungsi Menambahkan Instruktur**

<b>Nama Kasus Uji</b>	Tambah Instruktur Alternatif 2
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Memilih tombol “tambah instruktur”</li> </ol> <p>Tidak mengisi seluruh form yang berisi nama lengkap,alamat email,jenis kelamin,alamat,no telepon,lama melatih,pengalaman melatih,keahlian lain,sertifikasi,password</p>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.37 Pengujian *Blackbox* Fungsi Menambahkan *Member***

Pengujian *blackbox testing* fungsi menambahkan *member* ini didasarkan pada *use case scenario* menambahkan *member*.Hasil pengujiannya dapat dilihat pada Tabel 6.114 Pengujian *Blackbox* Fungsi Menambahkan *Member* dan hasil pengujian alternatifnya pada Tabel 6.115 Pengujian *Blackbox* Fungsi Menambahkan *Member* dan Tabel 6.116 Pengujian *Blackbox* Fungsi Menambahkan *Member*.

**Tabel 6.117 Pengujian *Blackbox* Fungsi Menambahkan *Member***

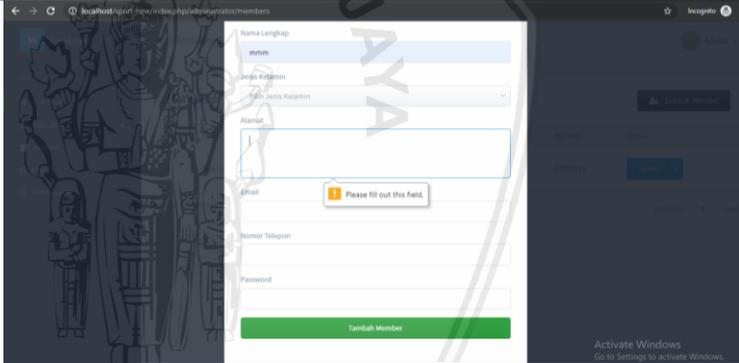
<b>Nama Kasus Uji</b>	Tambah Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Memilih tombol “tambah member”</li> <li>4. Mengisi form yang berisi nama lengkap,alamat email,jenis kelamin,alamat,no telepon,password</li> </ol>



<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data penambahan member baru
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data penambahan member baru
<b>Status</b>	Valid

**a. Kasus Uji Alternatif 1**

**Tabel 6.118 Pengujian *Blackbox* Fungsi Menambahkan Member**

<b>Nama Kasus Uji</b>	Tambah Member Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Memilih tombol “tambah member”</li> <li>4. Tidak mengisi seluruh form yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no telepon, password</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**b. Kasus Uji Alternatif 2**

**Tabel 6.119 Pengujian *Blackbox* Fungsi Menambahkan Member**

<b>Nama Kasus Uji</b>	Tambah Member Alternatif 2
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Memilih tombol “tambah member”</li> <li>4. Tidak mengisi seluruh form yang berisi nama lengkap, alamat email, jenis kelamin, alamat, no telepon, password</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “email telah terdaftar”
<b>Status</b>	Valid
<b>Screenshot</b>	



### 6.2.38 Pengujian *Blackbox* Fungsi Hapus Member

Pengujian *blackbox testing* fungsi hapus member ini didasarkan pada *use case scenario* hapus member. Hasil pengujiannya dapat dilihat pada Tabel 6.120 Pengujian *Blackbox* Fungsi Hapus Member.

**Tabel 6.121 Pengujian *Blackbox* Fungsi Hapus Member**

<b>Nama Kasus Uji</b>	Hapus Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Menekan tombol "option"</li> <li>4. Memilih tombol "delete profil"</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data member yang dihapus
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data member yang dihapus
<b>Status</b>	Valid

### 6.2.39 Pengujian *Blackbox* Fungsi Hapus Instruktur

Pengujian *blackbox testing* fungsi hapus instruktur ini didasarkan pada *use case scenario* hapus instruktur. Hasil pengujiannya dapat dilihat pada Tabel 6.122 Pengujian *Blackbox* Fungsi Hapus Instruktur.

**Tabel 6.123 Pengujian *Blackbox* Fungsi Hapus Instruktur**

<b>Nama Kasus Uji</b>	Hapus Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Menekan tombol "option"</li> <li>4. Memilih tombol "delete profil"</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data instruktur yang dihapus
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data instruktur yang dihapus
<b>Status</b>	Valid

### 6.2.40 Pengujian *Blackbox* Fungsi Melihat Dashboard

Pengujian *blackbox testing* fungsi melihat dashboard ini didasarkan pada *use case scenario* melihat dashboard. Hasil pengujiannya dapat dilihat pada Tabel 6.124 Pengujian *Blackbox* Fungsi Melihat Dashboard.

**Tabel 6.125 Pengujian *Blackbox* Fungsi Melihat Dashboard**

<b>Nama Kasus Uji</b>	Lihat Dashboard
-----------------------	-----------------

<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Menekan menu “overview”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil menampilkan dashboard admin yang berisi jumlah member dan instruktur, total kelas regular, event, paket, daftar event yang perlu diverifikasi, pendaftar instruktur, transaksi pembayaran sukses, daftar pembayaran yang perlu dikonfirmasi dan daftar pembayaran yang belum dibayarkan
<b>Hasil</b>	1. Aplikasi berhasil menampilkan dashboard admin yang berisi jumlah member dan instruktur, total kelas regular, event, paket, daftar event yang perlu diverifikasi, pendaftar instruktur, transaksi pembayaran sukses, daftar pembayaran yang perlu dikonfirmasi dan daftar pembayaran yang belum dibayarkan
<b>Status</b>	Valid

#### 6.2.41 Pengujian *Blackbox* Fungsi Edit Instruktur

Pengujian *blackbox testing* fungsi edit instruktur ini didasarkan pada *use case scenario* edit instruktur. Hasil pengujiannya dapat dilihat pada Tabel 6.126 Pengujian *Blackbox* Fungsi Edit Instruktur dan hasil pengujian alternatifnya pada Tabel 6.127 Pengujian *Blackbox* Fungsi Edit Instruktur.

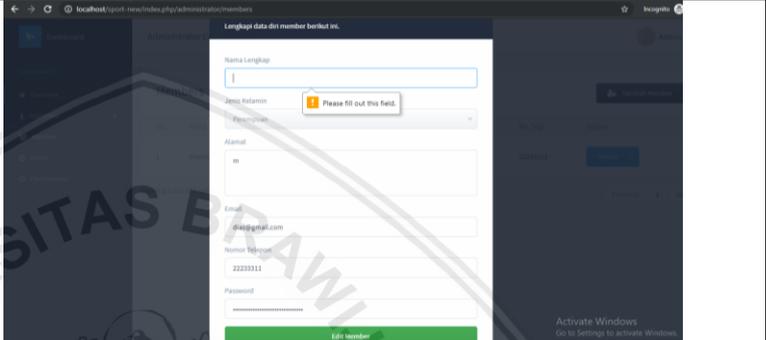
**Tabel 6.128 Pengujian *Blackbox* Fungsi Edit Instruktur**

<b>Nama Kasus Uji</b>	Edit Instruktur
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk ke halaman admin</li> <li>3. Menekan tombol “option”</li> <li>4. Memilih tombol “edit profil”</li> <li>5. Memasukkan data baru pada field yang ingin dilakukan perubahan</li> <li>6. Menekan tombol “Edit Instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Status</b>	Valid

##### a. Kasus Uji Alternatif 1

**Tabel 6.129 Pengujian *Blackbox* Fungsi Edit Instruktur**

<b>Nama Kasus Uji</b>	Edit Instruktur Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk ke halaman admin</li> </ol>

	<ol style="list-style-type: none"> <li>Menekan tombol “option”</li> <li>Memilih tombol “edit profil”</li> <li>Tidak mengisi seluruh field pada form edit instruktur yang ingin dilakukan perubahan</li> <li>Menekan tombol “Edit Instruktur”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshoot</b>	

### 6.2.42 Pengujian *Blackbox* Fungsi Edit *Member*

Pengujian *blackbox* testing fungsi edit member ini didasarkan pada *use case scenario* edit *member*. Hasil pengujiannya dapat dilihat pada Tabel 6.130 Pengujian *Blackbox* Fungsi Edit *Member* dan hasil pengujian alternatifnya pada Tabel 6.131 Pengujian *Blackbox* Fungsi Edit *Member*

**Tabel 6.132 Pengujian *Blackbox* Fungsi Edit *Member***

<b>Nama Kasus Uji</b>	Edit Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>Membuka aplikasi apiko</li> <li>Masuk ke halaman admin</li> <li>Menekan tombol “option”</li> <li>Memilih tombol “edit profil”</li> <li>Memasukkan data baru pada field yang ingin dilakukan perubahan</li> <li>Menekan tombol “Edit Member”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Hasil</b>	1. Aplikasi berhasil mengupdate data perubahan profil
<b>Status</b>	Valid



a.Kasus Uji Alternatif1

Tabel 6.133 Pengujian *Blackbox* Fungsi Edit *Member*

<b>Nama Kasus Uji</b>	Edit Member Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk ke halaman admin</li> <li>3. Menekan tombol “option”</li> <li>4. Memilih tombol “edit profil”</li> <li>5. Tidak mengisi seluruh field pada form yang ingin dilakukan perubahan</li> <li>6. Menekan tombol “Edit Member”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

6.2.43 Pengujian *Blackbox* Fungsi Edit Foto

Pengujian *blackbox testing* fungsi edit foto ini didasarkan pada *use case scenario* edit foto. Hasil pengujiannya dapat dilihat pada Tabel 6.134 Pengujian *Blackbox* Fungsi Edit Foto dan hasil pengujian alternatifnya pada Tabel 6.135 Pengujian *Blackbox* Fungsi Edit Foto.

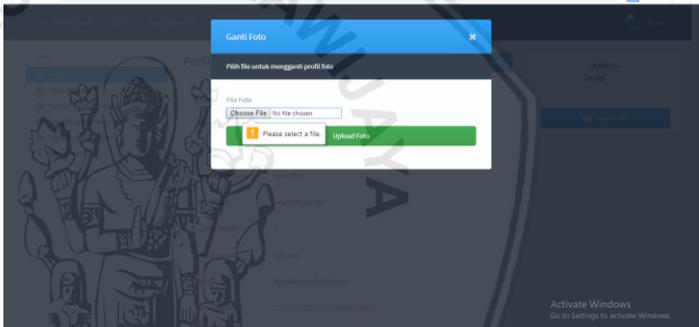
Tabel 6.136 Pengujian *Blackbox* Fungsi Edit Foto

<b>Nama Kasus Uji</b>	Edit Foto
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk ke halaman profil</li> <li>3. Menekan tombol “ganti foto”</li> <li>4. Memilih tombol “choose file”</li> <li>5. Memilih foto</li> <li>6. Menekan tombol “Upload foto”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan foto terbaru
<b>Hasil</b>	1. Aplikasi menampilkan foto terbaru

Status	Valid
--------	-------

**a.Kasus Uji Alternatif1**

**Tabel 6.137 Pengujian *Blackbox* Fungsi Edit Foto**

<b>Nama Kasus Uji</b>	Edit Foto Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk ke halaman profil</li> <li>3. Menekan tombol “ganti foto”</li> <li>4. Memilih tombol “choose file”</li> <li>5. Tidak memasukkan foto</li> <li>6. Menekan tombol “Upload foto”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Please out this file”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Please out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**6.2.44 Pengujian *Blackbox* Fungsi Mencari Kelas Reguler**

Pengujian *blackbox testing* fungsi mencari kelas reguler ini didasarkan pada *use case scenario* mencari kelas regular. Hasil pengujiannya dapat dilihat pada Tabel 6.138 Pengujian *Blackbox* Fungsi Mencari Kelas Reguler.

**Tabel 6.139 Pengujian *Blackbox* Fungsi Mencari Kelas Reguler**

<b>Nama Kasus Uji</b>	Cari Kelas Reguler
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih menu “kelas”</li> <li>3. Menekan menu “regular”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan nama instruktur, jenis olahraga ,foto instruktur, sesi, harga, durasi, jumlah murid
<b>Hasil</b>	1. Aplikasi menampilkan nama instruktur, jenis olahraga ,foto instruktur, sesi, harga, durasi, jumlah murid
<b>Status</b>	Valid



### 6.2.45 Pengujian *Blackbox* Fungsi Mencari Kelas Paket

Pengujian *blackbox testing* fungsi mencari kelas paket ini didasarkan pada *use case scenario* mencari kelas paket. Hasil pengujiannya dapat dilihat pada Tabel 6.140 Pengujian *Blackbox* Fungsi Mencari Kelas Paket.

**Tabel 6.141 Pengujian *Blackbox* Fungsi Mencari Kelas Paket**

<b>Nama Kasus Uji</b>	Cari Kelas Paket
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih menu “kelas”</li> <li>3. Menekan menu “paket”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan nama instruktur, jenis olahraga, foto instruktur, sesi, harga, durasi, jumlah murid
<b>Hasil</b>	1. Aplikasi menampilkan nama instruktur, jenis olahraga, foto instruktur, sesi, harga, durasi, jumlah murid
<b>Status</b>	Valid

### 6.2.46 Pengujian *Blackbox* Fungsi Mencari Berdasarkan Kriteria

Pengujian *blackbox testing* fungsi mencari berdasarkan harga ini didasarkan pada *use case scenario* mencari berdasarkan harga. Hasil pengujiannya dapat dilihat pada Tabel 6.142 Pengujian *Blackbox* Fungsi Mencari Berdasarkan Kriteria.

**Tabel 6.143 Pengujian *Blackbox* Fungsi Mencari Berdasarkan Kriteria**

<b>Nama Kasus Uji</b>	Cari Berdasarkan Harga
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih kriteria yang terdiri dari : jenis olahraga, urutan, jenis kelas</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan nama instruktur, jenis olahraga, foto instruktur, sesi, harga, durasi, jumlah murid
<b>Hasil</b>	1. Aplikasi menampilkan nama instruktur, jenis olahraga, foto instruktur, sesi, harga, durasi, jumlah murid
<b>Status</b>	Valid

### 6.2.47 Pengujian *Blackbox* Fungsi Melihat Daftar Pembayaran

Pengujian *blackbox testing* fungsi melihat daftar pembayaran ini didasarkan pada *use case scenario* melihat daftar pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.144 Pengujian *Blackbox* Fungsi Melihat Daftar Pembayaran.

**Tabel 6.145 Pengujian *Blackbox* Fungsi Melihat Daftar Pembayaran**

<b>Nama Kasus Uji</b>	Lihat Daftar Pembayaran
-----------------------	-------------------------

<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Menekan menu “pembayaran”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan daftar table detail pemesan, detail pemesanan, detail pembayaran, status pembayaran, bukti pembayaran dan konfirmasi pembayaran
<b>Hasil</b>	1. Aplikasi menampilkan daftar table detail pemesan, detail pemesanan, detail pembayaran, status pembayaran, bukti pembayaran dan konfirmasi pembayaran
<b>Status</b>	Valid

#### 6.2.48 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan

Pengujian *blackbox testing* fungsi melihat detail pemesanan ini didasarkan pada *use case scenario* melihat detail pemesanan. Hasil pengujiannya dapat dilihat pada Tabel 6.146 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan.

**Tabel 6.147 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan**

<b>Nama Kasus Uji</b>	Lihat Detail Pemesanan
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Menekan field “lihat” pada table detail pemesanan
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan daftar pemesanan yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi jadwal, lokasi
<b>Hasil</b>	1. Aplikasi menampilkan daftar pemesanan yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi jadwal, lokasi
<b>Status</b>	Valid

#### 6.2.49 Pengujian *Blackbox* Fungsi Melihat Bukti Pembayaran

Pengujian *blackbox testing* fungsi melihat bukti pembayaran ini didasarkan pada *use case scenario* melihat bukti pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.148 Pengujian *Blackbox* Fungsi Melihat Bukti Pembayaran.

**Tabel 6.149 Pengujian *Blackbox* Fungsi Melihat Bukti Pembayaran**

<b>Nama Kasus Uji</b>	Lihat Bukti Bayar
<b>Prosedur</b>	1. Membuka aplikasi apiko 2. Menekan field “lihat” pada table bukti pembayaran
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan foto bukti pembayaran
<b>Hasil</b>	1. Aplikasi menampilkan foto bukti pembayaran
<b>Status</b>	Valid

### 6.2.50 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan *Member*

Pengujian *blackbox testing* fungsi melihat detail pemesanan member ini didasarkan pada *use case scenario* melihat detail pemesanan member. Hasil pengujiannya dapat dilihat pada Tabel 6.150 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan *Member*.

**Tabel 6.151 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan *Member***

<b>Nama Kasus Uji</b>	Lihat Detail Pemesanan Member
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Menekan field “lihat” pada table detail pemesanan</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan daftar pemesanan yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi jadwal, lokasi
<b>Hasil</b>	1. Aplikasi menampilkan daftar pemesanan yang berisi nama instruktur, jenis olahraga, durasi, jumlah murid, sesi jadwal, lokasi
<b>Status</b>	Valid

### 6.2.51 Pengujian *Blackbox* Fungsi Melihat Detail Pembayaran

Pengujian *blackbox testing* fungsi melihat detail pembayaran ini didasarkan pada *use case scenario* melihat detail pembayaran. Hasil pengujiannya dapat dilihat pada Tabel 6.152 Pengujian *Blackbox* Fungsi Melihat Detail Pembayaran.

**Tabel 6.153 Pengujian *Blackbox* Fungsi Melihat Detail Pembayaran**

<b>Nama Kasus Uji</b>	Lihat Detail Pembayaran
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Masuk halaman admin</li> <li>3. Menekan field “lihat” pada table detail pembayaran</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan foto bukti pembayaran, logo bank, total pembayaran, kode bayar
<b>Hasil</b>	1. Aplikasi menampilkan foto bukti pembayaran, logo bank, total pembayaran, kode bayar
<b>Status</b>	Valid

### 6.2.52 Pengujian *Blackbox* Fungsi Melihat Detail Pemesan

Pengujian *blackbox testing* fungsi melihat detail pemesanan ini didasarkan pada *use case scenario* melihat detail pemesanan. Hasil pengujiannya dapat dilihat pada Tabel 6.154 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan *Member*.

**Tabel 6.155 Pengujian *Blackbox* Fungsi Melihat Detail Pemesanan *Member***

Nama Kasus Uji	Lihat Detail Pemesan
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>2. Membuka aplikasi apiko</li> <li>3. Masuk halaman admin</li> <li>4. Menekan field “lihat” pada table detail pemesanan</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan nama pemesanan dan no telpon
<b>Hasil</b>	1. Aplikasi menampilkan nama pemesanan dan no telpon
<b>Status</b>	Valid

### 6.2.53 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler

Pengujian *blackbox testing* fungsi melakukan pemesanan kelas reguler ini didasarkan pada *use case scenario* melakukan pemesanan kelas reguler. Hasil pengujiannya dapat dilihat pada Tabel 6.156 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler dan hasil pengujiannya alternatifnya Tabel 6.157 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler dan Tabel 6.158 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler.

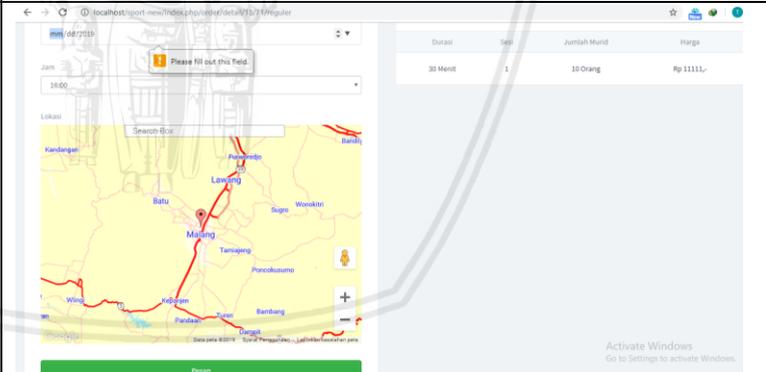
**Tabel 6.159 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler**

Nama Kasus Uji	Pesan Kelas Reguler
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih instruktur yang ingin dipesan</li> <li>3. Menekan tombol “pesan instruktur”</li> <li>4. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga,</li> <li>5. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon, jadwal latihan, jam latihan dan lokasi latihan</li> <li>6. Memilih jadwal latihan</li> <li>7. Memilih jam latihan</li> <li>8. Mengisi lokasi latihan</li> <li>9. Menekan tombol “pesan”</li> </ol>
<b>Hasil yang Diharapkan</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menyimpan data pemesanan</li> <li>2. Aplikasi menampilkan kode pembayaran</li> </ol>

<b>Hasil</b>	1.Aplikasi menyimpan data pemesanan 2.Aplikasi menampilkan kode pembayaran
<b>Status</b>	Valid

**a.Kasus Uji Alternatif1**

**Tabel 6.160 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler**

<b>Nama Kasus Uji</b>	Pesan Kelas Reguler Alternatif 1
<b>Prosedur</b>	1.Membuka aplikasi apiko 2.Memilih instruktur yang ingin dipesan 3.Menekan tombol “pesan instruktur” 4.Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga, 5.Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,jadwal latihan,jam latihan dan lokasi latihan 6.Tidak mengisi seluruh field pemesanan yaitu: jam latihan,jadwal latihan dan lokasi latihan 7.Menekan tombol “pesan”
<b>Hasil yang Diharapkan</b>	1.Aplikasi menampilkan alert “ Please fill out this file”
<b>Hasil</b>	1.Aplikasi menampilkan alert “ Please fill out this file”
<b>Status</b>	Valid
<b>Screenshoot</b>	

**b.Kasus Uji Alternatif 2**

**Tabel 6.161 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Reguler**

<b>Nama Kasus Uji</b>	Pesan Kelas Reguler Alternatif 2
<b>Prosedur</b>	1.Membuka aplikasi apiko 2.Memilih instruktur yang ingin dipesan 3.Menekan tombol “pesan instruktur” 4.Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga, 5.Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,jadwal latihan,jam latihan dan lokasi

	latihan 6. Mengisi seluruh field pemesanan yaitu: jam latihan, jadwal latihan dan lokasi latihan 7. Mengisi jadwal latihan yang sudah dipesan member lain 8. Menekan tombol “pesan”
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan alert “Tanggal yyyy-mm-dd Jam X penuh”
<b>Hasil</b>	1. Aplikasi menampilkan alert “Tanggal yyyy-mm-dd Jam X penuh”
<b>Status</b>	Valid
<b>Screenshot</b>	

### 6.2.54 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket

Pengujian *blackbox testing* fungsi melakukan pemesanan kelas paket ini didasarkan pada *use case scenario* melakukan pemesanan kelas paket. Hasil pengujiannya dapat dilihat pada Tabel 6.162 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket dan hasil pengujian alternatifnya pada Tabel 6.163 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket, Tabel 6.164 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket dan Tabel 6.165 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket.

**Tabel 6.166 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket**

<b>Nama Kasus Uji</b>	Pesan Kelas Paket
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih instruktur yang ingin dipesan</li> <li>3. Menekan tombol “pesan instruktur”</li> <li>4. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga,</li> <li>5. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon, hari latihan, jam latihan dan lokasi latihan</li> <li>6. Memilih hari latihan</li> <li>7. Memilih jam latihan</li> <li>8. Mengisi lokasi latihan</li> <li>9. Menekan tombol “pesan”</li> </ol>
<b>Hasil yang Diharapkan</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menyimpan data pemesanan</li> <li>2. Aplikasi menampilkan kode pembayaran</li> </ol>
<b>Hasil</b>	<ol style="list-style-type: none"> <li>1. Aplikasi menyimpan data pemesanan</li> <li>2. Aplikasi menampilkan kode pembayaran</li> </ol>

Status	Valid
--------	-------

**a.Kasus Uji Alternatif1**

**Tabel 6.167 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket**

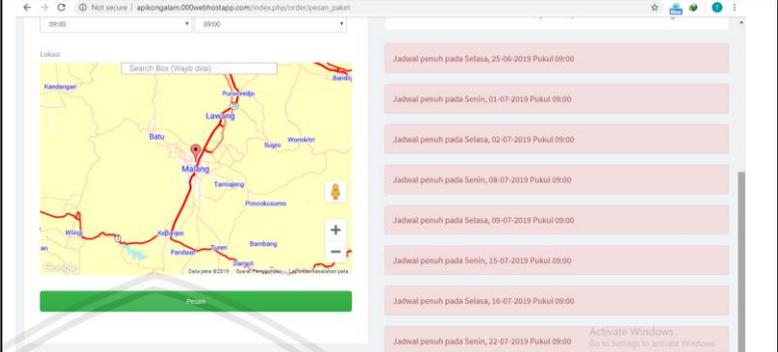
<b>Nama Kasus Uji</b>	Pesan Kelas Paket Alternatif 1
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1.Membuka aplikasi apiko</li> <li>2.Memilih instruktur yang ingin dipesan</li> <li>3.Menekan tombol “pesan instruktur”</li> <li>4.Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga,</li> <li>5.Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,hari latihan,jam latihan dan lokasi latihan</li> <li>6.Tidak mengisi seluruh field pemesanan yaitu: jam latihan,hari latihan dan lokasi latihan</li> <li>7.Menekan tombol “pesan”</li> </ol>
<b>Hasil yang Diharapkan</b>	1.Aplikasi menampilkan alert “ Please fill out this file”
<b>Hasil</b>	1.Aplikasi menampilkan alert “ Please fill out this file”
<b>Status</b>	Valid
<b>Screenshot</b>	

**b.Kasus Uji Alternatif2**

**Tabel 6.168 Pengujian *Blackbox* Fungsi Melakukan Pemesanan Kelas Paket**

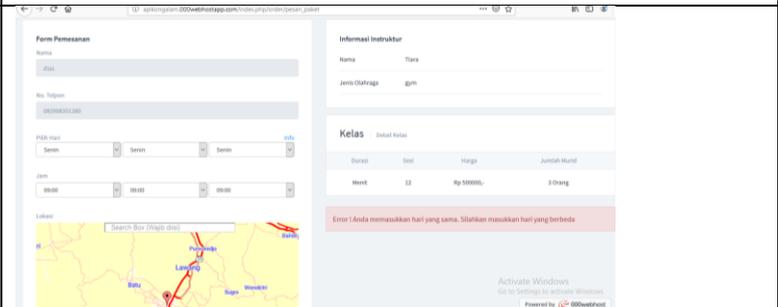
<b>Nama Kasus Uji</b>	Pesan Kelas Paket Alternatif 2
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1.Membuka aplikasi apiko</li> <li>2.Memilih instruktur yang ingin dipesan</li> <li>3.Menekan tombol “pesan instruktur”</li> <li>4.Aplikasi menampilkan detail pemesanan berisi nama,jenis olahraga,durasi,jumlah murid,sesi,harga,</li> <li>5.Aplikasi menampilkan form pemesanan yang berisi nama,no telepon,hari latihan,jam latihan dan lokasi latihan</li> <li>6.Mengisi seluruh field pemesanan yaitu: jam latihan,hari latihan dan lokasi latihan</li> <li>7.Mengisi jadwal latihan yang sudah dipesan member lain</li> <li>8.Menekan tombol “pesan”</li> </ol>



<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Jadwal penuh pada nama hari, dd-mm-yyyy Pukul X”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “Jadwal penuh pada nama hari, dd-mm-yyyy Pukul X”
<b>Status</b>	Valid
<b>Screenshot</b>	

**c. Kasus Uji Alternatif 3**

**Tabel 6.169 Pengujian Blackbox Fungsi Melakukan Pemesanan Kelas Paket**

<b>Nama Kasus Uji</b>	Pesan Kelas Paket Alternatif 3
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Membuka aplikasi apiko</li> <li>2. Memilih instruktur yang ingin dipesan</li> <li>3. Menekan tombol “pesan instruktur”</li> <li>4. Aplikasi menampilkan detail pemesanan berisi nama, jenis olahraga, durasi, jumlah murid, sesi, harga,</li> <li>5. Aplikasi menampilkan form pemesanan yang berisi nama, no telepon, hari latihan, jam latihan dan lokasi latihan</li> <li>6. Mengisi seluruh field pemesanan yaitu: jam latihan, hari latihan dan lokasi latihan</li> <li>7. Mengisi paket hari yang sama untuk satu minggu pada paket 8/12 sesi latihan</li> <li>8. Menekan tombol “pesan”</li> </ol>
<b>Hasil yang Diharapkan</b>	1. Aplikasi menampilkan pesan “Error ! Anda memasukkan hari yang sama. Silahkan masukkan hari yang berbeda”
<b>Hasil</b>	1. Aplikasi menampilkan pesan “Error ! Anda memasukkan hari yang sama. Silahkan masukkan hari yang berbeda”
<b>Status</b>	Valid
<b>Screenshot</b>	

### 6.1.55 Hasil Analisis Pengujian Blackbox Testing

Dari hasil pengujian 54 fungsionalitas beserta kasus uji alternatifnya diatas diperoleh hasil valid untuk semua fungsi dan kasus uji alternatifnya, sehingga tingkat validasi yang diperoleh untuk pengujian ini 100% valid.

### 6.3 Pengujian Non Fungsional

Browser	IE	Edge	Firefox	Safari	Opera	Chrome	iOS			Android	
Version	11	18	66	12	60	74	≤ 10	11	12	≤ 3	4*
Critical Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Major Issues	⊘	⊘	⊘	⊘	⊘	⊘	⊘	⊘	⊘	⊘	⊘
Minor Issues	⊘										

Gambar 6.7 Hasil Pengujian Non Fungsional

#### 6.3.1 Hasil Analisis Pengujian Non Fungsional

Dari hasil pengujian non fungsional yang sudah dilakukan dengan tools *SortSite* diperoleh hasil bahwa sistem dapat digunakan pada *Internet Explore* ,*Edge*, *Firefox*, *Safari*, *Opera*, *Chrome*, *iOS* dan *Android* sehingga dari hasil yang didapat disimpulkan bahwa sistem sudah memenuhi kebutuhan fungsional kategori *Compatibility*.

## BAB 7 PENUTUP

### 7.1 Kesimpulan

Berdasarkan hasil dari semua proses yang sudah dilakukan dalam pengembangan aplikasi, dapat disimpulkan sebagai berikut:

Pada proses analisis kebutuhan pengembangan aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web dilakukan dengan memberikan kuisisioner pada pengguna sistem yang merupakan 70 responden di Kota Malang terdiri dari 31% laki dan 69% perempuan dengan tipe pekerjaan 67,6 mahasiswa 15,5% pelajar dan 16,9% umum. Pada proses penyebaran kuisisioner bertujuan untuk identifikasi setiap kebutuhan yang mereka inginkan terhadap masalah yang ada serta melakukan wawancara terhadap 14 instruktur olahraga Kota Malang untuk mendapatkan informasi tambahan terkait proses bisnis yang dibutuhkan sistem, data profil instruktur olahraga serta kendala yang terjadi dalam proses pemasaran jasanya. Sehingga dari proses rekayasa kebutuhan ini diperoleh 54 kebutuhan fungsional dengan 4 aktor yang terlibat pada sistem yang mana 3 dari nya merupakan aktor utama dan diperoleh 1 kebutuhan non fungsional aplikasi.

Proses perancangan aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web dilakukan dengan beberapa tahapan antara lain: perancangan *sequence diagram*, perancangan *class diagram*, perancangan *interface* dan perancangan *algoritma*. Yang mana dari hasil keseluruhan perancangan tersebut digunakan untuk tahapan implementasi yang terbagi menjadi implementasi *interface* yang mengacu pada perancangan *interface* dan implementasi *source code* mengacu pada perancangan *algoritma*. Seluruh hasil perancangan berhasil diimplementasikan menjadi aplikasi pemesanan instruktur olahraga kota malang berbasis web. Pada proses implementasi aplikasi ini menggunakan bahasa pemrograman PHP dan *Framework Codeigniter*, dengan database *MySQL* serta *google maps api* untuk memilih lokasi latihan dan *email gateway* untuk notifikasi jadwal.

Pengujian aplikasi dilakukan dengan 2 cara yaitu: pengujian *whitebox testing* dan *blackbox testing*. Pengujian *whitebox* dilakukan dengan metode *cyclomatic complexity* dengan mengambil sampel 5 method dan didapatkan hasil probabilitas rendah untuk *error*. Pengujian *blackbox* dilakukan dengan menguji 54 fungsionalitas dengan kasus uji alternatifnya masing-masing dan diperoleh hasil 100% valid. Pengujian non fungsional dilakukan menggunakan *tools SortSite* dan diperoleh hasil yang memenuhi kriteria kebutuhan *Compatibility*.

### 7.2 Saran

Dari hasil pengembangan aplikasi pemesanan jasa instruktur olahraga kota malang berbasis web, terdapat saran untuk penelitian selanjutnya yaitu:

1. Menjadikan aplikasi berbasis *android* sehingga lebih efektif dalam digunakan.

## DAFTAR PUSTAKA

- Badie Uddin.(2017). Aplikasi Pemesanan Catering Menggunakan SMS Gateway Berbasis Web, Jurnal Telematika, Vol 12, No 1.
- Bimo, Sunarfrihantono.(2002). PHP Dan MySQL Untuk Web. Yogyakarta : Andi
- Causa Prima Wijaya, Kodrat Iman Satoto, dan R. Rizal Isnanto.(2013). Perancangan Sistem Informasi Pemesanan Tiket Travel Berbasis Web,
- ChokyWijaya. (2011). Six Pack-Tips Ampuh Membentuk Tubuh Ideal. Jakarta: Second Hope
- Faya Mahdia, Fiftin Noviyanto.(2013). Pemanfaatan Google Maps Api Untuk Pembangunan Sistem Informasi Manajemen Bantuan Logistik Pasca Bencana Alam Berbasis Mobile Web ( Studi Kasus : Badan Penanggulangan Bencana Daerah Kota Yogyakarta ), Jurnal Sarjana Teknik Informatika, Vol1, No 1
- Febriana, Shakuntala. (2011). Representasi Citra Diri Bagi Personal Trainer Celebrity Fitness La Piazza, Depok: Universitas Indonesia.
- Harri Singgih Pratikto, Suraya, Edhy Sutanta.(2014). Sistem Pencarian dan Pemesnan Rumah Kos Menggunakan Sistem Informasi Geografi(SIG), Jurnal SCRIPT , Vol 1, No 2.
- Jhonsen.(2004). Web Designer Untuk Pemula.Jakarta : Elex Media Komputindo Kelompok Gramedia, Anggota Ikapi.
- Jhon, W.Satzinger.(2010). Software Analysis and Design in a Changging Wolrd Edition 6<sup>th</sup>,America
- Rivaldi Masuara, Rizal Sengkey, Virginia Tulenan.(2015). Rancang Bangun e-Ticketing Bioskop Studio 21 Manado Berbasis Multiplatform, E-journal Teknik Elektro dan Komputer
- Roger,S.Pressman.(2010). Software Engineering: A Practitioner's Approach, McGraw Hill/Irwin, New York.
- Sommerville, I. (2011). Software Engineering 9th Edition. America.
- Sucianti Saiful, Arisandy Ambarita.(2017). Pembuatan Aplikasi Web Pencarian Jasa Pemantu Rumah Tangga(PRT) DiKota Ternate, Indonesian Journal on Information System, Vol 2, No 2.
- Teddy Marcus Zakaria, Oscar Wongso.(2010). Studi dan Implementasi Teknologi Flashdisk dan Email Gateway dalam Penyewaan Alat pada Perusahaan X, Jurnal Informatika, Vol 6, No 2.

Whitten, Bentley. (2007). System analysis & Design Methods, McGraw Hill/Irwin, New York.

Widianti, Sri.( 2000). Pengantar Basis Data.Jakarta : Fajar.

Yoki Firmansyah, Udi.(2018). Penerapan Metode SDLC Waterfall Dalam Pembuatan Sistem Informasi Akademik Berbasis Web Studi Kasus Pondok Pesantren Al-Habi Sholeh Kabupaten Kubu Raya, Kalimantan Barat, Jurnal Teknologi & Manajemen Informatika,Vol 4, No 1.

Yusi Ardi Binarso, Eko Adi Sarwoko, Nurdin Bahtiar.(2012). Pembangunan Sistem Informasi Alumni Berbasis Web Pada Program Studi Teknik Informatika Universitas Diponegoro, Journal of Informatics and technology, Vol 1, No 1.



### Lampiran 1 Surat Telah Melakukan Wawancara



## Lampiran 2 Hasil Wawancara

Berikut merupakan lampiran dari materi wawancara untuk mendukung penelitian ini, yang telah dilaksanakan pada:

Hari/Tanggal : 11 Februari – 18 Maret 2019  
 Waktu : 16.00 – 17.00 Wib  
 Tempat : UB Sport Center (Jl. Terusan Cibogo No.1, Penanggungan, Klojen, Kota Malang, Jawa Timur 65113)

Narasumber : Seluruh Instruktur Olahraga Gym, Zumba, Aerobik, Yoga

Berikut adalah lampiran dari materi wawancara untuk mendukung penelitian ini. Materi wawancara di bagi menjadi 3, berikut merupakan bagian-bagian tersebut:

### 1. Biodata dan Paket layanan jasa instruktur olahraga

Materi wawancara ini bertujuan untuk mengetahui tentang biodata yang terdiri dari nama, no telpon, alamat, lama melatih, pengalaman melatih, sertifikasi atau penghargaan, keahlian lain, dan mengetahui paket layanan jasa nyayang terdiri dari murid, jumlah sesi, dan harga.

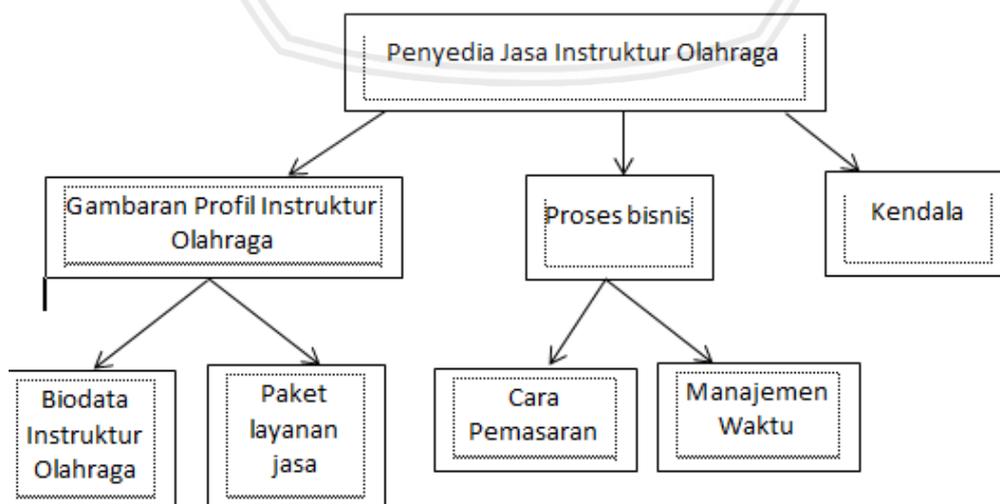
### 2. Proses Bisnis instruktur olahraga

Materi wawancara ini bertujuan untuk mengetahui tentang proses bisnis yang dilakukan instruktur olahraga mulai dari cara pemasarannya dan manajemen waktunya.

### 3. Penutup

Materi wawancara ini bertujuan untuk mengetahui kendala yang dihadapi Instruktur olahraga dalam menjalankan bisnisnya.

Berikut merupakan susunan tree/work structured terkait wawancara dengan narasumber yaitu penyedia jasa Instruktur Olahraga di Kota Malang yang dapat dilihat pada gambar dibawah ini.



Gambar A tree/work structured wawancara

### A.1 Pertanyaan Terkait Pengisian Data Profil Instruktur Olahraga serta mengetahui paket layanan jasa nya

Materi wawancara ini bertujuan untuk memperoleh data profil instruktur olahraga di Kota Malang serta paket layanannya.

No.	Pertanyaan	Jawaban
1.	Siapakah nama anda?	
2.	Adakah nomer yang bisa dihubungi?	
3.	Dimanakah alamat anda?	
4.	Berapa lama anda sudah melatih?	
5.	Dimana saja anda memiliki pengalaman melatih selain di UB Sport Center?	
6.	Apakah anda memiliki sertifikasi maupun penghargaan?	
7.	Apakah anda memiliki keahlian lain dalam bidang olahraga lain?	
8.	Berapakah jumlah murid anda sekarang ini?	
9.	Berapa jumlah sesi dalam latihan yang anda berikan dalam satu bulan?	
10.	Berapa harga latihan anda tiap sesinya?	

## A.2 Jawaban Terkait Data Profil Instruktur Olahraga beserta paket layanan jasanya

Materi wawancara ini bertujuan untuk mengetahui data profil siapa saja yang menjadi instruktur beserta paket layanannya olahraga di Kota Malang

No	Nama	No tlp	Alamat	Lama melatih	Pengalaman melatih	Sertifikasi atau Penghargaan	Jumlah Murid	Keahlian Lain	Jumlah Sesi	Harga
1	Zainul Afriansyah	081331158 201	Jln Bendungan Sengguruh No 15,Rt 03,Rw 07	6 bulan	Personal Trainer Private	Body Contes	30	-	12	500.000
2	Joko Listyawan	081233854 94	Jln Bendungan Tangga No 14	10 tahun	Personal Trainer di Club House,Gym Apartemen, di Pasuruan	-Juara 3 Junior Binaraga Nasional -Juara 1 Binaraga Jawa Bali -Juara 2 Binaraga Kejurprof -Juara 1 IFBB Junior Se Asia -Sertifikat Pelatih Fitnes	5	-	12	750.000

						Nasional -Sertifikat Pelatih Pambsi Tingkat Profinsi				
3	Andi Kurniawan	082321413 529	Jln Setia Budi Utara No 63	15 tahun	Personal Trainer di Komet Gym Sby,Lavayet, Mustika Blitar	-Juara 2 Open Nasional -Juara 2 Kejurda -Juara 2 Jatim -Sertifikat Pelatih Fitnes Nasional	3	-	12	600.000
4	Vety Tunjungsari	085233285 775	Jln Bromo No 29	2,5 tahun	Pelatih Yoga di Tidar	-Pelatih Yoga dari Sanggar Engar Yoga Milik Bu Riana	50	-	1	125.000- 150.000
									4	200.000
5	Siti Aminah	085335646 143	Jln Mangunsa ngkoso	7 tahun	-Pelatih private yoga	-Sertifikat Pelatih Yoga	15	-Aerobik -Pilates	1	250.000- 300.000

			Gang 3 No 15 Rt05 Rw06		-Pelatih yoga disanggar	-Sertifikat Pelatih Aerobik -Pengisi acara diworkshop -Ajang lomba diaerobik jawa bali			4	300.000
6	Septyane Tuegi Purnamasari	082234002081	Jln Setia Budi Utara No 63	4 tahun	-Pelatih zumba di Gym Power,Calosa,Max,Hars  -Pelatih zumba di Sanggar	-Sertifikat Pelatih Zumba dan Aerobik  -Juri Lomba Aerobik  -Ajang Lomba Aerobik Nasional	50	-Aerobik	1	150.000-500.000
									4	200.000
7	Neni Agus Hartawati	082234287667	Jln Suropati 2 No 7	25 tahun	-Pelatih Aerobik di Gym Max,Best,Arana,Bogor	-Sertifikat Pelatih Aerobik Nasional  -Sertifikat Melatih Di FK UM  -Ajang lomba seJawa Bali	50	-Body language	1	100.000-250.000
									4	200.000

						-Ajang lomba Nasional				
8	Titik Indah	081230218 733	Jln Kebun Jeruk No 7	4 tahun	Pelatih Aerobik di Best Gym,di Sanggar daerah Borobudur, Cengger ayam,Candi Mendut	-Sertifikat Pelatih Aerobik Nasional  -Ajang lomba antar kota,dan Nasional	30	-	1	200.000- 500.000
									4	200.000
9	Himmatul Yani	089841829 90	Jln Sanan Gang 8 No 20	21 tahun	Pelatih Aerobik di Best Gym,di Sanggar daerah Borobudur	-Sertifikat Pelatih Aerobik Nasional  -Ajang Lomba se Jawa Bali	50	Zumba	1	150.000- 200.000
									4	200.000
10	Wenny	081216955 221	Jln Jombang Gang 2 No 25	19 tahun	Pelatih Aerobik di Calosa,Egy Gym,Ika,do Sanggar Nona,Ida	-Sertifikat Pelatih  -Sertifikat Kelas Prof  -Ajang lomba Nasional	50	-Zumba  -Body Languange	1	150.000- 250.000
									4	200.000

11	Suiadi	085649608 915	Jln Raya Candi V No 5	10 tahun	Pelatih renang di sanggar,sd,s mp,sma,ma hasiswa,atlit lomba	-Lomba kejuaraan ajang renang tingkat jawa timur  -Guru olahraga  -Sertifikat pelatih	20	-	8	500.000
12	Christ	085725532 623	Jln Candi Panggung 2 No 7	5 tahun	Pelatih renang di sanggar,ma hasiswa,um um,atlit untuk kejuaraan	-Lomba kejuaraan renang tingkat jawa bali  -Event lomba lainnya  -Sertifikat pelatih	30	-	8	450.000
									4	250.000
13	Novian Rohmat	081233878 481	Jln Terusan Dieng No 22	2 tahun	Pelatih renang di sanggar sd,smp,sma, umum	-Lomba kejuaraan renang antar kerasidenan  -Event lomba lainnya  -Guru olahraga	50	-	1	20.000
									4	75.000

14	Audi	081333055 518	Jln Supriadi	1 tahun	Pelatih bela diri taekwondo, pelatih di sanggat,atlit umum untuk lomba	-Lomba kejuaraan bela diri taekwondo tingkat jawa bali -Ajang event lomba lainnya di bela diri -Sertifikat pelatih bela diri	10	Taekwondo	1	100.000
----	------	------------------	--------------	---------	--	--	----	-----------	---	---------



### A.3 Pertanyaan Terkait Proses Bisnis Instruktur yang terdiri dari cara pemasaran dan manajemen waktu pelayanan jasanya

Materi wawancara ini bertujuan untuk memperoleh data proses bisnis yang meliputi cara pemasaran dan manajemen pelayanan jasa instruktur olahraga di Kota Malang

No	Pertanyaan	Jawaban
1	Dimanakah anda mempromosikan jasa layanan instruktur olahraga anda?	
2	Bagaimana cara pelanggan dapat menghubungi dan melakukan pemesanan jasa layanan instruktur olahraga anda?	
3	Bagaimana proses bisnis pemesanan jasa yang selama ini berlangsung?	
4	Apakah anda menerima permintaan pelanggan untuk menjadi instruktur olahraga di luar kota?	
5	Berapa waktu yang di butuhkan untuk menyelesaikan paket latihan tersebut?	
6	Siapa saja yang biasanya memakai jasa instruktur olahraga anda selain per orang an?	
7	Hari dan jam berapakah anda melakukan pelayanan jasa instruktur olahraga kepada pelanggan?	
8	Dimana saja biasanya anda memberikan latihan olahraga tersebut?	

**A.4 Jawaban Terkait Proses Bisnis Instruktur yang terdiri dari cara pemasaran dan manajemen waktu pelayanan jasanya**

Materi wawancara ini bertujuan untuk memperoleh data proses bisnis yang meliputi cara pemasaran dan manajemen pelayanan jasa instruktur olahraga di Kota Malang.

No	Nama	Jenis Instruktur Olahraga	Media Promosi Jasa	Cara Pemesanan Jasa	Mekanisme Proses Pemesanan Jasa	Permintaan melatih diluar kota	Lama waktu berlatih	Pemakai Jasa Instruktur Olahraga	Jadwal Latihan	Tempat latihan
1	Zainul Afriansyah	Gym	-Instagram -Whatsup	Dm Ig dan hubung no W.A	1.Pelanggan menghubungi i melalui DM instagram lalu ke nomor WhatsApp  2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan  3.Kemudian instruktur	Tidak  (area kota Malang saja)	60  menit	-Per orang  -Dua orang	Senin- Minggu  (08.00- 20.00  Jika ada lomba/e vent maka tidak ada kelas	Indoor( Tempat Gym)

				<p>memberikan informasi terkait jadwal, harga, dan layanan paket jasa</p> <p>4. Pelanggan memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang dibutuhkan</p>					
--	--	--	--	---	--	--	--	--	--

				<p>yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p> <p>7.Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka</p>					
--	--	--	--	--	--	--	--	--	--

					pembayaran dilakukan diawal saat latihan pertama					
2	Joko Listyawan	Gym	-Instagram -Whatsup -Facebook	Langsung hubungi no W.A	1.Pelanggan menghubungi i melalui DM instagram/facebok lalu ke nomor WhatsApp  2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan  3.Kemudian instruktur memberikan informasi terkait	Tidak (area kota Malang saja)	60 menit	-Per orang	Senin-Minggu (08.00-20.00  Jika ada lomba /eventm aka tidak ada kelas	Indoor( Tempat Gym)

				<p>jadwal, harga, dan layanan paket jasa</p> <p>4. Pelanggan memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang dibutuhkan yaitu nama, tempat latihan,</p>					
--	--	--	--	--	--	--	--	--	--

				<p>no.HP, tanggal pemesanan, paket layanan jasa.</p> <p>7.Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal</p>					
--	--	--	--	--	--	--	--	--	--

					saat latihan pertama					
3	Andi Kurniawan	Gym	-Whatsup -Facebook	Langsung hubungi no W.A	<p>1.Pelanggan menghubungi i melalui ke nomor WhatsApp</p> <p>2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan</p> <p>3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa</p> <p>4.Pelanggan</p>	Tidak (area kota Malang saja)	60 menit	-Per orang	<p>Senin-Minggu (08.00-20.00</p> <p>Jika ada lomba/e vent maka tidak ada kelas</p>	Indoor( Tempat Gym)

				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7. Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8. Kemudian pelanggan dapat melakukan pembayaran ditempat/ sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan, atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--

4	Vety Tunjungsari	Yoga	-Instagram -Facebook -Whatsup -Mulut ke mulut	Dm ig/inbox fb lalu hubungi no w.a	1.Pelanggan menghubungi i melalui DM instagram/facebo ok lalu ke nomor WhatsApp  2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan  3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa  4.Pelanggan	Tidak (area kota Malang saja)	60 menit	-Per orang -Dua orang -Group	Senin-Minggu (08.00-19.00  Jika ada lomba/event maka tidak ada kelas	Indoor( Tempat Gym,Sanggar,Rumah,Sekolah)
---	------------------	------	--	---	--	-------------------------------	----------	------------------------------------	--	---

				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7.Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--

5	Siti Aminah	Yoga	-Instagram -Facebook -Whatsup -Mulut ke Mulut	Dm ig/inbox fb lalu hubungi no w.a	1.Pelanggan menghubungi i melalui DM instagram/facebok lalu ke nomor WhatsApp 2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan 3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa 4.Pelanggan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group	Senin-Minggu (08.00-17.00 Jika ada lomba maka tidak ada kelas	Indoor( Tempat Gym,Sanggar,Rumah,Sekolah)
---	-------------	------	--	---	--	--	----------	----------------------	--	---

				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7. Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8. Kemudian pelanggan dapat melakukan pembayaran ditempat/ sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan, atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



6	Septyane Tuegi Purnamasari	Zumba	-Instagram -Facebook -Whatsup -Mulut ke Mulut	Dm ig/inbox fb lalu hubungi no w.a	1.Pelanggan menghubungi i melalui DM instagram/facebook lalu ke nomor WhatsApp 2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan 3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa 4.Pelanggan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group -Instansi/perusahaan	Senin-Minggu (08.00-17.00 Jika ada lomba/event maka tidak ada kelas	Indoor( Tempat Gym,Sanggar,Rumah,Sekolah) Outdoor(Lapangan)
---	----------------------------	-------	--	---------------------------------------	---	--	----------	--	--	--

				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7. Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8. Kemudian pelanggan dapat melakukan pembayaran ditempat/ sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan, atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



7	Neni Agus Hartawati	Aerobik	-Facebook -Whatsup -Mulut ke Mulut	Langsung hubungi no W.A	<p>1.Pelanggan menghubungi i melalui nomor WhatsApp</p> <p>2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan</p> <p>3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa</p> <p>4.Pelanggan memilih paket jasa dan</p>	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group -Instansi/ perusahaan	<p>Senin-Minggu (08.00-17.00</p> <p>Jika ada lomba/event maka tidak ada kelas</p>	Indoor( Tempat Gym,Sa nggar,R umah,S ekolah) Outdoo r(Lapan gan)
---	---------------------	---------	--	-------------------------	---	--	----------	--	---	---

				<p>harga,lalu mulai melakukan perjanjian terkait jadwal</p> <p>5.Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6.Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p> <p>7.Instruktur memasukkan</p>					
--	--	--	--	---	--	--	--	--	--

					<p>jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--

8	Titik Indah	Aerobik	-Instagram -Facebook -Whatsup -Mulut ke Mulut	Dm ig/inbox fb lalu hubungi no w.a	1.Pelanggan menghubungi melalui DM instagram lalu ke nomor WhatsApp 2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan 3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa 4.Pelanggan memilih paket	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group -Instansi/ perusahaan	Senin-Minggu (08.00-17.00 Jika ada lomba/event maka tidak ada kelas	Indoor( Tempat Gym,Sanggar,Rumah,Sekolah) Outdoor(Lapangan)
---	-------------	---------	--	---	--	--	----------	--	---	--

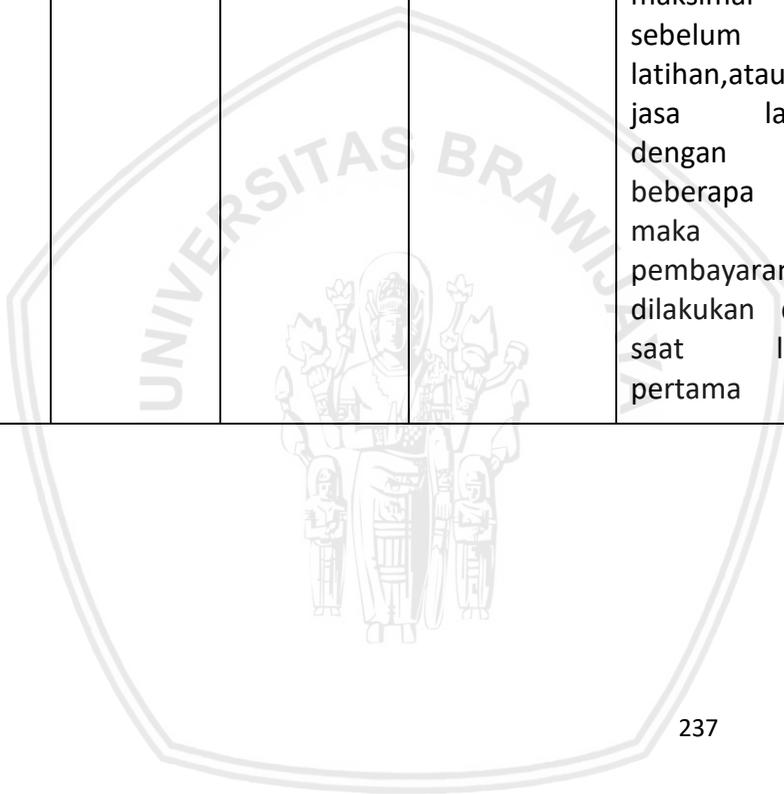
				<p>jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p> <p>7. Instruktur</p>					
--	--	--	--	---	--	--	--	--	--

				<p>memasukkan jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	---	--	--	--	--	--

9	Himmatul Yani	Aerobik	-Facebook -Whatsup -Mulut ke Mulut	Langsung hubungi no W.A	1.Pelanggan menghubungi i melalui ke nomor WhatsApp  2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan  3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa  4.Pelanggan memilih paket jasa dan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group -Instansi/ perusahaan	Senin-Minggu (08.00-17.00  Jika ada lomba/event maka tidak ada kelas	Indoor( Tempat Gym,Sanggar,Rumah,Sekolah)  Outdoor(Lapangan)
---	---------------	---------	--	-------------------------	---	--	----------	---	--	--

				<p>harga,lalu mulai melakukan perjanjian terkait jadwal</p> <p>5.Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6.Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p> <p>7.Instruktur memasukkan</p>					
--	--	--	--	---	--	--	--	--	--

					<p>jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



10	Wenny	Aerobik	-Facebook -Whatsup -Instagram -Mulut ke Mulut	-Dm instagram, message facebook atau Langsung hubungi no W.A	1. Pelanggan menghubungi i melalui DM instagram/facebook oklalu ke nomor WhatsApp 2. Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga, serta menanyakan terkait harga dan paket layanan 3. Kemudian instruktur memberikan informasi terkait jadwal, harga, dan layanan paket jasa 4. Pelanggan	Tidak (area kota Malang saja) namun jika ada event lomba bias keluar kota	60 menit	Per orang -Group -Instansi/ perusahaan	Senin-Minggu (08.00-17.00 Jika ada lomba/event maka tidak ada kelas	Indoor( Tempat Gym, Sanggar, Rumah, Sekolah) Outdoor(Lapangan)
----	-------	---------	--	--	--	---	----------	--	--	---

				<p>memilih paket jasa dan harga,lalu mulai melakukan perjanjian terkait jadwal</p> <p>5.Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6.Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	---	--	--	--	--	--

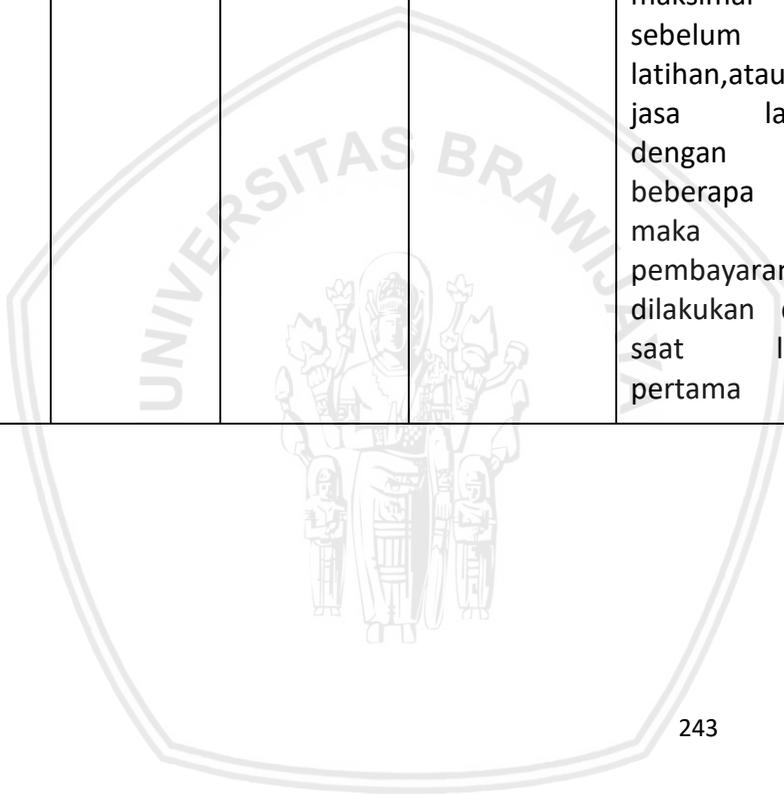
					<p>7. Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8. Kemudian pelanggan dapat melakukan pembayaran ditempat/ sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan, atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



11	Suiadi	Renang	-Facebook -Whatsup -Mulut ke Mulut	Langsung hubungi no W.A	1.Pelanggan menghubungi i melalui ke nomor WhatsApp  2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan  3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa  4.Pelanggan memilih paket jasa dan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group	Senin-Minggu (08.00-17.00  Jika harus mendampingi lomba atau urusan mendadak kelas tidak ada	Kolam renang indoor dan outdoor
----	--------	--------	--	-------------------------	---	--	----------	----------------------	--	---------------------------------

				<p>harga,lalu mulai melakukan perjanjian terkait jadwal</p> <p>5.Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6.Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p> <p>7.Instruktur memasukkan</p>					
--	--	--	--	---	--	--	--	--	--

					<p>jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



12	Christ	Renang	-Facebook -Whatsup -Mulut ke Mulut -Instagram	-Dm instagram,m essage facebook atau Langsung hubungi no W.A	1.Pelanggan menghubungi i melalui DM instagram/facebo oklalu ke nomor WhatsApp 2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan 3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa 4.Pelanggan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group	Senin-Minggu (08.00-17.00 Jika harus mendam pingi lomba atau lomba maupun event tidak ada kelas	Kolam renang indoor dan outdoor
----	--------	--------	--	--	--	--	----------	----------------------	---	---------------------------------

				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7.Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8.Kemudian pelanggan dapat melakukan pembayaran ditempat/sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan,atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



13	Novian Rohmat	Renang	-Facebook -Whatsup -Mulut ke Mulut -Instagram	-Dm instagram,m essage facebook atau Langsung hubungi no W.A	1.Pelanggan menghubungi i melalui DM instagram/facebo oklalu ke nomor WhatsApp 2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan 3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa 4.Pelanggan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group	Senin- Minggu (08.00- 17.00 Jika harus mendam pingi lomba atau urusan mendad ak kelas tidak ada	Kolam renang indoor dan outdoo r
----	---------------	--------	--	---	--	---	-------------	----------------------	--	---

				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang dibutuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7. Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8. Kemudian pelanggan dapat melakukan pembayaran ditempat/ sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan, atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--



14	Audi	Muay Thai	-Facebook -Whatsup -Mulut ke Mulut -Instagram	-Dm instagram, message facebook atau Langsung hubungi no W.A	1.Pelanggan menghubungi i melalui DM instagram/facebook oklalu ke nomor WhatsApp 2.Pelanggan memberitahu dan menanyakan jadwal mereka ingin menggunakan jasa instruktur olahraga,serta menanyakan terkait harga dan paket layanan 3.Kemudian instruktur memberikan informasi terkait jadwal,harga,dan layanan paket jasa 4.Pelanggan	Tidak (area kota Malang saja)namun jika ada event lomba bias keluar kota	60 menit	-Per orang -Group	Senin-Minggu (08.00-20.00 Jika ada lomba/event maka tidak ada kelas	Tempat indoor( sanggar ,rumah) dan outdoor(lapan gan)
----	------	-----------	--	--	---	--	----------	----------------------	--	---

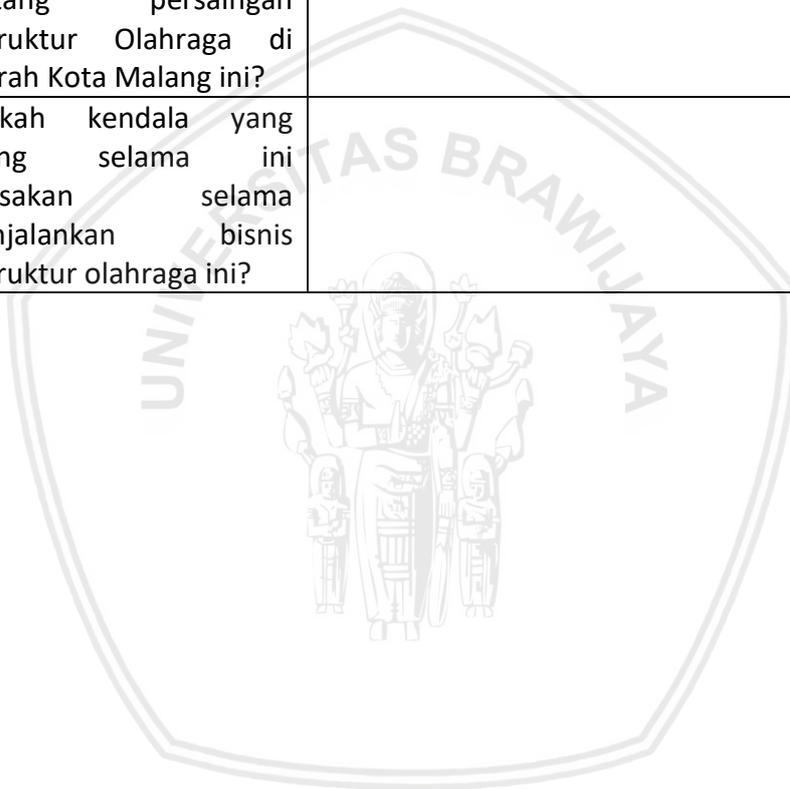
				<p>memilih paket jasa dan harga, lalu mulai melakukan perjanjian terkait jadwal</p> <p>5. Apabila jadwal yang diinginkan pelanggan kosong dapat dilakukan pemesanan terkait jadwal, apabila tidak ada jadwal kosong maka pemesanan tidak dilanjutkan</p> <p>6. Pelanggan mengisi informasi yang butuhkan yaitu nama, tempat latihan, no.HP, tanggal pemesanan, paket layanan jasa.</p>					
--	--	--	--	--	--	--	--	--	--

					<p>7. Instruktur memasukkan jadwal pelanggan dalam note nya</p> <p>8. Kemudian pelanggan dapat melakukan pembayaran ditempat/ sehabis melakukan kegiatan latihan atau jika ingin mentransfer maksimal h-1 sebelum latihan, atau jika jasa layanan dengan paket beberapa sesi maka pembayaran dilakukan diawal saat latihan pertama</p>					
--	--	--	--	--	--	--	--	--	--	--

#### A.5 Penutup(pertanyaan terkait kendala yang dihadapi instruktur olahraga dalam menjalankan bisnisnya)

Materi wawancara ini bertujuan untuk mengetahui kendala yang dihadapi instruktur olahraga dalam menjalankan bisnisnya.

No	Pertanyaan	Jawaban
1	Bagaimana pendapat anda tentang perkembangan dunia instruktur olahraga di Kota Malang?	
2	Bagaimana menurut anda tentang persaingan Instruktur Olahraga di daerah Kota Malang ini?	
3	Apakah kendala yang sering selama ini dirasakan selama menjalankan bisnis instruktur olahraga ini?	



### A.6 Jawaban Penutup(pertanyaan terkait kendala yang dihadapi instruktur olahraga dalam menjalankan bisnisnya)

Materi wawancara ini bertujuan untuk mengetahui kendala yang dihadapi instruktur olahraga dalam menjalankan bisnisnya.

No	Nama	Jenis Instruktur Olahraga	Perkembangan dunia jasa instruktur olahraga	Persaingan dalam dunia jasa instruktur olahraga	Kendala dalam dunia pemasaran jasa instruktur olahraga
1	Zainul Afriansyah	Gym	Sangat pesat,sekarang ini banyak instruktur yang tidak harus berasal dari background pendidikan olahraga dapat menjadi instruktur olahraga,pendidikannya pun tidak perlu tinggi	Sudah banyak instruktur yang muda dan bertalenta,juga sudah banyak instruktur yang melalui media social untuk memberi tentor latihan	Penyebaran informasi jasa masih sebatas dari instagram
2	Joko Listyawan	Gym	Sangat pesat karena sekarang semua bisa menjadi instruktur kalau mau belajar,tidak harus sekolah khusus,bisa belajar otodidak maupun mempelajari dari social media	Banyak sekali instruktur yang berpotensi sekalipun mereka baru merintis pengalaman sebentar dan banyak juga instruktur yang tidak ada batasan gender sehingga kita akan bersaing dari pengalaman dan sertifikat yang sudah didapat	Penyebaran informasi jasa masih sebatas dari instagram dan facebook saja
3	Andi Kurniawan	Gym	Perkembangannya sangat pesat apalagi sekarang terdapat berbagai macam olahraga untuk tujuan sendiri-sendiri,instruktur	Banyak sekali instruktur olahraga yang mulainya memiliki background belajar otodidak sampai diminta mengajari teman,dan menjadi personal trainer private namun	Penyebaran informasi jasa masih sebatas dari whatsapp dan facebook saja

			olahraga yang masih muda dan fresh serta tidak adanya keterbatasan gender dalam olahraga tersebut	dalam hal ini kita tetap bersaing masalah pengalaman, harga dan sertifikasi yang bisa menarik pelanggan	
4	Vety Tunjungsari	Yoga	Sekarang instruktur olahraga bisa dari berbagai jenis usia dan gender, juga banyaknya tutorial dari instruktur olahraga melalui social media	Banyak orang yang sudah menjadi instruktur olahraga dengan belajar dari les sebelumnya maupun mengikuti pelatihan khusus sehingga dalam persaingan ini kita akan bersaing tentang harga dan juga pelayanan jasa	Penyebaran masih informasi jasa masih melalui social media dan dari omongan mulut ke mulut
5	Siti Aminah	Yoga	Sekarang banyak sekali macam olahraga yang dibuka dikota Malang, mereka rata-rata menyediakan tempat dan instruktur olahraga yang profesional	Banyak sekali orang yang memiliki keahlian dibidang berbagai olahraga sehingga dalam hal ini persaingan instruktur olahraga harus berdasar pada keahlian apa saja yang bisa mereka berikan terhadap pengguna jasa, pengalaman melatih dan juga	Penyebaran informasi jasa masih melalui social media dan dari omongan mulut ke mulut sehingga jika orang yang jauh dari daerah biasa kita melatih akan tidak mengetahuinya

6	Septyane Tuegi Purnamasari	Zumba	Sekarang ini banyak sekali orang yang bersekolah khusus demi menjadi instruktur olahraga terlebih dimalang,banyak pengguna yang membutuhkan jasanya sehingga terdapat banyak peluang	Banyak sekali orang yang mencoba menjadi instruktur olahraga dari otodidak namun dalam hal ini persaingan kita hadapi dengan keahlian,pengalaman,sertifikasi serta harga	Penyebaran informasi jasa masih melalui dunia online yang sebatas social media saja,namun ada juga yang mendapat informasinya dari mulut ke mulut
7	Neni Agus Hartawati	Aerobik	Sekarang banyak sekali macam olahraga yang dibuka dikota Malang,mereka rata-rata menyediakan tempat dan instruktur olahraga yang profesional	Banyak sekali orang yang mencoba menjadi instruktur olahraga dari otodidak namun dalam hal ini persaingan kita hadapi dengan keahlian,pengalaman,sertifikasi serta harga	Penyebaran informasi jasa masih melalui social media dan dari omongan mulut ke mulut sehingga jika orang yang jauh dari daerah biasa kita melatih akan tidak mengetahuinya

8	Titik Indah	Aerobik	Sekarang ini banyak sekali orang yang bersekolah khusus demi menjadi instruktur olahraga terlebih dimalang,banyak pengguna yang membutuhkan jasanya sehingga terdapat banyak peluang	Banyak sekali orang yang mencoba menjadi instruktur olahraga dari otodidak namun dalam hal ini persaingan kita hadapi dengan keahlian,pengalaman,sertifikasi serta harga	Penyebaran informasi jasa masih melalui dunia online yang sebatas social media saja,namun ada juga yang mendapat informasinya dari mulut ke mulut
9	Himmatul Yani	Aerobik	Sekarang instruktur olahraga bisa dari berbagai jenis usia dan gender,juga banyaknya tutorial dari instruktur olahraga melalui social media	Banyak orang yang sudah menjadi instruktur olahraga dengan belajar dari les sebelumnya maupun mengikuti pelatihan khusus sehingga dalam persaingan ini kita akan bersaing tentang harga dan juga pelayanan jasa	Penyebaran masih informasi jasa masih melalui social media dan dari omongan mulut ke mulut

10	Wenny	Aerobik	<p>Sekarang ini banyak sekali orang yang bersekolah khusus demi menjadi instruktur olahraga terlebih dimalang,banyak pengguna yang membutuhkan jasanya sehingga terdapat banyak peluang</p>	<p>Banyak sekali orang yang mencoba menjadi instruktur olahraga dari otodidak namun dalam hal ini persaingan kita hadapi dengan keahlian,pengalaman,sertifikasi serta harga</p>	<p>Penyebaran informasi jasa masih melalui dunia online yang sebatas social media saja,namun ada juga yang mendapat informasinya dari mulut ke mulut</p>
11	Suadi	Renang	<p>Perkembangannya sangat pesat apalagi sekarang terdapat berbagai macam olahraga untuk tujuan sendiri-sendiri,instruktur olahraga yang masih muda dan fresh serta tidak adanya keterbatasan gender dalam olahraga tersebut</p>	<p>Banyak sekali instruktur olahraga yang mulainya memiliki background belajar otodidak sampai diminta mengajari teman,dan menjadi personal trainer private namun dalam hal ini kita tetap bersaing masalah pengalaman,harga dan sertifikasi yang bisa menarik pelanggan</p>	<p>Penyebaran informasi jasa masih sebatas dari whatsapp dan facebook saja,dari mulut ke mulut seperti dari orang tua dari murid yang dilatih</p>

12	Christ	Renang	Sangat pesat,sekarang ini banyak instruktur yang tidak harus berasal dari background pendidikan olahraga dapat menjadi instruktur olahraga,pendidikannya pun tidak perlu tinggi	Sudah banyak instruktur yang muda dan bertalenta,juga sudah banyak instruktur yang melalui media social untuk memberi tentor latihan	Penyebaran informasi jasa masih sebatas dari instagram,dan media social lainnya juga masih dari mulut ke mulut seperti murid yang dilatih punya teman lalu mengajak temannya
13	Novian Rohmat	Renang	Sangat pesat karena sekarang semua bisa menjadi instruktur kalau mau belajar,tidak harus sekolah khusus,bisa belajar otodidak maupun mempelajari dari social media	Banyak sekali instruktur yang berpotensi sekalipun mereka baru merintis pengalaman sebentar dan banyak juga instruktur yang tidak ada batasan gender sehingga kita akan bersaing dari pengalaman dan sertifikat yang sudah didapat	Penyebaran informasi jasa masih sebatas dari instagram dan facebook saja,serta whatsapp juga dari mulut ke mulut antar teman murid maupun orang tua murid

14	Audi	Muay Thai	Sekarang instruktur olahraga bisa dari berbagai jenis usia dan gender,juga banyaknya tutorial dari instruktur olahraga melalui social media	Banyak orang yang sudah menjadi instruktur olahraga dengan belajar dari les sebelumnya maupun mengikuti pelatihan khusus sehingga dalam persaingan ini kita akan bersaing tentang harga dan juga pelayanan jasa	Penyebaran masih informasi jasa masih melalui social media dan dari omongan mulut ke mulut,dari para UKM bela diri juga
----	------	-----------	---	---	---



### Lampiran 3 Hasil Kuisisioner Online

Observasi dilakukan dengan melakukan penyebaran kuisisioner online yang berguna untuk menggali kebutuhan pengguna terkait system dan untuk memperdalam permasalahan yang harus diselesaikan dengan system. Kuisisioner ini ada yang harus melengkapi jawabannya sendiri dan ada yang sudah dilengkapi dengan pilihan jawaban yang harus dipilih.

#### B.1 Pertanyaan Terkait Kuisisioner Online

Materi kuisisioner ini bertujuan untuk memperoleh data pengguna yang memiliki permasalahan yang akan diselesaikan oleh system, menggali kebutuhan pengguna dan persetujuan pengguna dengan adanya system

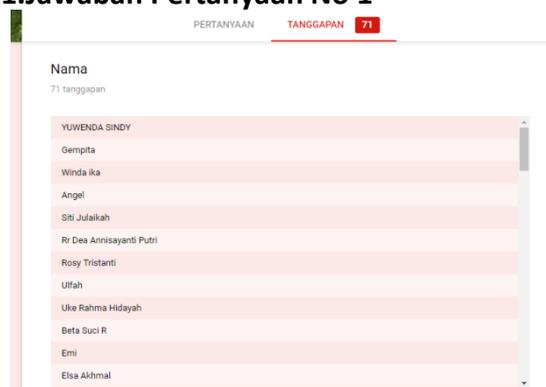
No.	Pertanyaan	Jawaban
1.	Siapakah nama anda?	
2.	Apa jenis kelamin anda?	-Laki-laki -Perempuan
3.	Berapakah umur anda?	-15-60 ->60
4.	Apa pekerjaan anda?	-Pelajar -Umum -Mahasiswa
5.	Apakah anda menyukai olahraga?	-Ya -Tidak
6.	Olahraga apa yang anda sukai?	-Aerobik -Zumba -Gym -Yoga -Renang -Muay Thai
7.	Apakah anda membutuhkan instruktur dalam melakukan kegiatan olahraga tersebut?	-Ya -Tidak

8.	Apakah anda pernah mengalami kesulitan dalam mencari instruktur olahraga tersebut di Kota Malang?	-Ya -Tidak
9.	Menurut anda, bagaimana peran instruktur olahraga dalam kegiatan olahraga di Kota Malang?	-Sangat Penting -Cukup Penting -Penting -Tidak Penting
10.	Menurut anda, kendala apa saja yang sering ditemui dalam mencari instruktur olahraga di Kota Malang	-sulitnya mengetahui profil instruktur -sulit mencari instruktur yang dapat menyesuaikan dengan jadwal kita -kurangnya refrensi -kurangnya informasi paket harga instruktur per sesi
11.	Apa saja kriteria anda dalam mencari instruktur olahraga di Kota Malang?	-harga -jadwal latihan -paket kelas -profil pelatih -jenis kelamin
12.	Jika dibuat sistem untuk menangani permasalahan tersebut, apakah anda tertarik menggunakannya	-Ya -Tidak

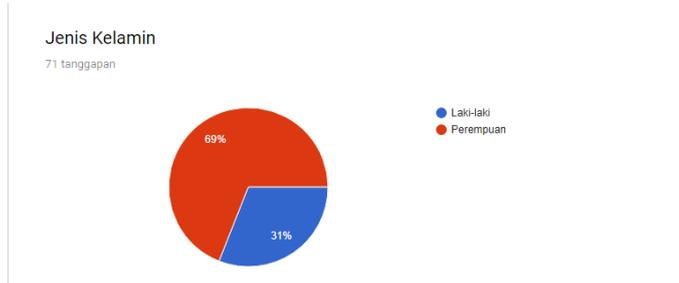
### B.2 Jawaban Terkait Kuisiner Online

Materi kuisiner ini menggunakan 71 responden sebagai bahan sampel untuk memperoleh data pengguna yang memiliki permasalahan yang akan diselesaikan oleh system, menggali kebutuhan pengguna dan persetujuan pengguna dengan adanya system.

#### 1. Jawaban Pertanyaan No 1

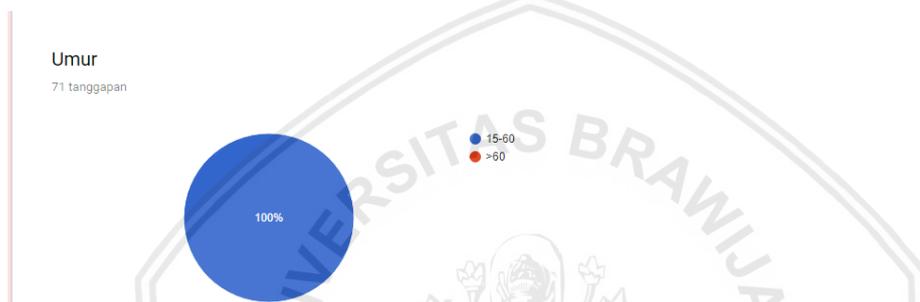


## 2. Jawaban Pertanyaan No 2



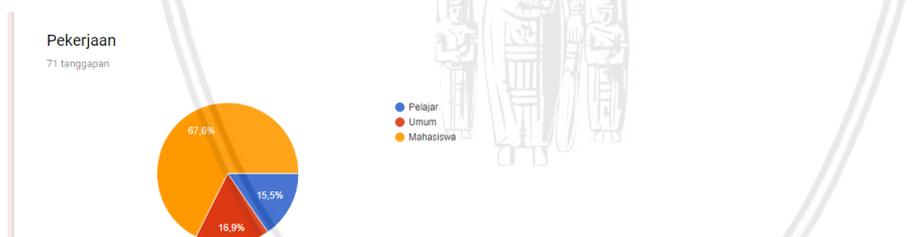
Dari jawaban pertanyaan kuisiner nomer 2 diperoleh data 69% terdiri dari perempuan dan 31% laki-laki dari 71 sampel responden yang mengisi data

## 3. Jawaban Pertanyaan No 3



Dari jawaban pertanyaan kuisiner nomer 3 diperoleh data 100% berumur 15-60 dari 71 sampel responden yang mengisi data

## 4. Jawaban Pertanyaan No 4



Dari jawaban pertanyaan kuisiner nomer 4 diperoleh data 67,6 memiliki pekerjaan mahasiswa, 16,9 umum dan 15,5% pelajar dari 71 sampel responden yang mengisi data

## 5. Jawaban Pertanyaan No 5

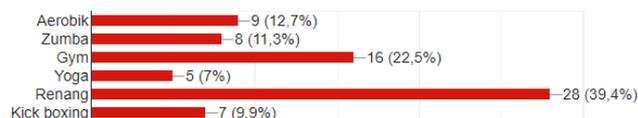


Dari jawaban pertanyaan kuisiner nomer 5 diperoleh data 83,1 % menyukai olahraga dan 16,9% tidak menyukai olahraga dari 71 sampel responden yang mengisi data.

### 6. Jawaban Pertanyaan No 6

Olahraga apa yang anda sukai?

71 tanggapan

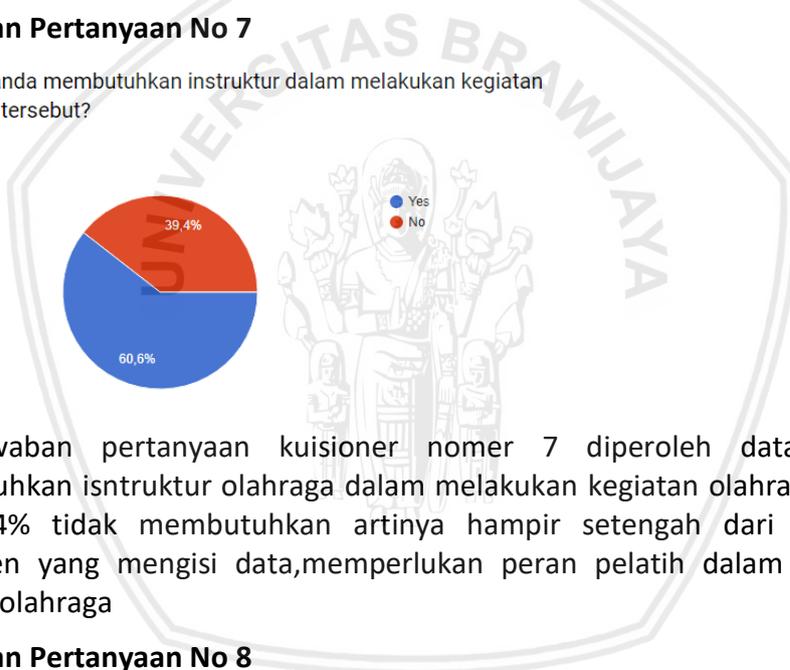


Dari jawaban pertanyaan kuisiner nomer 6 diperoleh data 12,7 % menyukai olahraga aerobik,11,3% menyukai zumba,22,5% menyukai gym,7% menyukai yoga,39,4% menyukai renang,9,9% menyukai muay thai dan sisanya lain-lain dari 71 sampel responden yang mengisi data.

### 7. Jawaban Pertanyaan No 7

Apakah anda membutuhkan instruktur dalam melakukan kegiatan olahraga tersebut?

71 tanggapan

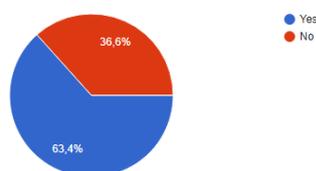


Dari jawaban pertanyaan kuisiner nomer 7 diperoleh data 60,6, % membutuhkan instruktur olahraga dalam melakukan kegiatan olahraga tersebut dan 39,4% tidak membutuhkan artinya hampir setengah dari 71 sampel responden yang mengisi data,memerlukan peran pelatih dalam melakukan kegiatan olahraga

### 8. Jawaban Pertanyaan No 8

Apakah anda pernah mengalami kesulitan dalam mencari instruktur olahraga tersebut di kota Malang?

71 tanggapan



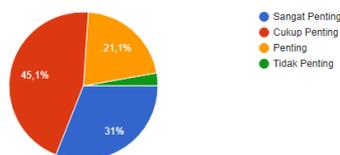
Dari jawaban pertanyaan kuisiner nomer 8 diperoleh data 63,4 % mengalami kesulitan mencari instruktur olahraga dalam melakukan kegiatan olahraga tersebut dan 36,6% tidak mengalami kesulitan artinya hampir setengah dari 71 sampel responden yang mengisi data,memerlukan penyelesaian masalah dari kesulitan tersebut.



## 9. Jawaban Pertanyaan No 9

Menurut anda, bagaimana peran instruktur dalam kegiatan olahraga di kota Malang?

71 tanggapan

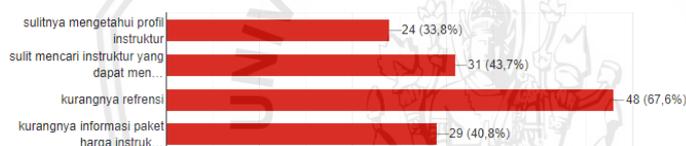


Dari jawaban pertanyaan kuisioner nomer 9 diperoleh data 45,1 % menganggap cukup pentingnya peran instruktur olahraga, 21,1% menganggap penting peran instruktur olahraga dan 31% menganggap sangat penting peran instruktur olahraga sehingga disimpulkan dari 71 sampel responden yang mengisi data, setuju akan peran penting instruktur olahraga dalam kegiatan olahraganya.

## 10. Jawaban Pertanyaan No 10

Menurut anda, kendala apa saja yang sering anda temui dalam mencari instruktur olahraga di Malang?

71 tanggapan

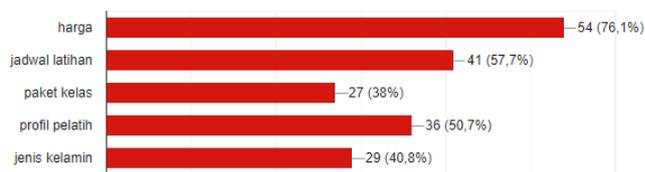


Dari jawaban pertanyaan kuisioner nomer 10 diperoleh data 33,8% menjawab kendala nya adalah sulitnya mengetahui profil instruktur, 43,7% menjawab sulit mencari instruktur yang dapat menyesuaikan jadwal kita, 67,6% menjawab kurangnya referensi, 40,8% menjawab kurangnya informasi paket harga untuk latihan tiap sesi sehingga disimpulkan dari 71 sampel responden yang mengisi data, sangat sering menemui beberapa kesulitan seperti diatas.

## 11. Jawaban Pertanyaan No 11

Apa saja kriteria anda dalam mencari instruktur olahraga di Malang?

71 tanggapan

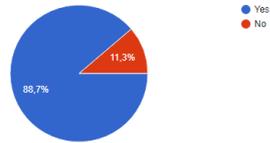


Dari jawaban pertanyaan kuisioner nomer 11 diperoleh data 76,1% menjawab kriteria dalam mencari instruktur dilihat dari harga, 57,7% menjawab dilihat dari jadwal latihan, 38% menjawab dilihat dari paket kelas, 50,7% menjawab dilihat dari profil pelatih, 40,8% menjawab dilihat dari jenis kelamin sehingga disimpulkan dari 71 sampel responden yang mengisi data, banyak kriteria yang dibutuhkan pengguna dalam mencari instruktur olahraga seperti diatas.

## 12. Jawaban Pertanyaan No 12

Jika dibuat sistem pencarian instruktur olahraga di Malang, apakah anda tertarik menggunakannya?

71 tanggapan



Dari jawaban pertanyaan kuisisioner nomer 12 diperoleh data 88,7 % setuju dengan dibuatnya sebuah sistem guna mengatasi permasalahan mereka dan 11,3% tidak setuju dengan adanya sistem artinya hampir setengah dari 71 sampel responden yang mengisi data, menyetujui dibuatnya sistem untuk penyelesaian masalah dari kesulitan tersebut.

