

PEMBANGUNAN APLIKASI *DASHBOARD* UNTUK ANALISA EFEKTIVITAS *MARKETING ONLINE* BERBASIS *WEBSITE*

(Studi Kasus: Digital Marketing Malang)

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Ubaidillah Hakim Fadly

NIM: 155150201111290



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

PENGESAHAN

PEMBANGUNAN APLIKASI *DASHBOARD* UNTUK ANALISA EFEKTIVITAS
MARKETING ONLINE BERBASIS *WEBSITE*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Ubaidillah Hakim Fadly
NIM: 155150201111290

Skripsi ini telah diuji dan dinyatakan lulus pada
19 Juni 2019

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I



Fajar Pradana, S.ST, M.Eng
NIP. 19871121 201504 1 004

Dosen Pembimbing II



Nurudin Santoso, S.T., M.T
NIP. 197409162000121001

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 referensi.

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, 16 Mei 2019



Ubaidillah Hakim Fadly

NIM: 155150201111290

PRAKATA

Assalamu'alaikum Wr. Wb. AlhadulillahiRobbil'Alamin. Segala puji bagi Allah SWT yang telah memberikan kita kesehatan dan kesempatan sehingga penulis bisa menyelesaikan skripsi yang berjudul "Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*" dengan baik.

Pada kesempatan ini, penulis menyampaikan ucapan terima kasih kepada pihak-pihak yang telah membantu penulis selama proses pengerjaan skripsi, diantaranya adalah:

1. Allah SWT yang telah memberikan kemudahan selama proses penulisan skripsi ini.
2. Kepada kedua orang tua saya yang selalu mendukung, menyemangati serta mendoakan saya selama menulis skripsi ini.
3. Bapak Fajar Pradana, S.ST, M.Eng selaku dosen pembimbing I, yang telah meluangkan dan menyempatkan waktunya untuk memberikan ilmu, saran serta masukan dalam proses penyusunan skripsi ini.
4. Bapak Nurudin Santoso, S.T., M.T selaku dosen pembimbing II, yang juga telah meluangkan waktunya untuk memberikan ilmu, saran, serta masukan dalam proses penyusunan skripsi ini.
5. Bapak S. Nowo Aprilianto, S.Psi Selaku Owner Digital Marketing Malang dan semua anggota dari Digital Marketing Malang yang telah mendukung serta memberikan wawasan dan ilmu baru yang sangat berharga dalam proses penyusunan skripsi ini.
6. Keluarga Anonyminus yang telah memberikan hiburan, cerita, dan semangat selama proses penyusunan skripsi ini.
7. Keluarga Kontrakan Al-Khawarizmi II yang telah memberikan motivasi dan semangat dalam proses penyusunan skripsi ini.
8. Tim *Startup* Buatin, Villa, Wahyu, Nugroho dan Rayyan yang telah memberikan semangat serta dukungan selama penyusunan skripsi ini.
9. Seluruh teman-teman Fakultas Ilmu Komputer yang telah memberikan segala bantuan serta dukungan selama proses penyusunan skripsi ini.
10. Kepada seluruh pihak yang tidak dapat disebutkan satu per satu yang telah membantu selama proses penyusunan skripsi ini.

Semoga setiap bantuannya mendapatkan balasan dari Allah SWT. Penulis menyadari bahwa dalam penyusunan skripsi ini masih ditemukan banyak kekurangan seperti peribahasa tiada gading yang tak retak. Terakhir, semoga skripsi ini bisa bermanfaat baik untuk pembaca maupun untuk pengembangan penelitian selanjutnya. Oleh karena itu, segala bentuk kritik dan saran yang bersifat membangun sangat penulis harapkan dan bisa disampaikan melalui email penulis.

Malang, 16 Mei 2019

Penulis
ubaidillahhf@gmail.com



ABSTRAK

Ubaidillah Hakim Fadly, Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*

Pembimbing: Fajar Pradana dan Nurudin Santoso

Digital Marketing Malang merupakan sebuah usaha yang bergerak dibidang *marketing online* atau *internet marketing* yang menyediakan jasa berupa pemasangan iklan beserta optimisasinya di *social media* maupun *search engine*. Dalam pelaksanaan usahanya terdapat sebuah permasalahan utama yang dialami dan dirasakan oleh pihak Digital Marketing Malang. Permasalahannya adalah ketika menjalankan pemasangan iklan pada *Social Media Advertising* (SMA) seperti Facebook dengan menggunakan teknik *splitting*, DMM akan merasa kesulitan dan kerepotan ketika menghentikan banyak iklan dalam satu waktu, sedangkan ketika banyak iklan yang berjalan akan membuat biaya anggaran iklan juga semakin banyak atau dalam istilah Digital Marketing Malang disebut *boncos*. Perlu diketahui untuk dapat menghasilkan racikan iklan yang pas dan menguntungkan diperlukan banyak iklan untuk mengiklankan 1 produk yang sama atau dalam istilah digital marketing disebut dengan *splitting*. Sehingga ketika menjalankan banyak iklan dan kemudian mendapatkan wawasan dari keseluruhan iklan tersebut, maka akan dilakukan analisis untuk mengetahui iklan dengan racikan yang pas atau efektif dan iklan yang kurang efektif. Untuk iklan yang pas akan dilakukan pemfokusan iklan agar lebih efektif sedangkan iklan yang kurang efektif akan dihentikan agar tidak menghabiskan anggaran iklan dengan sia-sia. Untuk itulah dilakukan pembangunan aplikasi *dashboard* untuk Analisa efektivitas *marketing online* yang bisa melakukan pengecekan otomatis setiap 1 jam sekali terhadap iklan yang berjalan dalam satu set iklan dan akan mengambil iklan teratas yang diinginkan dan menghentikan iklan yang kurang efektif. Proses pembangunan aplikasi perangkat lunak pada penelitian ini menggunakan metode pendekatan *waterfall*. Tahapan yang dilakukan dalam penelitian ini adalah: studi literatur, analisis kebutuhan, perancangan, implementasi, pengujian, pengambilan kesimpulan dan saran. Aplikasi ini dikembangkan menggunakan basis website dengan bahasa pemrograman *PHP*. Pengujian yang dilakukan berupa pengujian unit yang menghasilkan nilai 100% valid dari 3 kasus uji, pengujian integrasi yang menghasilkan nilai 100% valid dari 1 kasus uji, dan pengujian validasi yang menghasilkan 100% valid dari 52 kasus uji.

Kata kunci : Digital Marketing Malang, *PHP*, *cURL*, *Cron Job*, *Facebook Marketing API*.

ABSTRACT

Ubaidillah Hakim Fadly, Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*

Supervisors: Fajar Pradana and Nurudin Santoso

Digital Marketing Malang is a service business focus in marketing online or internet marketing that provide social media advertising and search engine optimization service. In the implementation Digital Marketing Malang found a main problem, when running ads in Social Media Advertising (SMA) like a facebook ads with splitting strategy, they will be difficulties and hassles when trying to stop many ads running in the same time, even though the budget ads will absolutely spend a lot that follow the many of ads. For you information, when try to running ads perfectly we must provide many of ads for 1 goods. So, when running a lot of ads and we got an insight wes must keep the perfect one or whatever do you need and stop other ads early, because the other ads will spend not under control which can spend budget quickly. Based on the main problem, they need a dashboard application development to analyze ads effectiveness that can checking the ads once a hour and pick the most effectiveness and shutdown the ads others, so they can save the budget of ads more efficient. The application development process for this research is waterfall model approach. The steps taken in this research are: literatur study, requirement analysis, design, implementation, testing, taking conclusion and suggestions. This application developing with PHP language. The testing include unit testing get 100% valid from 3 test case, Integration testing get 100% valid from 1 test case, and validation testing get 100% valid from 52 test case.

Keywords: Malang Digital Marketing, *PHP*, *cURL*, *Cron Job*, *Facebook Marketing API*.

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
PRAKATA.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL.....	xv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	6
2.1 Kajian Pustaka	6
2.2 <i>Digital Marketing Malang (DMM)</i>	7
2.3 Rekayasa Perangkat Lunak	8
2.4 Pengembangan Perangkat Lunak	9
2.4.1 <i>Software Development Life Cycle (SDLC)</i>	9
2.4.2 Pendekatan Berorientasi Objek	11
2.5 <i>Unified Modelling Language (UML)</i>	11
2.5.1 Diagram <i>Use Case</i>	11
2.5.2 Diagram <i>Sequence</i>	12
2.5.3 Diagram <i>Class</i>	13
2.6 <i>Model View Controller (MVC)</i>	14
2.7 Codeigniter.....	14
2.8 PHP	15
2.9 CURL.....	15
2.10 Cron Job	15

2.11 Facebook API.....	16
2.12 <i>Hyper Text Markup Language</i> (HTML).....	16
2.13 Javascript	17
2.14 jQuery	17
2.15 AJAX	17
2.16 MySQL.....	18
2.17 <i>Dashboard</i>	18
2.18 Marketing Online	19
2.19 <i>Click Through Rate</i> (CTR).....	19
2.20 <i>Return on Ads Spending</i> (ROAS).....	19
2.21 <i>Return on Marketing Investment</i> (ROMI)	20
2.22 Pengujian Perangkat Lunak.....	20
BAB 3 METODOLOGI PENELITIAN	22
3.1 Studi Literatur	22
3.2 Analisis Kebutuhan	23
3.3 Perancangan Sistem.....	23
3.4 Implementasi	23
3.5 Pengujian	24
3.6 Kesimpulan dan Saran	24
BAB 4 ANALISIS KEBUTUHAN	25
4.1 Deskripsi Umum Sistem	25
4.2 Proses Rekayasa Kebutuhan	25
4.2.1 Elisitasi Kebutuhan dan Hasil Analisis	25
4.3 Identifikasi Aktor.....	30
4.4 Kebutuhan Fungsional	30
4.5 Kebutuhan Non-Fungsional	37
4.6 Diagram <i>Use Case</i>	37
4.7 Skenario <i>Use Case</i>	38
BAB 5 PERANCANGAN DAN IMPLEMENTASI	66
5.1 Perancangan	66
5.1.1 Perancangan Arsitektur.....	66
5.1.2 Perancangan Basis Data	71



5.1.3 Perancangan Komponen	73
5.1.4 Perancangan Antarmuka.....	76
5.2 Implementasi	78
5.2.1 Spesifikasi Sistem	78
5.2.2 Implementasi Basis Data	80
5.2.3 Implementasi Kode Program	80
5.2.4 Implementasi Antarmuka	84
BAB 6 PENGUJIAN	89
6.1 Pengujian Unit.....	89
6.1.1 Pengujian Unit Klas Setiklan pada Fungsi tambahSetIklan().....	89
6.1.2 Pengujian Unit Klas Dashboard pada Fungsi getIklanGrafik().....	94
6.1.3 Pengujian Unit Klas Dashboard pada Fungsi tambahKampanye()	95
6.2 Pengujian Integrasi	97
6.2.1 Pengujian Integrasi Method ctrCheck() Klas Dashboard	97
6.3 Pengujian Validasi	103
6.3.1 Pengujian Validasi Login.....	104
6.3.2 Pengujian Validasi Lupa Password	105
6.3.3 Pengujian Validasi Pendaftaran	107
6.3.4 Pengujian Validasi Melihat <i>Preface</i>	108
6.3.5 Pengujian Validasi Melihat 3 Komponen Iklan Facebook	108
6.3.6 Pengujian Validasi Melihat ROAS dan ROMI.....	109
6.3.7 Pengujian Validasi Melihat <i>About</i> SIDAMAR.....	109
6.3.8 Pengujian Validasi Melihat Dokumentasi	110
6.3.9 Pengujian Validasi Login Facebook	110
6.3.10 Pengujian Validasi Menambah Kampanye.....	111
6.3.11 Pengujian Validasi Melihat Kampanye.....	113
6.3.12 Pengujian Validasi Mengedit Kampanye.....	113
6.3.13 Pengujian Validasi Menghapus Kampanye	115
6.3.14 Pengujian Validasi Memulai Kampanye.....	116
6.3.15 Pengujian Validasi Menjeda Kampanye.....	116
6.3.16 Pengujian Validasi Menambah Set Iklan	116
6.3.17 Pengujian Validasi Melihat Set Iklan	118

6.3.18 Pengujian Validasi Mengedit Set Iklan.....	118
6.3.19 Pengujian Validasi Melihat Detail Set Iklan.....	119
6.3.20 Pengujian Validasi Menghapus Set Iklan	119
6.3.21 Pengujian Validasi Memulai Set Iklan	120
6.3.22 Pengujian Validasi Menjeda Set Iklan	121
6.3.23 Pengujian Validasi Menambah Iklan	121
6.3.24 Pengujian Validasi Melihat Iklan	122
6.3.25 Pengujian Validasi Memulai Iklan	123
6.3.26 Pengujian Validasi Menjeda Iklan	123
6.3.27 Pengujian Validasi Mengedit Iklan	123
6.3.28 Pengujian Validasi Menghapus Iklan.	125
6.3.29 Pengujian Validasi Menambah Ad Creative	125
6.3.30 Pengujian Validasi Melihat Ad Creative	126
6.3.31 Pengujian Validasi Memotong Gambar	127
6.3.32 Pengujian Validasi Melihat Preview Ad Creative	128
6.3.33 Pengujian Validasi Mengedit Ad Creative	128
6.3.34 Pengujian Validasi Menghapus Ad Creative	130
6.3.35 Pengujian Validasi Menambah Ad Pixel.....	130
6.3.36 Pengujian Validasi Melihat Ad Pixel.....	132
6.3.37 Pengujian Validasi Mengedit Ad Pixel.....	133
6.3.38 Pengujian Validasi Set Halaman Website	134
6.3.39 Pengujian Validasi Mengedit Halaman Website.....	135
6.3.40 Pengujian Validasi Menggenerate Ad Pixel	137
6.3.41 Pengujian Validasi Copy to Clipboard	137
6.3.42 Pengujian Validasi Mengganti Akun Iklan	137
6.3.43 Pengujian Validasi Melihat Sisa Anggaran	138
6.3.44 Pengujian Validasi Melihat Iklan Pada Dashboard.....	139
6.3.45 Pengujian Validasi Melihat Grafik Wawasan	139
6.3.46 Pengujian Validasi Melihat Status Terakhir	140
6.3.47 Pengujian Validasi Melihat Set Iklan Pada Dashboard	141
6.3.48 Pengujian Validasi Compare Set Iklan	141
6.3.49 Pengujian Validasi Hapus Compare Set Iklan.....	142

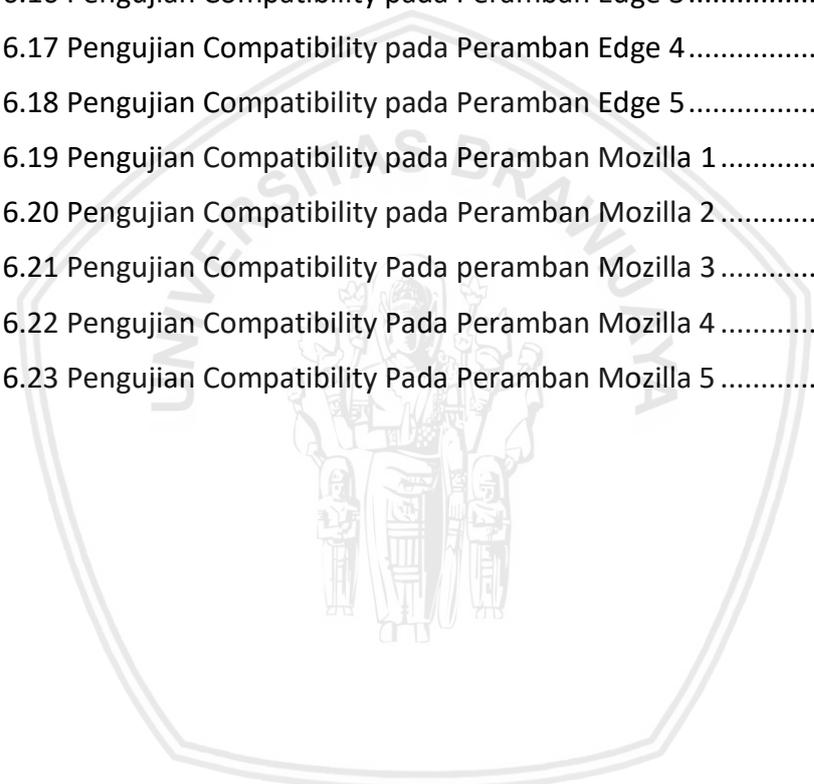
6.3.50 Pengujian Validasi Melihat Log	143
6.3.51 Pengujian Validasi Melihat Notifikasi.....	144
6.3.52 Pengujian Validasi <i>Logout</i>	144
6.4 Pengujian <i>Compatibility</i>	144
BAB 7 PENUTUP	155
7.1 Kesimpulan.....	155
7.2 Saran	156
DAFTAR REFERENSI	157



DAFTAR GAMBAR

Gambar 2.1 <i>Waterfall Model</i>	11
Gambar 2.2 Contoh Diagram <i>Use Case</i>	12
Gambar 2.3 Contoh <i>Class Diagram</i>	14
Gambar 3.1 Diagram Alir Penelitian.....	22
Gambar 4.1 Pembuatan Iklan dan Info Wawasan Sebelumnya	27
Gambar 4.2 Pembuatan Pixel dan Info Wawasan Website sebelumnya	28
Gambar 4.3 Aplikasi Dashboard Monitoring Efektifitas Iklan	29
Gambar 4.4 Use Case Diagram.....	38
Gambar 5.1 Diagram Perancangan Sistem Aplikasi	66
Gambar 5.2 <i>Sequence Diagram</i> Menambah Set Iklan	67
Gambar 5.3 <i>Sequence Diagram</i> Melihat Grafik	68
Gambar 5.4 <i>Sequence Diagram</i> Menambah Kampanye.....	69
Gambar 5.5 <i>Class Diagram</i>	70
Gambar 5.6 <i>Entity Relationship Diagram</i>	72
Gambar 5.7 <i>Physical Data Model</i>	73
Gambar 5.8 Perancangan Antarmuka Dashboard	76
Gambar 5.9 Perancangan Antarmuka Kampanye.....	77
Gambar 5.10 Perancangan Antarmuka Iklan	77
Gambar 5.11 Implementasi Antarmuka Melihat Dashboard	85
Gambar 5.12 Implementasi Antarmuka Menambah Kampanye.....	85
Gambar 5.13 Implementasi Antarmuka Menambah Set Iklan	86
Gambar 5.14 Implementasi Antarmuka Melihat Ad Creative	87
Gambar 5.15 Implementasi Antarmuka Melihat Ad Pixel	87
Gambar 5.16 Implementasi Antarmuka Menambah Iklan	88
Gambar 6.1 <i>Flow Graph</i> tambahSetIklan()	90
Gambar 6.2 <i>Flow Graph</i> getIklanGrafik()	94
Gambar 6.3 <i>Flow Graph</i> tambahKampanye()	96
Gambar 6.4 <i>Flow Graph</i> ctrCheck().....	99
Gambar 6.5 Pengujian Compatibility pada Peramban Opera 1.....	145
Gambar 6.6 Pengujian Compatibility pada Peramban Opera 2.....	145
Gambar 6.7 Pengujian Compatibility pada Peramban Opera 3.....	146

Gambar 6.8 Pengujian Compatibility pada Peramban Opera 4.....	146
Gambar 6.9 Pengujian Compatibility pada Peramban Chrome 1.....	147
Gambar 6.10 Pengujian Compatibility pada Peramban Chrome 2.....	147
Gambar 6.11 Pengujian Compatibility pada Peramban Chrome 3.....	148
Gambar 6.12 Pengujian Compatibility pada Peramban Chrome 4.....	148
Gambar 6.13 Pengujian Compatibility pada Peramban Chrome 5.....	149
Gambar 6.14 Pengujian Compatibility pada Peramban Edge 1.....	149
Gambar 6.15 Pengujian Compatibility pada Peramban Edge 2.....	150
Gambar 6.16 Pengujian Compatibility pada Peramban Edge 3.....	150
Gambar 6.17 Pengujian Compatibility pada Peramban Edge 4.....	151
Gambar 6.18 Pengujian Compatibility pada Peramban Edge 5.....	151
Gambar 6.19 Pengujian Compatibility pada Peramban Mozilla 1.....	152
Gambar 6.20 Pengujian Compatibility pada Peramban Mozilla 2.....	152
Gambar 6.21 Pengujian Compatibility Pada peramban Mozilla 3.....	153
Gambar 6.22 Pengujian Compatibility Pada Peramban Mozilla 4.....	153
Gambar 6.23 Pengujian Compatibility Pada Peramban Mozilla 5.....	154



DAFTAR TABEL

Tabel 2.1 Komponen <i>Sequence</i> Diagram	13
Tabel 4.1 Kebutuhan Hasil Wawancara	26
Tabel 4.2 Identifikasi Aktor Sistem.....	30
Tabel 4.3 Kebutuhan Fungsional.....	30
Tabel 4.4 Kebutuhan Non Fungsional	37
Tabel 4.5 Skenario <i>Use Case Login</i>	39
Tabel 4.6 Skenario <i>Use Case Lupa Password</i>	39
Tabel 4.7 Skenario <i>Use Case Pendaftaran</i>	40
Tabel 4.8 Skenario <i>Use Case Melihat Preface</i>	40
Tabel 4.9 Skenario <i>Use Case Melihat 3 Komponen Iklan Facebook</i>	41
Tabel 4.10 Skenario <i>Use Case Melihat ROAS dan ROMI</i>	41
Tabel 4.11 Skenario <i>Use Case Melihat About SIDAMAR</i>	42
Tabel 4.12 Skenario <i>Use Case Melihat Dokumentasi</i>	42
Tabel 4.13 Skenario <i>Use Case Login Facebook</i>	42
Tabel 4.14 Skenario <i>Use Case Menambah Kampanye</i>	43
Tabel 4.15 Skenario <i>Use Case Melihat Kampanye</i>	44
Tabel 4.16 Skenario <i>Use Case Mengedit Kampanye</i>	44
Tabel 4.17 Skenario <i>Use Case Menghapus Kampanye</i>	45
Tabel 4.18 Skenario <i>Use Case Memulai Kampanye</i>	45
Tabel 4.19 Skenario <i>Use Case Menjeda Kampanye</i>	46
Tabel 4.20 Skenario <i>Use Case Menambah Set Iklan</i>	46
Tabel 4.21 Skenario <i>Use Case Melihat Set Iklan</i>	47
Tabel 4.22 Skenario <i>Use Case Mengedit Set Iklan</i>	47
Tabel 4.23 Skenario <i>Use Case Melihat Detail Set Iklan</i>	48
Tabel 4.24 Skenario <i>Use Case Menghapus Set Iklan</i>	48
Tabel 4.25 Skenario <i>Use Case Memulai Set Iklan</i>	49
Tabel 4.26 Skenario <i>Use Case Menjeda Set Iklan</i>	49
Tabel 4.27 Skenario <i>Use Case Menambah Iklan</i>	50
Tabel 4.28 Skenario <i>Use Case Melihat Iklan</i>	50
Tabel 4.29 Skenario <i>Use Case Memulai Iklan</i>	51
Tabel 4.30 Skenario <i>Use Case Menjeda Iklan</i>	51



Tabel 4.31 Skenario <i>Use Case</i> Mengedit Iklan	52
Tabel 4.32 Skenario <i>Use Case</i> Menghapus Iklan.....	52
Tabel 4.33 Skenario <i>Use Case</i> Menambah Ad Creative	53
Tabel 4.34 Skenario <i>Use Case</i> Melihat Ad Creative	53
Tabel 4.35 Skenario <i>Use Case</i> Memotong Gambar	54
Tabel 4.36 Skenario <i>Use Case</i> Melihat <i>Preview</i> Ad Creative	54
Tabel 4.37 Skenario <i>Use Case</i> Mengedit Ad Creative	55
Tabel 4.38 Skenario <i>Use Case</i> Menghapus Ad Creative.....	55
Tabel 4.39 Skenario <i>Use Case</i> Menambah Ad Pixel	56
Tabel 4.40 Skenario <i>Use Case</i> Melihat Ad Pixel	57
Tabel 4.41 Skenario <i>Use Case</i> Mengedit Ad Pixel	57
Tabel 4.42 Skenario <i>Use Case</i> Set Halaman Website.....	58
Tabel 4.43 Skenario <i>Use Case</i> Mengedit Halaman Website	58
Tabel 4.44 Skenario <i>Use Case</i> <i>Menggenerate</i> Ad Pixel.....	59
Tabel 4.45 Skenario <i>Use Case</i> <i>Copy to Clipboard</i>	59
Tabel 4.46 Skenario <i>Use Case</i> Mengganti Akun Iklan	60
Tabel 4.47 Skenario <i>Use Case</i> Melihat Sisa Anggaran	60
Tabel 4.48 Skenario <i>Use Case</i> Melihat Iklan Pada <i>Dashboard</i>	61
Tabel 4.49 Skenario <i>Use Case</i> Melihat Grafik Wawasan.....	61
Tabel 4.50 Skenario <i>Use Case</i> Melihat Status Terakhir.....	62
Tabel 4.51 Skenario <i>Use Case</i> Melihat Set Iklan Pada <i>Dashboard</i>	62
Tabel 4.52 Skenario <i>Use Case</i> <i>Compare</i> Set Iklan	63
Tabel 4.53 Skenario <i>Use Case</i> Hapus <i>Compare</i> Set Iklan	63
Tabel 4.54 Skenario <i>Use Case</i> Melihat Log	64
Tabel 4.55 Skenario <i>Use Case</i> Melihat Notifikasi.....	64
Tabel 4.56 Skenario <i>Use Case</i> <i>Logout</i>	65
Tabel 5.1 Spesifikasi Perangkat Keras	78
Tabel 5.2 Spesifikasi Perangkat Lunak	79
Tabel 5.3 Spesifikasi Sistem Operasi	79
Tabel 5.4 Spesifikasi Server	79
Tabel 5.5 Implementasi Basis Data Dengan DDL	80
Tabel 6.1 Hasil Pengujian Unit Pada Fungsi tambahSetIklan.....	91

Tabel 6.2 Hasil Pengujian Unit Pada Fungsi getIklanGrafik.....	95
Tabel 6.3 Hasil Pengujian Unit Pada Fungsi tambahKampanye.....	96
Tabel 6.4 Hasil Pengujian Integrasi Pada Fungsi tambahSetIklan.....	100
Tabel 6.5 Kasus Uji Berhasil Login	104
Tabel 6.6 Kasus Uji Login Alternatif 1.....	104
Tabel 6.7 Kasus Uji Login Alternatif 2.....	105
Tabel 6.8 Kasus Uji Berhasil Lupa Password	105
Tabel 6.9 Kasus Uji Lupa Password Alternatif 1	106
Tabel 6.10 Kasus Uji Lupa Password Alternatif 2	106
Tabel 6.11 Kasus Uji Lupa Password Alternatif 3	107
Tabel 6.12 Kasus Uji Berhasil Melakukan Pendaftaran.....	107
Tabel 6.13 Kasus Uji Melakukan Pendaftaran Alternatif 1	108
Tabel 6.14 Kasus Uji Berhasil Melihat Preface	108
Tabel 6.15 Kasus Uji Berhasil Melihat 3 Komponen Iklan Facebook	108
Tabel 6.16 Kasus Uji Berhasil Melihat ROAS dan ROMI.....	109
Tabel 6.17 Kasus Uji Berhasil Melihat Informasi <i>About SIDAMAR</i>	109
Tabel 6.18 Kasus Uji Berhasil Melihat Dokumentasi.....	110
Tabel 6.19 Kasus Uji Berhasil Login Facebook	110
Tabel 6.20 Kasus Uji Login Facebook Alternatif 1	110
Tabel 6.21 Kasus Uji Berhasil Menambah Kampanye.....	111
Tabel 6.22 Kasus Uji Menambah Kampanye Alternatif 1	112
Tabel 6.23 Kasus Uji Menambah Kampanye Alternatif 2	112
Tabel 6.24 Kasus Uji Menambah Kampanye Alternatif 3	113
Tabel 6.25 Kasus Uji Berhasil Menampilkan Data Seluruh Kampanye	113
Tabel 6.26 Kasus Uji Berhasil Mengedit Kampanye.....	113
Tabel 6.27 Kasus Uji Mengedit Kampanye Alternatif 1	114
Tabel 6.28 Kasus Uji Mengedit Kampanye Alternatif 2	114
Tabel 6.29 Kasus Uji Berhasil Menghapus Kampanye	115
Tabel 6.30 Kasus Uji Menghapus Kampanye Alternatif 1	115
Tabel 6.31 Kasus Uji Berhasil Memulai Kampanye	116
Tabel 6.32 Kasus Uji Berhasil Menjeda Kampanye	116
Tabel 6.33 Kasus Uji Berhasil Menambah Set Iklan	116



Tabel 6.34 Kasus Uji Menambah Set Iklan Alternatif 1.....	117
Tabel 6.35 Kasus Uji Menambah Set Iklan Alternatif 2.....	117
Tabel 6.36 Kasus Uji Berhasil Melihat Set Iklan	118
Tabel 6.37 Kasus Uji Berhasil Mengedit Set Iklan	118
Tabel 6.38 Kasus Uji Mengedit Set Iklan Alternatif 1.....	118
Tabel 6.39 Kasus Uji Mengedit Set Iklan Alternatif 2.....	119
Tabel 6.40 Kasus Uji Berhasil Melihat Detail Set Iklan.....	119
Tabel 6.41 Kasus Uji Berhasil Menghapus Set Iklan.....	119
Tabel 6.42 Kasus Uji Menghapus Set Iklan Alternatif 1	120
Tabel 6.43 Kasus Uji Berhasil Memulai Set Iklan	120
Tabel 6.44 Kasus Uji Berhasil Menjeda Set Iklan	121
Tabel 6.45 Kasus Uji Berhasil Menambah Iklan	121
Tabel 6.46 Kasus Uji Menambah Iklan Alternatif 1.....	121
Tabel 6.47 Kasus Uji Menambah Iklan Alternatif 2.....	122
Tabel 6.48 Kasus Uji Berhasil Melihat Iklan	122
Tabel 6.49 Kasus Uji Berhasil Memulai Iklan	123
Tabel 6.50 Kasus Uji Berhasil Menjeda Iklan	123
Tabel 6.51 Kasus Uji Berhasil Mengedit Iklan	123
Tabel 6.52 Kasus Uji Mengedit Iklan Alternatif 1.....	124
Tabel 6.53 Kasus Uji Mengedit Iklan Alternatif 2.....	124
Tabel 6.54 Kasus Uji Berhasil Menghapus Iklan.....	125
Tabel 6.55 Kasus Uji Menghapus Iklan Alternatif 1	125
Tabel 6.56 Kasus Uji Berhasil Menambah Ad Creative	125
Tabel 6.57 Kasus Uji Menambah Ad Creative Alternatif 1.....	126
Tabel 6.58 Kasus Uji Berhasil Melihat Ad Creative	126
Tabel 6.59 Kasus Uji Berhasil Memotong Gambar	127
Tabel 6.60 Kasus Uji Memotong Gambar Alternatif 1	127
Tabel 6.61 Kasus Uji Berhasil Melihat Preview Ad Creative	128
Tabel 6.62 Kasus Uji Melihat Preview Ad Creative Alternatif 1	128
Tabel 6.63 Kasus Uji Berhasil Mengedit Iklan	128
Tabel 6.64 Kasus Uji Mengedit Mengedit Ad Creative Alternatif 1	129
Tabel 6.65 Kasus Uji Mengedit Ad Creative Alternatif 2.....	129



Tabel 6.66 Kasus Uji Berhasil Menghapus Ad Creative.....	130
Tabel 6.67 Kasus Uji Menghapus Ad Creative Alternatif 1	130
Tabel 6.68 Kasus Uji Berhasil Menambah Ad Pixel	130
Tabel 6.69 Kasus Uji Menambah Ad Pixel Alternatif 1.....	131
Tabel 6.70 Kasus Uji Menambah Ad Pixel Alternatif 2.....	131
Tabel 6.71 Kasus Uji Menambah Ad Pixel Alternatif 3.....	132
Tabel 6.72 Kasus Uji Berhasil Melihat Ad Pixel	132
Tabel 6.73 Kasus Uji Berhasil Mengedit Ad Pixel	133
Tabel 6.74 Kasus Uji Mengedit Mengedit Ad Pixel Alternatif 1	133
Tabel 6.75 Kasus Uji Mengedit Ad Pixel Alternatif 2.....	133
Tabel 6.76 Kasus Uji Berhasil Set Halaman Website.....	134
Tabel 6.77 Kasus Uji Set Halaman Website Alternatif 1	134
Tabel 6.78 Kasus Uji Set Halaman Website Alternatif 2	135
Tabel 6.79 Kasus Uji Berhasil Mengedit Halaman Website	135
Tabel 6.80 Kasus Uji Mengedit Halaman Website Alternatif 1.....	136
Tabel 6.81 Kasus Uji Mengedit Halaman Website Alternatif 2.....	136
Tabel 6.82 Kasus Uji Berhasil Menggenerate Ad Pixel.....	137
Tabel 6.83 Kasus Uji Berhasil Copy to Clipboard.....	137
Tabel 6.84 Kasus Uji Berhasil Mengganti Akun Iklan	137
Tabel 6.85 Kasus Uji Mengganti Akun Iklan Alternatif 1.....	138
Tabel 6.86 Kasus Uji Berhasil Melihat Sisa Anggaran	138
Tabel 6.87 Kasus Uji Berhasil Melihat Iklan Pada Dashboard.....	139
Tabel 6.88 Kasus Uji Berhasil Melihat Grafik Wawasan.....	139
Tabel 6.89 Kasus Uji Berhasil Melihat Status Terakhir.....	140
Tabel 6.90 Kasus Uji Melihat Status Terakhir Alternatif 1	140
Tabel 6.91 Kasus Uji Berhasil Melihat Set Iklan Pada Dashboard	141
Tabel 6.92 Kasus Uji Berhasil Compare Set Iklan	141
Tabel 6.93 Kasus Uji Compare Set Iklan Alternatif 1.....	141
Tabel 6.94 Kasus Uji Berhasil Hapus Compare Set Iklan	142
Tabel 6.95 Kasus Uji Hapus Compare Set Iklan Set Iklan Alternatif 1	142
Tabel 6.96 Kasus Uji Berhasil Melihat Log	143
Tabel 6.97 Kasus Uji Melihat Log Alternatif 1	143

Tabel 6.98 Kasus Uji Berhasil Melihat Notifikasi 144
Tabel 6.99 Kasus Uji Berhasil Logout 144



BAB 1 PENDAHULUAN

1.1 Latar belakang

Internet Marketing merupakan sebuah bentuk usaha untuk mengenalkan dan memasarkan *product* atau jasa kepada pelanggan melalui media *internet*. Kelebihan menggunakan *Internet Marketing* dibandingkan dengan *Conventional Marketing* adalah karena lebih cepat dan tepatnya penyebaran informasi promosi produk yang akan diiklankan. Selain itu, dengan menggunakan strategi *Internet marketing* yang baik dan benar dapat membuat harga promosi yang jauh lebih terjangkau, dengan target pemirsa yang spesifik untuk wilayah yang luas (Kotler Phillip, 2008). Salah satu cara untuk melakukan *Internet marketing* adalah dengan menggunakan situs sosial media atau bisa disebut dengan *Social Media Advertising* (SMA). SMA adalah suatu metode untuk mengiklankan, mempromosikan dan menjual produk maupun jasa dengan menggunakan sosial media sebagai alat untuk *marketing*. Keunggulan promosi dengan menggunakan metode SMA adalah akan terjadinya sebuah dampak secara langsung terhadap keputusan pembelian karena sosial media akan membuat terjalinnya hubungan sosial antara pembeli dan penjual secara tidak langsung (Swastha, 2012). Salah satu sosial media yang dapat dimanfaatkan untuk melakukan SMA adalah Facebook. Facebook merupakan situs web yang pertama kali dibuat oleh Mark Zuckenberk pada tahun 2004 yang berhasil menduduki peringkat ke-3 situs teratas dunia (Alexa, 2018) dan di Indonesia facebook menduduki peringkat ke-2 sebagai media sosial paling aktif (We Are Social, 2018).

Digital Marketing Malang (DMM) merupakan sebuah usaha yang bergerak dibidang *marketing online* atau *internet marketing* yang menyediakan jasa berupa pemasangan iklan berbayar beserta optimasinya pada *social media* dan *search engine*. Dalam pelaksanaan usahanya terdapat sebuah permasalahan utama yang dialami dan dirasakan oleh pihak DMM. Permasalahannya adalah ketika mengawasi banyaknya iklan yang berjalan pada Facebook Ads. Banyaknya iklan ini disebabkan oleh penerapan strategi iklan berbayar yang disebut dengan *splitting*. *Splitting* adalah sebuah strategi memecah iklan dengan komposisi berbeda pada satu produk yang diiklankan untuk mendapatkan iklan yang paling efektif. Dikarenakan banyak iklan yang berjalan dengan pengawasan yang kurang *realtime*, akan membuat ludesnya seluruh anggaran iklan berbayar dengan sangat cepat atau dalam istilah DMM disebut boncos. Istilah ini muncul karena anggaran yang dikeluarkan untuk beriklan tidak diimbangi dengan hasil yang diharapkan. Perlu diketahui untuk mendapatkan iklan yang benar-benar efektif, dibutuhkan teknik *splitting* dalam jumlah yang banyak untuk satu waktu sekaligus dan ketika semua iklan yang berjalan bersamaan telah menghasilkan wawasan iklan maka saat itu juga akan dianalisis dan dihentikan untuk iklan yang tidak efektif dan akan difokuskan untuk iklan yang efektif. Padahal wawasan iklan ini biasanya muncul secara *random* dan tidak dapat diprediksi (Wowo, 2019).

Dari permasalahan diatas, diperlukannya adanya sebuah aplikasi yang dapat digunakan untuk memonitoring iklan secara *realtime* yang bisa dimanfaatkan oleh pihak *marketer* DMM untuk menghentikan iklan secara otomatis dalam satu waktu pada saat *splitting*, mengetahui berapa konversi yang telah dihasilkan dari web yang diiklankan, mengetahui jumlah iklan berjalan dan jumlah iklan yang berhenti. Selain itu juga diperlukan sebuah metrik efektivitas iklan *Return on Ads Spending* (ROAS) dan *Return on Marketing Investment* (ROMI) sebagai ukuran untuk analisa efektifitas SMA tambahan.

Berdasarkan permasalahan tersebut, maka akan dibangun sebuah aplikasi *dashboard* yang dapat digunakan oleh *marketer* untuk *monitoring* iklan facebook secara terpusat dalam satu lokasi web berbentuk *dashboard system*. *Dashboard* merupakan sebuah tampilan yang merepresentasikan informasi atau data penting untuk menggapai informasi secara objektif yang ditata dan diatur kedalam satu layar sehingga suatu informasi bisa diawasi dalam sekejap untuk menghasilkan keputusan tepat dan cepat (Few, 2006). Dalam *dashboard system* ini juga akan dilengkapi dengan fitur untuk membuat satu set iklan lengkap (terdiri dari: Kampanye, set iklan, dan iklan), membuat ad pixel, membuat ad creative menghentikan iklan secara otomatis, memilih iklan dengan CTR terbaik, melihat wawasan iklan yang berjalan, melihat wawasan halaman website (*page view* dan konversi), serta menampilkan metrik efektifitas iklan dengan menampilkan skor ROAS dan ROMI dengan judul "Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*". Dimana aplikasi ini akan dibangun menggunakan basis *website* dengan menggunakan metode *waterfall*. Pada bagian pengujian, akan dilakukan *unit testing* yang berfokus pada bagian terkecil dari suatu aplikasi, pengujian integrasi untuk menguji penggabungan dari beberapa unit serta terdapat pengujian validasi yang digunakan untuk menguji fungsionalitasnya dan pengujian *compatibility testing* atau pengujian kompatibilitas yang digunakan untuk menguji non fungsionalnya.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah diuraikan pada subbab 1.1, maka diperoleh perumusan masalah yang terdiri dari:

1. Bagaimana hasil yang diperoleh dari penggalan dan analisis kebutuhan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*?
2. Bagaimana hasil perancangan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* yang dibuat berdasarkan analisis kebutuhan yang telah didapatkan?
3. Bagaimana hasil dari implementasi Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* berdasarkan perancangan sistem?
4. Bagaimana hasil pengujian dari Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* setelah dibangun?

1.3 Tujuan

Tujuan utama penelitian yang akan dilakukan adalah sebagai berikut.

1. Menganalisis dan menyusun kebutuhan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* sesuai dengan kebutuhan.
2. Merancang aplikasi yang sesuai dengan hasil dari analisis kebutuhan yang telah didapatkan.
3. Mengimplementasikan hasil yang diperoleh dari perancangan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*.
4. Melakukan pengujian pada Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*.

1.4 Manfaat

Dalam Pembangunan aplikasi *dashboard* untuk analisa efektivitas *marketing online* berbasis *website* ini diharapkan penelitian ini dapat memberikan manfaat untuk:

Marketer Online:

1. Tidak khawatir untuk membuat banyak percobaan iklan agar mendapatkan racikan yang pas dan mendapatkan iklan dengan CTR tertinggi.
2. Dapat melihat wawasan iklan facebook dan wawasan website yang diiklankan secara satu layar *dashboard*.
3. Dapat melihat skor ROAS dan ROMI dalam bentuk grafik untuk memudahkan analisa efektivitas *marketing online*.
4. Dapat menghentikan iklan secara otomatis dan *realtime* untuk selain jumlah iklan yang akan diambil untuk difokuskan pada iklan yang akan diambil saja sehingga lebih menghemat anggaran iklan.

1.5 Batasan masalah

Agar tidak terjadi pelencengan, maka akan diberikan ruang lingkup terhadap sebuah permasalahan dari penelitian yang akan dilakukan dengan tujuan untuk menghindari penyimpangan serta pelebaran berdasarkan tujuan awal penelitian yang telah direncanakan. Adapun beberapa batasan masalahnya adalah:

1. Aplikasi akan menggunakan media sosial facebook saja.
2. *Application Programming Interface* (API) yang digunakan hanya *Marketing API*, *Pages API* dan *Facebook Login API*.
3. Mekanisme pertukaran informasi API akan menggunakan *library PHP CURL*.
4. Aplikasi akan dibangun dengan platform web.
5. Aplikasi akan dibangun dengan PHP, HTML5, CSS3, Javascript, JQuery serta AJAX sebagai bahasa pemrogramannya.

6. Aplikasi akan dibangun menggunakan kerangka dasar atau *framework* PHP, yaitu Codeigniter.
7. Aplikasi akan menggunakan MySQL sebagai database.
8. Aplikasi akan berjalan jika terdapat koneksi internet.
9. Aplikasi akan bekerja dengan baik jika diakses dengan peramban yang *support*.

1.6 Sistematika pembahasan

Untuk melihat secara garis besar apa yang ada pada laporan ini, maka dibuatlah sistematika pembahasan yang terdiri dari:

BAB I PENDAHULUAN

Berisikan tentang sebuah latar belakang masalah yang dihadapi, kemudian rumusan sebuah masalah yang didapatkan, Tujuan dari diadakannya penelitian yang ingin dicapai, untuk siapakah manfaat dari penelitian ini, dan batasan masalah agar tidak keluar dari lingkup permasalahan.

BAB II LANDASAN KEPUSTAKAAN

Berisikan metode dan teori-teori yang pernah digunakan dalam penelitian sebelumnya yang diambil sebagai inspirasi dan dasar pada penelitian yang akan dikerjakan.

BAB III METODOLOGI PENELITIAN

Berisi tahapan-tahapan dalam sebuah penelitian dan menjelaskan apa saja yang digunakan dalam setiap tahapan Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*.

BAB IV ANALISIS KEBUTUHAN

Analisis kebutuhan akan berbicara masalah apa yang dibutuhkan oleh aktor yang akan digunakan sebagai pondasi Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website*.

BAB V PERANCANGAN DAN IMPLEMENTASI

Bab Perancangan dan Implementasi menjabarkan tentang perancangan mulai dari perancangan komponen, perancangan antarmuka dan perancangan sistem dari data yang telah di dapatkan dari rekayasa kebutuhan. Kemudian pada sub bab selanjutnya berisi implementasi dari Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* yang mengacu pada perancangan aplikasi yang dibuat sebelumnya.

BAB VI PENGUJIAN

Bab pengujian akan berisi seputar pengujian beserta teknik pengujian yang akan dilakukan dari aplikasi yang telah dibangun kemudian akan didapatkan analisis

hasil pengujian terhadap aplikasi.

BAB VII PENUTUP

Bab penutup adalah bab terakhir dari metodologi penelitian yang berisikan pengambilan kesimpulan setelah semua tahap telah selesai dan akan diberikan saran untuk yang akan melanjutkan penelitian ini.



BAB 2 LANDASAN KEPUSTAKAAN

Pada bab dua akan dibahas terkait dengan kajian pustaka serta teori mendasar yang akan dipakai sebagai referensi untuk mendukung penyusunan penelitian. Pembahasan kajian pustaka pada penelitian ini digunakan untuk pembandingan serta inspirasi yang diambil dari penelitian sebelumnya yang mirip atau berkaitan. Kemudian dalam penyusunan penelitian ini juga akan disajikan dasar teori yang meliputi:

2.1 Kajian Pustaka

Referensi yang akan diambil sebagai kajian pustaka dalam penelitian ini adalah yang hampir sama atau berkaitan dengan *online marketing* terutama yang berkaitan dengan pemanfaatan *dashboard* sebagai solusi atau alat untuk analisa efektivitas iklan sehingga bisa meningkatkan atau memaksimalkan trafik menuju website yang diinginkan yang berdampak pada penjualan barang.

Penelitian pertama adalah penelitian yang dilakukan oleh Pichit Ngamjarussrivichai, Papada Jeemali, dan Wuttipong Panitsettakorn dengan judul "*Designing an Online Marketing Investment Evaluation System Case study of Cosmetics Manufacturing Company*". Penelitian ini memiliki tujuan untuk membuat aplikasi dashboard yang dapat memonitor penjualan, wawasan tentang pelanggan serta untuk mengetahui efektivitas dari sebuah iklan dengan cara menghitung ROAS serta ROMI pada studi kasus iklan kosmetik *online* pada Google Ads. Alasan penelitian ini menggunakan Google Ads adalah karena pada produk kosmetik ini perusahaan mengincar pasar orang yang mencari kosmetik melalui pencarian google. Adapun untuk langkah-langkah penelitian ini adalah:

1. Membuat suatu konseptual *framework* yang menggambarkan mulai dari *input* suatu masalah yang menyebabkan suatu penelitian, proses untuk mendapatkan data kemudian mendesain suatu sistem evaluasi *marketing*, sampai *output* berupa grafik ROAS, ROMI serta biaya perjalanan pelanggan.
2. Mencari data dengan cara observasi pada perusahaan kosmetik yang berguna untuk menggali kebutuhan yang akan digunakan untuk implementasi fitur sistem evaluasi pemasaran secara *online*.
3. Melakukan pengukuran dari metrik yang telah dihasilkan oleh Google Ads, diantara parameter yang diamati adalah impressi, klik ke website, konversi (telepon kontak, isi form, E-mail, add line@), lead, *office visit*, *new customer*, *revenue from new customer*, kemudian yang terakhir adalah menghitung efektivitas iklan dari ROAS dan ROMI.

Penelitian kedua yang dilakukan oleh Jimmy Moedjahedy yang berjudul "*Implementasi Cron Job Linux Sebagai Bel Pergantian Kelas Otomatis Di Universitas Klabat*". Penelitian ini memiliki tujuan utama yaitu untuk membangun bel otomatis yang berbunyi pada setiap jam yang telah ditentukan dengan

menggunakan *task scheduler* Cron Job pada linux. Alasan dipilihnya *task scheduler* pada linux dibandingkan dengan *task scheduler* pada windows adalah karena *software* yang bersifat gratis dengan fungsional yang sama. Dari hasil penelitian ini didapatkan bahwa dengan menggunakan Cron Job sebagai *task scheduler* adalah sistem dapat berjalan dengan baik serta konsisten dalam menjalankan *task* kecuali terdapat permasalahan teknis terkait kabel karena terdapat perbaikan bangunan, bel juga dapat berjalan dengan tepat sesuai jam yang telah diatur dengan konsistem dibandingkan dengan bel manual dengan hanya melihat jam tangan dari petugas.

Dari kedua penelitian ini didapatkan bahwa untuk mengukur suatu efektifitas dalam iklan online bisa menggunakan suatu *key performance indicator* (KPI) berupa ROAS dan ROMI. Dan dalam penyajian wawasan digunakan suatu evaluasi dari *Customer Journey* untuk melihat impresi atau jumlah tayangan dan jumlah konversi yang digunakan untuk menghitung ROAS yang disajikan dalam bentuk *dashboard system*. kemudian didapatkan bahwa dengan menggunakan cron job dapat digunakan sebagai *task scheduler* otomatis sehingga akan mengeksekusi pada waktu yang ditentukan, dan berdasarkan hasil penelitian kedua didapatkan bahwa cron job cukup konsisten dan tepat waktu dalam mengeksekusi perintah yang akan dijalankan.

Pada penelitian yang akan dilaksanakan adalah dengan menggunakan platform sosial media facebook, karena di Indonesia sendiri facebook menduduki peringkat ke-2 sebagai sosial media teraktif (We Are Social, 2018). Untuk mengetahui *Customer Journey* seperti kunjungan dan konversi akan digunakan facebook pixel sebagai alat untuk memperoleh wawasan yang berasal dari website yang sedang diiklankan. Untuk menghentikan iklan secara otomatis akan digunakan perbandingan skor CTR guna mendapatkan iklan dengan racikan yang pas menggunakan cron job sebagai *task scheduler* yang akan melakukan pengecekan kondisi iklan *splitting* yang sedang dijalankan. Untuk mengetahui apakah iklan dijalankan dengan efektif atau tidak akan menggunakan skor ROAS dan ROMI.

2.2 Digital Marketing Malang (DMM)

Digital marketing malang adalah sebuah usaha yang bergerak dalam bidang jasa yang melayani *online advertising* seperti *Social Media Marketing* (SMM) dan *Search Engine Marketing*, pelayanan pembuatan website serta pembuatan konten untuk web maupun iklan. DMM berdiri sejak pertengahan 2018 di Kota Malang dan sampai saat ini memiliki 3 karyawan *marketer* dan 2 *developer website*. (Wowo, 2019).

Dalam menyediakan jasanya, DMM menyediakan beberapa paket yang bisa dipilih diantaranya:

A. Paket *online advertising plus website*

1. Paket *standard* (Rp.1.750.000) dengan rincian: Domain, 500MB *disk hosting capacity, non database, unlimited bandwidth data transfer*.
2. Paket *custom* (Rp.2.500.000) dengan rincian: Domain, 1GB *disk capacity, Database, unlimited bandwidth data transfer, Simple theme*.
3. Paket *powered custom* (Rp. 4.000.000) dengan rincian: Domain, 2GB *disk capacity, Database, unlimited bandwidth data transfer, Premium theme by request*.
4. Paket *full* (Rp.8.500.000) dengan rincian: Domain, 4GB *disk capacity, Database, Automatic E-commerce system ready, IDR currency, All Indonesian shipping service, Exclusive theme. unlimited bandwidth data transfer. It's almost like you have your own marketplace*.

B. Paket *online advertising only*

1. Paket *Advertising* dengan harga Rp.750.000

Semua paket yang disediakan sudah termasuk untuk anggaran beriklan dengan jumlah 30%-40% dari harga setiap paket yang dipilih dan jika merasa kurang maka bisa menambah anggaran iklan secara *custom* sesuai yang diinginkan. Ketika ingin menggunakan jasanya, *customer* selain memilih paket yang diinginkan juga harus menyediakan *fanspage* atau halaman facebook yang akan digunakan untuk SMM Facebook Ads, Foto produk, serta deskripsi dari produk yang akan diiklankan. Jika belum mempunyai kelengkapan tersebut, *customer* bisa melakukan *request custom* diluar paket dengan biaya tambahan untuk melengkapi seluruh kebutuhan *advertising* seperti pembuatan halaman atau *fanspage* serta fotografi. Untuk menggunakan jasa DMM biasanya *customer* memesan melewati *website*, whatsapp, atau bisa langsung ke lokasi DMM. Dalam satu bulan, biasanya DMM menerima pesanan 9-10 barang untuk diiklankan (Wowo, 2019).

2.3 Rekayasa Perangkat Lunak

Adalah merupakan sebuah metode yang sistematis atau tersusun, disiplin, dan dapat diukur untuk merekayasa perangkat lunak. Karena perangkat lunak bukan produk fabrikasi jadi tidak bisa mengalami aus atau kelelahan sehingga harus direkayasa (Pressman, 2009). Rekayasa perangkat lunak juga disebut sebagai disiplin ilmu yang mengambil sebuah pendekatan rekayasa yaitu bagaimana metode yang digunakan, bagaimana prosesnya, bagaimana cara mengubah atau mengorganisasinya, bagaimana menjamin kualitasnya sehingga bisa mengembangkan perangkat lunak skala besar dengan produktivitas yang tinggi, terkontrol, dan terjadwal, tapi dengan biaya yang rendah (Mohapatra, 2010). Jadi dapat disimpulkan bahwa rekayasa perangkat lunak adalah sebuah pendekatan yang sistematis yang harus dilalui dengan berbagai fase untuk menghasilkan perangkat lunak yang berkualitas dengan harga yang rendah.

2.4 Pengembangan Perangkat Lunak

Proses pengembangan dari sebuah perangkat PL dari subbab metode *software development life cycle* (SDLC) dan pendekatan pengembangan yang akan dipakai dalam membangun aplikasi, yaitu pendekatan berorientasi objek.

2.4.1 *Software Development Life Cycle* (SDLC)

SDLC atau bisa disebut dengan daur hidup perangkat lunak adalah suatu siklus yang menggambarkan tahapan dalam proses pengembangan perangkat lunak. Menurut sumber yang lain, SDLC tradisional yang biasa digunakan pada saat ini adalah berisi suatu kerangka model yang terstruktur yang berisikan tahapan sebuah sistem yang akan dibangun (R. Kelly Rainer, 2009). SDLC mempunyai berbagai model yang bisa dipakai yaitu: *Waterfall* (Waterfall & V), *Increment Process Model*, dan *Evolutionary Process Models* (Prototyping & Spiral).

Model *waterfall* akan dipilih menjadi model utama dalam penelitian ini. Model ini memiliki alur hidup perangkat lunak seperti air terjun yaitu terus mengalir kebawah tanpa kembali, karena *waterfall* mempunyai tahapan secara berurutan atau sekuensial yang biasanya dimulai dari tahap analisis kebutuhan, desain atau perancangan, pengkodean atau implementasi, pengujian dan terakhir tahap *support* atau pemeliharaan (Rosa and Shalahudin, 2011). *Waterfall* termasuk kedalam model klasik karena bersifat sistematis, berurutan dan tidak bisa diulang dalam membangun perangkat lunak, untuk gambar model *waterfall* bisa dilihat pada Gambar 2.1. Menurut (Sommerville, 2010) tahapan dalam model SDLC *waterfall* adalah sebagai berikut.

1. *Requirement/Analisis* Kebutuhan

Proses analisa kebutuhan, membutuhkan komunikasi antara pengembang dengan pengguna untuk mendapatkan informasi seperti apa perangkat lunak yang diinginkan, memahami kebutuhan pengguna, serta batasan ketika akan membuat atau mengembangkan perangkat lunak yang diinginkan. Bentuk komunikasi yang bisa dilakukan adalah dengan wawancara, diskusi, maupun survei. Setelah informasi dapat digali dan dikumpulkan, maka selanjutnya pengembang atau *developer* bisa menganalisis kebutuhan apa saja yang akan dibutuhkan dalam perangkat lunak yang akan dibuat.

2. *System Design* atau Desain Sistem

Berdasarkan hasil analisa kebutuhan pada tahap sebelumnya, maka dalam tahap ini akan ditentukan desain sistem seperti apa yang akan digunakan atau dimodelkan pada perangkat lunak yang akan dibuat. Dari fase desain sistem ini, dapat terlihat gambaran arsitektur dari sebuah sistem yang akan dibuat, bagaimana aktor dapat berinteraksi dengan sistem, bagaimana sebuah proses mempengaruhi proses yang lain dalam bentuk yang berurutan atau sekuensial, bagaimana sebuah basis data berelasi satu sama lain, dan bagaimana antarmuka yang diharapkan sesuai dengan kebutuhan.

3. *Implementation* atau Implementasi

Tahap implementasi adalah sebuah tahapan dimana sistem akan mulai diimplementasikan berdasarkan hasil perancangan yang diperoleh dari tahap desain sistem. Tahapan implementasi yang dilakukan pada penelitian ini adalah spesifikasi sistem, dilanjutkan dengan implementasi basis data, implementasi kode program dalam bentuk unit atau fungsi dan yang terakhir adalah implementasi antarmuka.

4. *Testing* atau Pengujian

Pengujian merupakan suatu tahap pada SDLC yang berguna untuk menguji perangkat lunak apakah terdapat kesalahan atau *bug* atau tidak. Langkah awal dalam pengujian pada penelitian ini adalah menggunakan unit testing atau pengujian masing-masing fungsi untuk mengetahui kesalahan pada tiap-tiap unit. Kemudian dilakukan pengujian integrasi untuk menguji hubungan antar unit untuk mencapai suatu kebutuhan fungsional yang lengkap secara satu kesatuan.

5. *Maintenance* atau Pemeliharaan

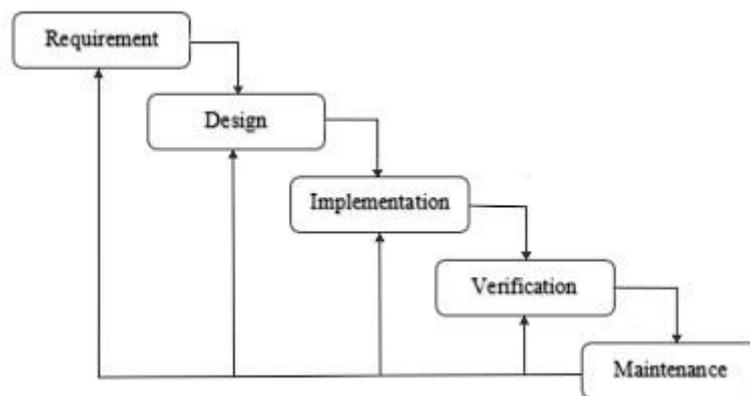
Sebuah aplikasi yang telah melewati pengujian dan bisa dijalankan dengan baik, tahap selanjutnya adalah pemeliharaan perangkat lunak. Segala *error* yang telah ditemukan pada saat melakukan pengujian akan diubah dan diperbaiki pada tahap *maintenance*.

Adapun kelebihan dari model *waterfall* ini adalah sebagai berikut.

1. Penerapannya lebih mudah karena hanya menggunakan satu iterasi dalam pengembangan perangkat lunak.
2. Pendefinisikan kebutuhan sistem yang mudah karena hanya dilakukan pada awal proyek saja.

Sedangkan model *waterfall* memiliki kekurangan sebagai berikut.

1. Sulit terjadinya perubahan ketika mendapatkan kesalahan atau error pada saat pengembangan, dikarenakan model *waterfall* memiliki sifat seperti aliran air terjun maka yang terjadi adalah setiap tahapan cenderung kaku dan tidak bisa untuk kembali ke tahapan pengembangan yang telah dilalui.
2. Digunakan buat proyek yang besar yang di *develop* secara *multisite* yaitu dikerjakan pada banyak tempat, karena tujuan awal yang sudah jelas, kebutuhan yang telah terdefinisi dengan baik dan cara mencapainya juga sangat jelas (Pressman, 2009).



Gambar 2.1 *Waterfall Model*

2.4.2 Pendekatan Berorientasi Objek

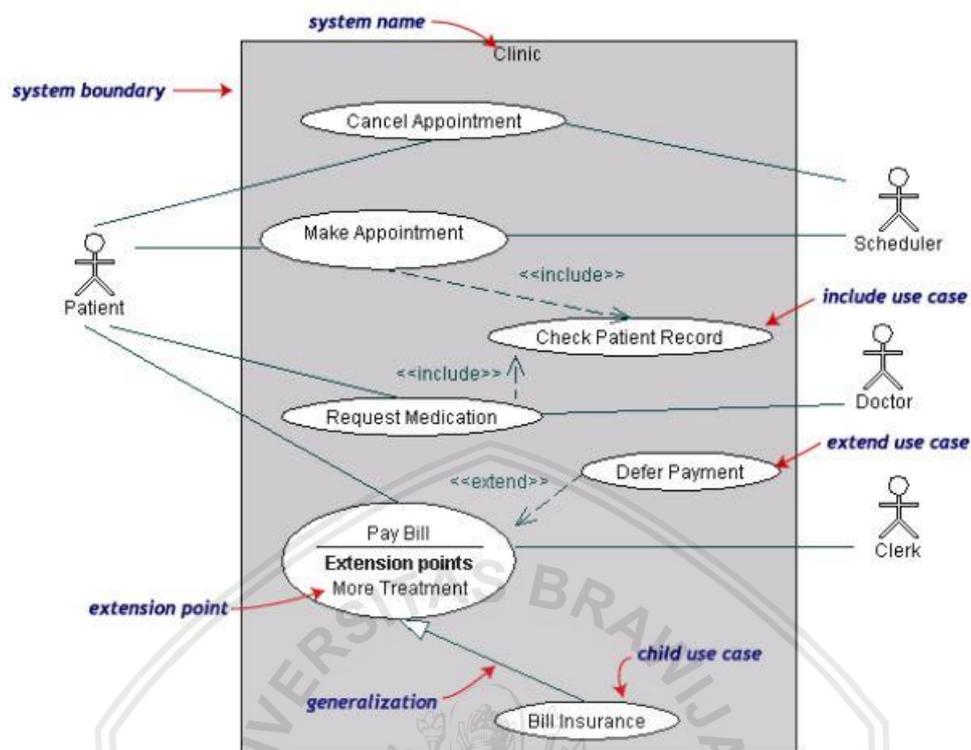
Pendekatan OO adalah salah satu pendekatan dengan paradigma bahwa semua fungsi maupun data harus dibungkus dalam suatu *class* atau objek yang didalamnya bisa memproses data, serta bisa saling berinteraksi satu dengan yang lainnya (Sakur, 2010). Desain pada OO akan memfokuskan data desainnya kedalam suatu *object* maupun *class* seperti pada skenario di dunia nyata yang menegaskan perilaku atau *behavior*, keadaan atau *state*, dan interaksi *object*. OO juga memberikan manfaat akan kebebasan dalam proses *developing*, peningkatan kualitas perangkat lunak, mempermudah *maintenance* atau pemeliharaan, meningkatkan cara untuk modifikasi serta meningkatkan *reusable software* (J.E.N.I, 2007).

2.5 Unified Modelling Language (UML)

UML adalah *standar language* dari sebuah pemodelan yang dipakai untuk melakukan berbagai macam model perancangan (Dharwiyanti & Wahono, 2003). Dengan menggunakan UML yang standar, maka akan dapat mempermudah proses pengembangan *software* pada saat pemodelan baik pemodelan kebutuhan maupun pemodelan perancangan. Dokumentasi pada UML juga menjadikan proses penelusuran kode program saat *maintenance* menjadi jauh lebih cepat.

2.5.1 Diagram Use Case

Diagram *Use Case* adalah merupakan suatu UML sistem dari sisi fungsionalitas yang memiliki arti “apa” yang bisa dilakukan aktor terhadap sistem. Dengan memodelkan daftar kebutuhan menjadi *use case* diagram maka suatu kebutuhan akan menjadi lebih mudah untuk dipahami karena direpresentasikan dengan menggunakan simbol-simbol yang sederhana (Dharwiyanti and Wahono, 2003).



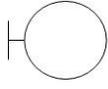
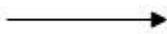
Gambar 2.2 Contoh Diagram Use Case

Dari Gambar 2.2 menunjukkan bagaimana contoh dari diagram use case. Pada gambar tersebut terlihat empat aktor yang akan menggunakan sistem. Kemudian terdapat tiga bentuk asosiasi, diantaranya adalah *include*, *extend*, dan generalisasi. Asosiasi *extend* memiliki arti bahwa fungsional *extended use case* tidak bisa dilakukan ketika fungsionalitas *base use case* tidak dilakukan. Sebaliknya, fungsionalitas *base use case* tetap bisa dijalankan meskipun fungsionalitas *extended use case* tidak dijalankan atau *extended use case* bisa disebut bersifat opsional. Untuk asosiasi *include*, memiliki arti bahwa *base use case* tidak bisa dijalankan ketika *include use case* tidak dijalankan atau bersifat komposisi terhadap *base use case*. Yang terakhir ada asosiasi generalisasi, yang menandakan bawa *parent* aktor menjadi aksi umum yang diturunkan pada *child* aktor, atau biasa disebut dengan pewarisan.

2.5.2 Diagram Sequence

Diagram *Sequence* merupakan bentuk pemodelan UML yang merepresentasikan *interraction* antar objek yang bersifat berurutan atau sekuensial pada suatu fungsionalitas sistem (Dharwiyanti and Wahono, 2003). Fungsi yang lain adalah untuk penggambaran skenario yang dilakukan ketika objek dihidupkan dan kemudian dapat mengirim dan menerima *message* pada objek yang lain. Komponen diagram *sequence* biasanya terdiri dari *entity class*, *boundary class*, *control class*, *message*, *recursive* dan *activation* seperti pada Tabel 2.1.

Tabel 2.1 Komponen *Sequence Diagram*

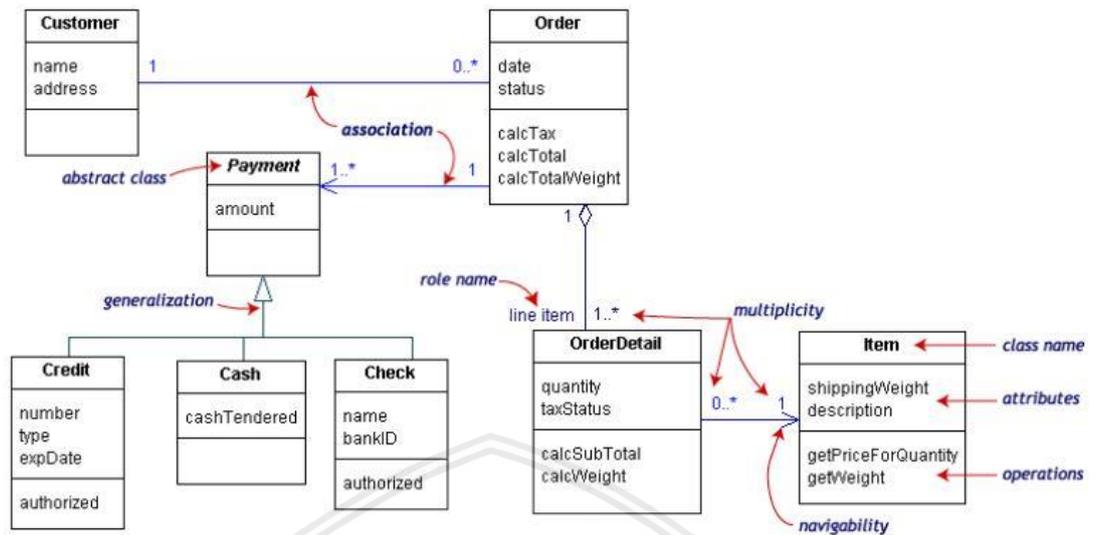
Gambar	Keterangan
	Kelas <i>Entity</i> menggambarkan sebuah komponen yang menjadi database dari sistem yang akan dibangun.
	Kelas <i>Boundary</i> merupakan komponen yang merupakan jembatan yang menghubungkan antar aktor dengan sistem menggunakan tampilan visual.
	Kelas <i>Control</i> berisi <i>logic</i> atau logika sebagai pengatur antara kelas <i>entity</i> dan kelas <i>boundary</i> .
	<i>Synchronous Message</i> direpresentasikan dengan panah dengan ujung penuh yang merupakan sebuah simbol untuk pengiriman pesan antar objek.
	<i>Recursive</i> direpresentasikan dengan panah ujung penuh yang menunjuk ke dirinya sendiri yang memiliki arti pengiriman pesan untuk dirinya sendiri.
	<i>Activation</i> direpresentasikan seperti persegi panjang vertikal yang menggambarkan objek yang dihidupkan dan akan melakukan eksekusi operasi pada suatu objek yang lain.
	<i>Asynchronous Message</i> direpresentasikan dengan panah dengan ujung tidak penuh yang merupakan symbol untuk mengirimkan pesan secara <i>Asynchronous</i> seperti ajax.
	<i>Callback</i> direpresentasikan dengan panah ke arah kiri yang menunjukkan <i>callback</i> dari objek yang dipanggil dengan menggunakan ajax.
	<i>Return Message</i> direpresentasikan dengan panah ke arah kiri yang merupakan sebuah symbol yang menunjukan balikan atau <i>return message</i> dari objek yang telah dipanggil melalui <i>message</i> .

Sumber: www.uml-diagrams.org

2.5.3 Diagram *Class*

Diagram *Class* menggambarkan struktur bagaimana sebuah *class* dideskripsikan mulai dari *package*, objek serta hubungan antar kelas satu dengan kelas yang lainnya (Dharwiyanti and Wahono, 2003). *Class* diagram dapat menunjukkan komponen atribut dan fungsi dalam satu model yang sederhana. Komponen *class* diagram terdiri dari atribut, operasi atau fungsi, *relation* (*asosiation*, *agregation*, dan generalisasi), serta visibilitas dari sebuah atribut maupun fungsi.





Gambar 2.3 Contoh Class Diagram

Sumber : (Dharwiyanti and Wahono, 2003)

2.6 Model View Controller (MVC)

MVC adalah sebuah kerangka kerja yang melakukan pendekatan dengan memisahkan antara *File view* (Tampilan), *File controller* (Pengatur atau pengontrolan), dan *File model* (database). Ketiga fokus file tersebut kemudian dibagi pada tiap folder sehingga menjadi struktur proyek yang lebih rapi dan akan memudahkan proses *maintenance* (Deacon, 2005).

2.7 Codeigniter

Codeigniter adalah sebuah salah satu *framework php open source* yang menerapkan konsep kerangka kerja MVC. Didalam codeigniter juga menyediakan fungsionalitas unggulan seperti *library*, *helper*, *magic method* yang tentu saja sangat membantu dalam mempercepat proses pengembangan perangkat lunak. Codeigniter juga pernah diakui oleh pembuat PHP yakni Rasmus Lerdorf sebagai *framework PHP* yang cepat pada saat frOSCon pada tahun 2008 (Daqiqil, 2011).

Keunggulan *framework* codeigniter:

1. Gratis alias tidak bayar ketika menggunakannya.
2. Sintaks menjadi lebih rapi dan terstruktur.
3. Sangat mudah untuk digunakan.
4. Adanya fitur *helper* dan *library* yang membantu mempercepat pengerjaan proyek.
5. Sudah menyediakan sistem keamanan yang cukup bagus untuk sebuah *framework*.

Kekurangan dari *framework* codeigniter dibandingkan dengan *framework* PHP yang lainnya adalah:

1. Ketersediaan *library* yang terbatas.
2. *Update framework* yang sangat lambat dibandingkan dengan *framework* PHP yang lain seperti Laravel sehingga mulai ditinggalkan.
3. Sangat cocok untuk membangun website yang berskala kecil tapi tidak untuk membangun web dengan skala besar.

2.8 PHP

PHP adalah sebuah bahasa pemrograman *backend* yang berjalan pada sisi server. Dengan menggunakan PHP kita bisa membuat fitur dinamis pada sebuah *website* seperti melakukan *login* dan penyimpanan serta me-*retrieve* data pada *server*. Mekanisme eksekusi pada bahasa pemrograman PHP adalah semua permintaan akan dikirim dan dieksekusi pada sisi *server* yang hasilnya akan dikirim kembali pada peramban berbentuk HTML yang akan di *render* oleh *browser* pengguna sehingga akan menghasilkan tampilan visual yang dapat dibaca oleh pengguna. Itulah kenapa ketika *inspect element* pada browser tidak akan ditampilkan kode pemrograman PHP yang digunakan sehingga menjamin keamanan dari pencurian *source code* oleh orang yang tidak bertanggung jawab (Arief, 2011).

2.9 CURL

CURL adalah sebuah *library* yang digunakan untuk mengirim dan mengambil data melalui URL (*Uniform Resource Locator*) dengan protokol. CURL memungkinkan untuk saling berkomunikasi antar *server* dengan protokol yang berbeda. CURL sendiri menyediakan dua macam fungsi yaitu *libcurl* dan *curl*. *Libcurl* adalah nama dari sebuah *library* yang digunakan untuk akses ke berbagai protocol sedangkan *curl* adalah istilah untuk sebuah sintaks atau baris Perintah untuk mengirimkan dan menerima data dengan menggunakan sintaks URL (Khalid, 2006). CURL digunakan untuk komunikasi dengan API Facebook baik itu untuk melakukan pengambilan data atau GET ataupun untuk mengirim data atau POST melalui *HTTP Request*.

2.10 Cron Job

Cron Job merupakan sebuah fitur *task scheduler* yang berjalan pada sistem operasi Linux yang digunakan untuk melakukan perintah, *file batch*, *script* secara otomatis atau *by system* dengan jadwal yang telah ditentukan sebelumnya. Dengan menggunakan Cron Job ini, bisa digunakan untuk memanggil suatu *function* secara berkesinambungan setiap jam, hari, atau minggu tertentu dan *system* akan otomatis mengeksekusi *function* sesuai dengan jadwal.

2.11 Facebook API

Facebook API merupakan suatu fitur yang disediakan oleh facebook untuk mempermudah pengguna terutama pengembang dalam mengakses fitur-fitur yang terdapat dalam Facebook seperti autentifikasi dengan login facebook, mendapatkan akses untuk mengambil data analisis iklan, untuk bisa mempublikasikan dari aplikasi pihak ketiga dll. Dengan adanya API atau *Application Programming Interface* yang disediakan oleh facebook developer dapat mempercepat proses pengembangan perangkat lunak karena tidak membuat suatu fitur dari nol.

Dalam penelitian yang dilakukan oleh penulis digunakanlah API Facebook *Login*, dan API *Marketing* yang setiap API memiliki peran:

1. API Facebook *Login*

API Facebook *login* digunakan untuk *login* pada aplikasi pihak ketiga dengan menggunakan autentifikasi yang disediakan oleh facebook sehingga ketika *login* pada aplikasi pihak ketiga kita hanya perlu memasukkan id dan *password* sama persis dengan yang digunakan untuk *login* ke facebook dengan keamanan dari API Login Facebook. Data yang dikirimkan akan dikembalikan dalam bentuk JSON yang datanya bisa diambil dan dimanfaatkan untuk memberikan ID dan nama sesuai dengan ID dan nama facebook.

2. API *Marketing*

API *Marketing* digunakan untuk membuat iklan yang terdiri dari nama iklan, penargetan, budget, dan apa yang diinginkan ketika mengiklankan seperti klik ke website tertentu, *like fanspage*, kunjungi halaman dan kemudian bisa mengawasi iklan setelah iklan dipublikasikan atau bisa melihat berapa impresi tampilan yang telah dijangkau, berapa *budget* yang telah dihabiskan, berapa jumlah klik, berapa interaksi dll. Yang kemudian dengan hasil iklan bisa dianalisis apakah iklan yang telah dipublikasikan efektif atau tidak dengan menghitung ROAS dan ROMI.

2.12 Hyper Text Markup Language (HTML)

Merupakan dokumen atau file berisi bahasa *markup* yang berisi instruksi-instruksi yang kemudian dapat di-*render* oleh *browser* komputer pada pengguna yang menghasilkan tampilan informasi visual pada peramban (Yeni Kustiyahningsih, 2011).

HTML memiliki 3 komponen penting yang harus ada, yaitu sebagai berikut.

- a. *Element*, adalah suatu baris kode atau seperti sebuah kalimat yang harus diapit oleh dua komponen tanda (*tag*) yaitu: *tag* pembuka, isi kalimat, dan *tag* penutup. Contohnya sederhananya adalah: `<h1>Hello World</h1>`.
- b. *Tag* adalah sebuah teks yang diapit oleh karakter `<` dan `>`. Contoh dari tag adalah: `<h1>`, `<h2>`, `<head>`, dan `<div>`.

- c. *Atribut* merupakan *script* yang berisi *properties* dan *value* yang berada dalam sebuah *element* yang berfungsi memberikan informasi tambahan pada suatu *element*. *Value* dari suatu *atribut* harus berada setelah tanda (=) dan berada dalam tanda kutip. Contoh atribut: `<div class="iniValue"></div>`

2.13 Javascript

Bahasa pemrograman web yang satu ini berjalan pada sisi *browser* (meskipun pada saat ini terdapat javascript yang berjalan pada sisi server dengan nodejs) sehingga ketika dilakukan *inspect element* maka kode program yang ditulis dengan javascript akan dapat dilihat. Fungsi utama dari javascript adalah untuk memanipulasi dan memetakan element HTML menjadi sebuah DOM (*Document Object Model*) dan membuat halaman web menjadi lebih interaktif (Suehering, 2013).

2.14 jQuery

jQuery adalah javascript *library* yang dibuat dan dirancang untuk menyederhanakan *syntax* javascript sehingga dapat mempercepat proses pengembangan perangkat lunak. JQuery memiliki slogan yang menyatakan eksistensinya sebagai penyederhana javascript, yaitu "*Write less, do more*". Pertama dibuat pada tahun 2006 oleh seseorang bernama John Resig. Adapun kelebihan lain yang dimiliki oleh jQuery adalah sebagai berikut.

1. Kemudian mengakses dan memanipulasi DOM (Document Object Model) atau element HTML.
2. Penanganan efek HTML.
3. Memanipulasi CSS.
4. Penanganan *event* pada HTML.

Sintak dasar pada jQuery agak berbeda dengan javascript karena lebih sederhana sebagai contoh untuk mengakses selector (class/id) dalam sebuah HTML yaitu : `$(selector).action()` akan digantikan dengan sintak `$` saja sebagai pendefinisian jQuery (Beighley, 2010)

2.15 AJAX

AJAX (*Asynchronous Javascript and XML*) adalah sebuah metode untuk melakukan pembangunan web secara dinamis dan interaktif. Dengan adanya AJAX dapat dilakukan pengiriman data dari *server* secara *background process* akan tetapi tidak semua komponen halaman web ditransmisikan setiap *user* mengirimkan *request*. Hal ini membuat peningkatan kecepatan dan interaktif pada sebuah web. Secara contoh sederhana ketika kita menggunakan ajax adalah saat mengirimkan data dari sebuah *form* ke server maka akan terjadi *loading page*, nah ketika menggunakan AJAX saat mengirimkan data *form* ke *server* maka data akan

di kirim secara *background process* jadi tidak akan terlihat *loading page* sehingga web kita menjadi lebih interaktif.

2.16 MySQL

MySQL sebenarnya merupakan sebuah manajemen database relasional atau biasa disebut dengan *Relational Database Management System* (RDBMS). MySQL AB adalah perusahaan yang pertama kali membuat MySQL sebelum diakuisisi oleh Oracle *Corporation* pada tahun 2010 sampai sekarang (Tjahyadi, et al., 2007). Untuk mempermudah penggunaan MySQL seperti tampilan yang lebih bagus, dan fungsional yang lebih mudah digunakan maka dapat menggunakan sebuah klien untuk MySQL. Adapun *client* MySQL yang dapat dipakai adalah melalui *commandline*, PHPMyAdmin, Navicat, HeidiSQL dan MySQL Workbench.

2.17 Dashboard

Dashboard adalah sebuah tampilan visual satu layar yang menampilkan informasi penting dengan harapan dapat mudah dipantau dan cepat dipahami oleh user untuk mencapai sebuah tujuan dengan hanya diawasi secara sekilas (Few, 2006). Setidaknya terdapat empat kriteria yang harus dimiliki oleh sebuah *dashboard* menurut Novell (2004), yaitu sebagai berikut.

1. Menyediakan informasi yang relevan dan menyajikannya dalam satu pandang.
2. Informasi disampaikan secara cepat, tepat serta akurat.
3. Terdapat mekanisme keamanan yang baik untuk melindungi informasi sensitif atau penting agar tidak jatuh pada orang yang tidak berkepentingan.
4. Dapat memberikan informasi menyeluruh, artinya terdapat solusi terkait dengan domain masalah yang ada.

Dashboard sendiri memiliki setidaknya tiga tipe yaitu: (1) *Strategical Dashboard* yang berfungsi sebagai pendukung garis perusahaan yaitu menyediakan informasi cepat yang digunakan untuk pengambilan keputusan dengan memonitor kesehatan dan peluang dalam sebuah bisnis, (2) *Tactical Dashboard* yang digunakan sebagai pendukung pengukuran progress dalam kunci atau inisiatif proyek, dan (3) *Operational Dashboard* yang digunakan sebagai pendukung *monitoring* dari aktifitas proses bisnis yang spesifik yang memberikan informasi tentang aktivitas yang sedang terjadi beserta perubahannya secara *real time* untuk memberikan kewaspadaan terhadap hal yang perlu direspon secara cepat (Few, 2006).

Adapun dalam penelitian ini akan digunakan *dashboard* dengan tipe *Operational Dashboard* sebagai *monitoring* untuk melihat hasil wawasan atau informasi dari iklan beserta website yang akan diiklankan yang menghasilkan skor efektifitas iklan berupa *Key Performasnce Indicator* (KPI) yang akan diperbarui secara otomatis setiap 1 menit sekali dan *splitting* 1 jam sekali.

2.18 Marketing Online

Marketing Online atau *Online marketing* adalah suatu pemasaran yang dilakukan dengan memanfaatkan internet sebagai alat untuk melakukan aktivitas pemasaran. Pemasaran sendiri adalah bentuk dari interaksi sosial baik individu maupun kelompok untuk saling menawarkan sesuatu untuk mencapai pemenuhan kebutuhan masing-masing (Kotler, 2014).

Internet *marketing* atau *online marketing* adalah sebuah proses pemasaran yang terkandung aktivitas *online*, baik yang berupa ide, produk maupun jasa dengan tujuan untuk kepuasan pelanggan (Rafi Mohammed, Robert J. Fisher, Bernard Jaworski, 2003). Dalam definisi yang lain menyebutkan bahwa *marketing online* adalah suatu kegiatan komunikasi pemasaran yang dilakukan dengan media internet (Dianawati, 2007).

Dalam pemasaran *online* yang penulis ambil adalah dengan menggunakan facebook sebagai media pemasarannya. Alasannya karena facebook menduduki peringkat 3 situs teratas dunia setelah google dan youtube (Alexa, 2018). Di Indonesia sendiri facebook menduduki peringkat ke-2 sebagai sosial media teraktif setelah youtube (We Are Social, 2018). Sehingga peluang pemasaran *online* melalui facebook sangatlah terbuka.

2.19 Click Through Rate (CTR)

CTR adalah sebuah metrik yang digunakan sebagai pengukuran jumlah klik iklan yang didapatkan dari klik iklan yang dijalankan dibagi dengan jumlah *impression* yang didapatkan kemudian dikalikan dengan 100. CTR akan memberikan wawasan bagaimana kondisi iklan yang dijalankan, apakah dengan banyaknya *impression* akan sebanding dengan jumlah orang yang tertarik dengan iklan. CTR disebut juga sebagai *The Essential Operational Effectiveness Metric* yang berarti sangatlah berguna untuk permulaan perhitungan performa iklan atau dapat dikatakan sebagai metrik identifikator awal untuk melihat apakah iklan berjalan secara efektif atau tidak (Jeffery, 2010).

2.20 Return on Ads Spending (ROAS)

ROAS adalah suatu cara yang sederhana untuk melakukan pengukuran pendapatan berdasarkan metrik yang diterima yang kemudian dilakukan pembagian dari biaya iklan berbayar sehingga didapatkan skor ROAS (Allen, 2018). Dewasa ini *marketing online* yang terbaik adalah dengan menggunakan data, Metrik dan wawasan iklan tidak hanya berbicara terkait peningkatan sebuah trafik tetapi seorang *marketer* harus bisa mengetahui berapa pendapatannya setelah dihitung dengan biaya iklan berbayar. ROAS akan menghasilkan sebuah laporan berapa tepatnya pendapatan setelah dilakukan iklan berbayar dan mana iklan yang paling efektif menghasilkan pendapatan terbesar.

Jika skor ROAS mendapatkan nilai dibawah 300% maka bisa dipastikan mengalami kerugian. Jika 300% maka pengeluaran anggaran untuk sebuah iklan mendapatkan perbandingan *revenue* dan *expense ads* 3:1 yang menunjukkan kemungkinan masih mengalami kerugian. Jika mendapatkan skor ROAS 400% atau perbandingan *revenue* dan *expense ads* 4:1 maka akan mengalami *turning to profit* yang berarti akan mulai berbalik mendapatkan keuntungan. Jika skor ROAS 500% atau diatasnya, maka iklan memiliki performa yang bagus dan sudah pasti mendapatkan keuntungan (Andrus, 2018).

2.21 Return on Marketing Investment (ROMI)

ROMI adalah cara yang digunakan untuk menghitung *margin* yang diperoleh dari pembagian keuntungan bersih dengan biaya iklan berbayar. Biaya dari iklan berbayar disebut juga sebagai investasi pemasaran, karena tujuannya adalah untuk mendapatkan keuntungan, *market share*, *customer equity*, dan *brand equity*. Konsep investasi pemasaran pada dasarnya seperti investasi pada umumnya yaitu setidaknya terdapat empat elemen: *Expenditure or Investment*, *Return*, *Risk*, *Hurdle Rates*. ROMI memiliki konsep yang mencakup keempat elemen tersebut. ROMI adalah sebuah alat yang digunakan untuk mengkonsep program pemasaran, perencanaan, dan parameter penganggaran iklan yang berguna untuk membantu keberhasilan mencapai target dalam beriklan, memilih prioritas, *gain approval*, eksekusi dan manajemen iklan, serta memonitor iklan. (Powell, 2002).

ROMI bisa dikatakan profit atau mendapatkan keuntungan jika skor yang diperoleh adalah nilai positif dan jika mendapatkan nilai positif semakain besar artinya semakin bagus dan jika mendapatkan nilai negatif maka bisa dipastikan mendapatkan kerugian atau tidak mendapatkan keuntungan sama sekali (Amy Gallo, 2017).

2.22 Pengujian Perangkat Lunak

Definisi dari pengujian perangkat lunak adalah tahapan yang digunakan untuk menemukan *bug* atau kesalahan yang dibuat secara tidak sengaja pada saat perangkat lunak dirancang dan dibangun, kemudian kesalahan tersebut akan dicatat hasilnya yang kemudian akan diperbaiki dan dikembangkan (Pressman, 2009). Untuk melakukan pengujian setidaknya dibutuhkan hasil dari konfigurasi perangkat lunak yang telah didapatkan dan kemudian akan dibandingkan dengan hasil uji yang diharapkan sehingga akan diketahui apakah sudah benar atau tidak. Ketika dilakukan pengujian dan ditemukan kesalahan maka akan dilakukan perbaikan kode program dan kemudian akan dilakukan pengujian ulang. Dalam pengujian perangkat lunak akan digunakan *unit testing*, *integration testing* dan *validation testing* untuk pengujian fungsional dan *compatibility testing* untuk pengujian non-fungsional.

a. Pengujian Unit

Pengujian unit adalah salah satu metode pengujian yang dilakukan untuk memastikan logika dari unit terkecil dari perancangan perangkat lunak, komponen atau modul perangkat lunak (Pressman, 2009).

b. Pengujian Integrasi

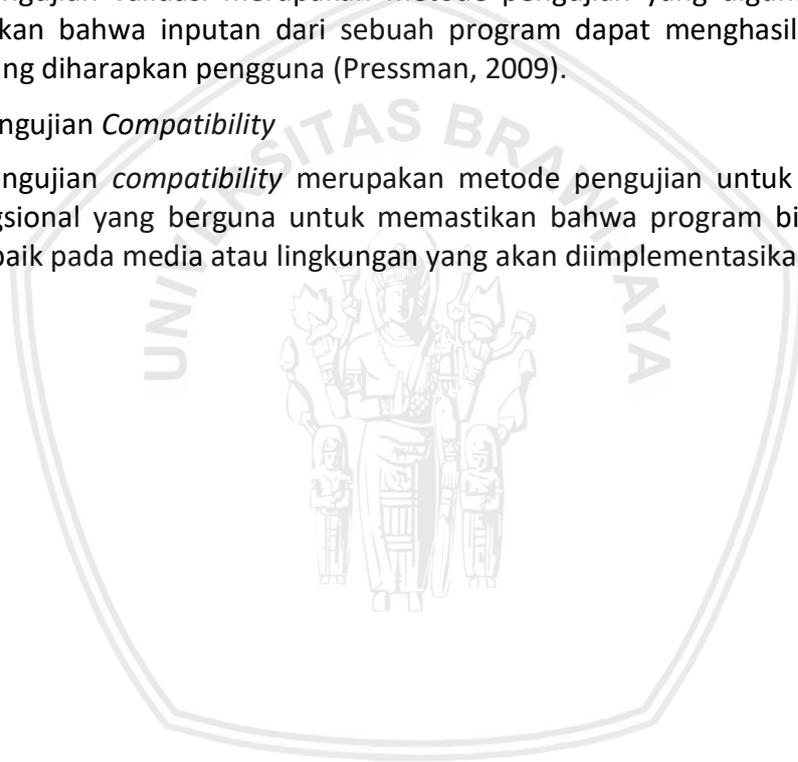
Pengujian integrasi adalah salah satu metode pengujian yang dilakukan untuk menguji keterkaitan hubungan antar *method* sebelum program telah jadi tetapi sudah bisa menjalankan sebuah fungsi sebagaimana mestinya.

c. Pengujian Validasi

Pengujian validasi merupakan metode pengujian yang digunakan untuk memastikan bahwa inputan dari sebuah program dapat menghasilkan *output* sesuai yang diharapkan pengguna (Pressman, 2009).

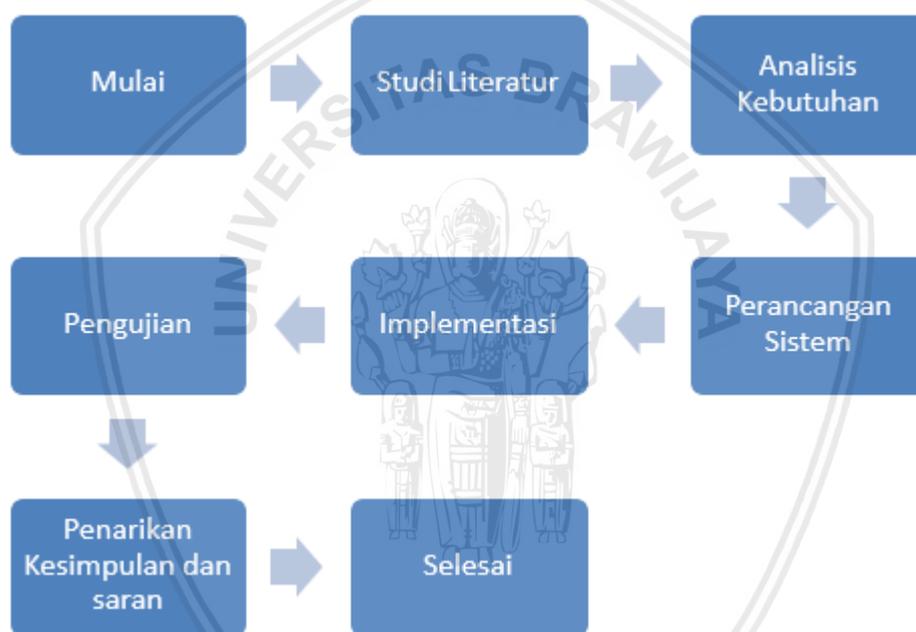
d. Pengujian *Compatibility*

Pengujian *compatibility* merupakan metode pengujian untuk kebutuhan non-fungsional yang berguna untuk memastikan bahwa program bisa berjalan dengan baik pada media atau lingkungan yang akan diimplementasikan.



BAB 3 METODOLOGI PENELITIAN

Metodologi penelitian berbicara masalah alur perencanaan dari sebuah penelitian yang akan digunakan. Penelitian yang akan dilakukan merupakan penelitian dengan bersifat *implementatif*, yaitu pengembangan (*development*). Adapun tahapan alur yang akan diambil adalah: melakukan studi literatur (Penelitian sejenis dan daftar kepustakaan dari istilah-istilah yang digunakan), penggalan dan analisis kebutuhan dari aktor, sistem *design* yang akan dibangun, implementasi dalam kode program, pengujian aplikasi yang dibangun, dan terakhir penarikan kesimpulan serta saran untuk penelitian selanjutnya. *Flow chart* atau Diagram alir dari metodologi yang akan dipakai pada penelitian bisa diketahui dari Gambar 3.1.



Gambar 3.1 Diagram Alir Penelitian

3.1 Studi Literatur

Pada subbab ini akan dijelaskan beberapa referensi penelitian sebelumnya serta pengertian istilah-istilah yang berasal dari berbagai macam sumber seperti jurnal ilmiah atau buku yang dapat dipertanggung jawabkan, Studi kepustakaan atau literatur yang akan dipakai diantaranya adalah:

1. SDLC atau *System Development Life Cycle*
2. UML atau *Unified Modeling Language*
3. MVC atau *Model View Controller*
4. Codeigniter
5. HTML

6. PHP
7. CURL
8. Cron Job
9. *Javascript*
10. jQuery
11. AJAX
12. MySQL
13. *Online Marketing*
14. Dashboard
15. CTR (*Click Through Rate*)
15. ROAS (*return on ads spending*) dan ROMI (*return on marketing investment*)
16. Facebook *Login*, dan *Marketing API*

3.2 Analisis Kebutuhan

Analisis kebutuhan adalah dasar pondasi dalam pembangunan *software* atau perangkat lunak agar menjadi berkualitas dan sesuai dengan keinginan pengguna. Analisis kebutuhan merupakan bagian dari proses kebutuhan perangkat lunak yang tujuannya adalah menjembatani rekayasa sistem dengan perancangan perangkat lunak (Pressman, 2009).

Analisis kebutuhan dilakukan dengan cara mengidentifikasi kebutuhan-kebutuhan pengguna pada Aplikasi *Dashboard Untuk Analisa Efektivitas Marketing Online Berbasis Website*. Pada tahap analisis kebutuhan ini dilakukan dengan menggunakan metode wawancara untuk mendapatkan informasi terkait bagaimana cara menjalankan iklan secara efektif dan apa permasalahan yang didapatkan ketika menjalankan iklan tersebut.

3.3 Perancangan Sistem

Tahapan selanjutnya adalah melakukan perancangan sistem. Tahapan ini akan dilaksanakan ketika telah mendapatkan daftar kebutuhan yang telah *fix* yang berasal dari aktor. Kebutuhan tersebut akan di *design* atau dimodelkan menjadi UML. Adapun UML yang akan dibuat antara lain: *Class Diagram*, *Sequence Diagram*, *Entity Relationship Diagram (ERD)*, serta *Physical Data Model (PDM)*.

3.4 Implementasi

Tahap implementasi akan merubah perancangan sebelumnya menjadi kode pemrograman teknis sesuai dengan kebutuhan yang membentuk suatu unit, yang kemudian unit-unit tersebut diintegrasikan menjadi satu kesatuan untuk membentuk satu sistem perangkat lunak utuh. Kode pemrograman yang akan

dipakai dalam pembangunan aplikasi adalah bahasa pemrograman PHP, Javascript, Ajax, CSS, HTML, dan MySQL. Kode ini kemudian akan diorganisasikan dalam satu kerangka Kerja atau *framework* PHP Codeigniter serta *Frontend Framework* menggunakan Bootstrap dan dibangun untuk platform berbasis *website*.

3.5 Pengujian

Pengujian bertujuan untuk menemukan *bug*, *defect* atau kesalahan yang tidak disengaja dan memastikan bahwa seluruh kebutuhan perangkat lunak sudah sesuai dengan kebutuhan user. Karena pada dasarnya tidak ada perangkat lunak yang terbebas dari *bug* atau *error*. Metode pengujian yang akan dilakukan adalah dengan *unit testing*, *integration testing* dan *validation testing* Jika pada saat pengujian ditemukan kesalahan atau *error* dan menyebabkan fungsionalitas menjadi cacat maka akan dilaksanakan perbaikan kesalahan atau *error* pada perangkat lunak yang kemudian akan diuji kembali.

3.6 Kesimpulan dan Saran

Kesimpulan dan saran merupakan tahapan terakhir dalam metodologi penelitian. Berdasarkan perancangan dan hasil pengujian didapatkan sebuah kesimpulan yang dapat menjawab rumusan masalah pada bab 1. Setelah kesimpulan diambil maka akan ada pemberian saran yang ditujukan untuk pembaca untuk memperbaiki kekurangan sistem pada pengembangan selanjutnya.

BAB 4 ANALISIS KEBUTUHAN

Merupakan tahapan pertama dari sebuah pengembangan perangkat lunak. Analisis ini digunakan untuk mengetahui kebutuhan-kebutuhan yang digunakan untuk melakukan percobaan guna memberikan solusi dari masalah yang dialami oleh Digital Marketing Malang saat mempromosikan iklan milik klien. Teknik yang digunakan penulis untuk melakukan analisis kebutuhan adalah dengan menggunakan teknik wawancara. Pada analisis kebutuhan juga akan dideskripsikan aktor apa saja yang terlibat dan akan memakai aplikasi, fungsionalitas yang dapat dilakukan oleh setiap aktor, bagaimana pemodelan kebutuhan aktor yang disediakan oleh sistem, serta bagaimana deskripsi *flow* utama dan alternatif *flow* pada aplikasi.

4.1 Deskripsi Umum Sistem

Aplikasi *dashboard* untuk analisa efektivitas *marketing online* adalah suatu aplikasi yang dibangun untuk membantu usaha jasa *marketing online* dalam mengelola iklan online yang di iklankan pada media sosial facebook sehingga bisa mengetahui efektivitas suatu iklan yang dibuat serta untuk memonitoring kondisi iklan dalam satu set iklan secara otomatis sehingga tidak menghabiskan anggaran iklan yang sedang berjalan dengan cara mengambil CTR tertinggi sesuai dengan keinginan *marketer* dan menghentikan iklan selain yang diambil secara otomatis, selain itu terdapat juga terdapat skor ROAS dan ROMI yang digunakan sebagai analisa tambahan untuk melihat efektifitas iklan *online*. Terdapat juga wawasan website tujuan seperti: jumlah *page view*, jumlah konversi, harga setiap konversi, dan *revenue* yang diperoleh dengan menggunakan Facebook Pixel. Aplikasi ini dibangun dengan menggunakan *platform* berbasis *website* sehingga bisa diakses dimanapun dengan perangkat apapun dengan syarat terdapat koneksi internet. Aplikasi ini menyediakan fungsionalitas utama seperti membuat iklan, melihat wawasan iklan, melihat wawasan website, menambahkan dan *generate* facebook pixel, mengambil iklan dengan skor CTR tertinggi, menghentikan iklan yang lain secara otomatis dan melihat skor efektivitas iklan tambahan dengan menggunakan parameter ROAS dan ROMI.

4.2 Proses Rekayasa Kebutuhan

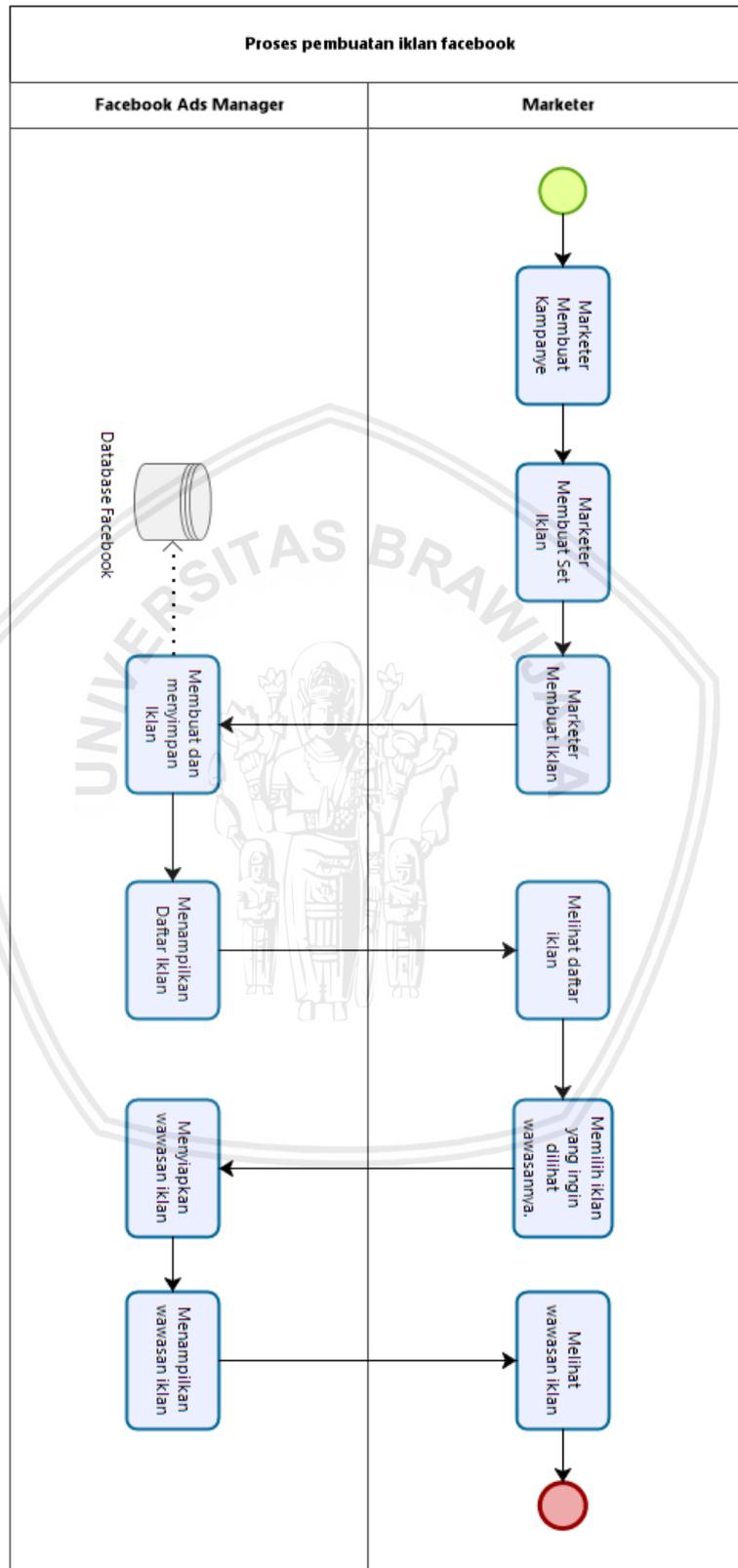
Proses rekayasa kebutuhan adalah salah satu langkah dalam analisis kebutuhan yang digunakan untuk menggali kebutuhan dari pemangku kebutuhan agar mendapatkan kebutuhan sistem yang tepat.

4.2.1 Elisitasi Kebutuhan dan Hasil Analisis

Elisitasi merupakan tahap penggalian kebutuhan yang digunakan untuk memahami aplikasi yang akan dirancang dan diimplementasikan. Dari hasil elisitasi dengan pemangku kepentingan maka didapatkan kebutuhan aplikasi yang didapatkan pada tabel 4.1.

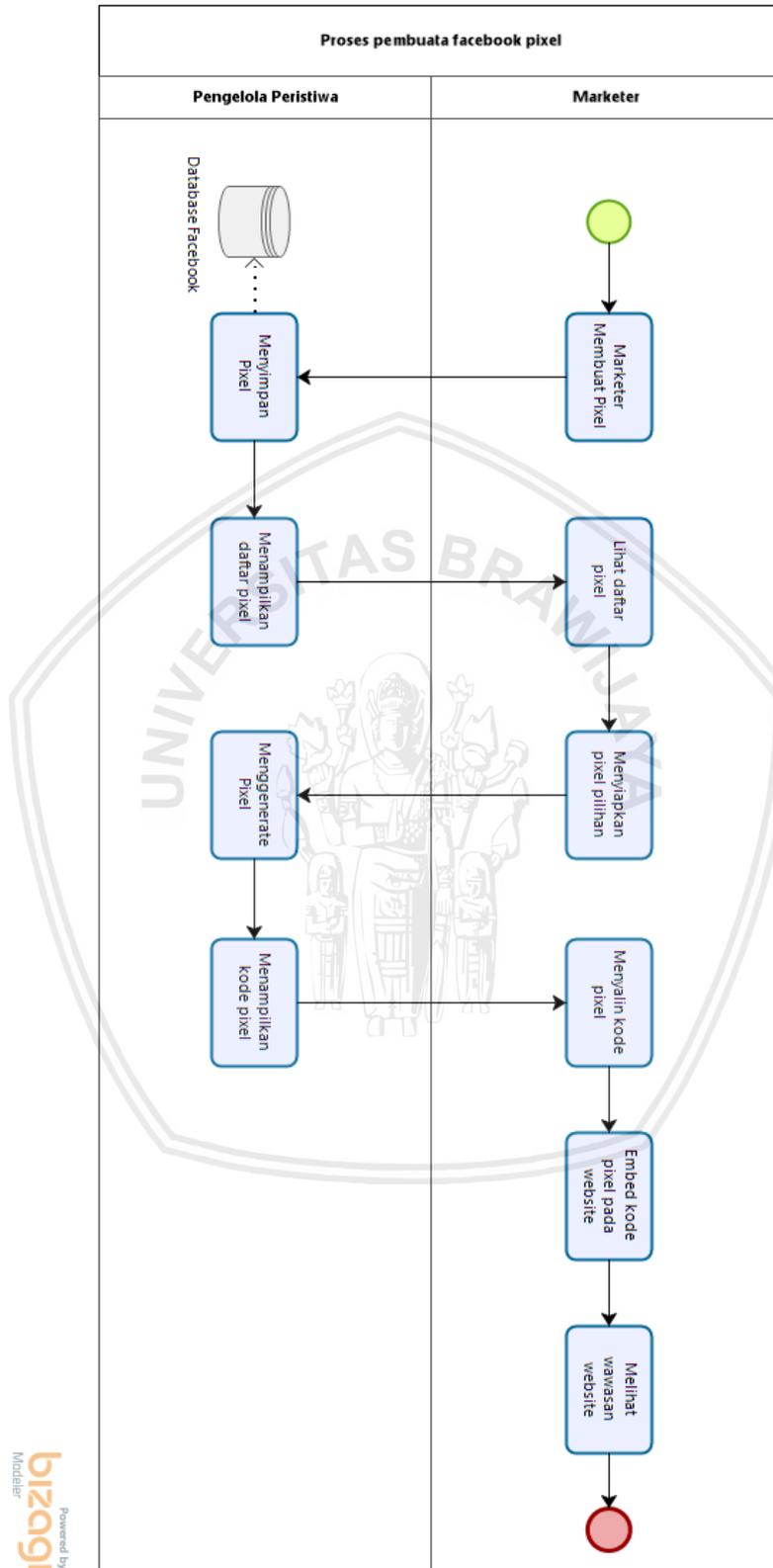
Tabel 4.1 Kebutuhan Hasil Wawancara

No.	Kebutuhan Yang Didapat
1.	DMM selain menerima jasa <i>online advertising</i> juga melayani pembuatan website beserta optimasinya. Memiliki 3 orang <i>marketer</i> dan 2 <i>web developer</i> .
2.	Pelanggan berasal dari seluruh Indonesia, dan jika ingin menggunakan jasa DMM bisa melewati website untuk melihat paket dan pemesanannya. Terdapat 5 paket yang disediakan yaitu 4 paket dengan pembuatan website dan 1 paket untuk <i>online advertising</i> saja. Dari paket yang diambil akan dialokasikan 30%-40% untuk anggaran iklan.
3.	Bagi pelanggan yang akan mengiklan harus menyediakan website, halaman atau <i>fanspage</i> facebook dan juga konten seperti foto atau gambar kemudian harus menjadikan DMM sebagai pengiklan atau admin. Jika tidak, maka bisa dibuatkan oleh DMM dengan harga sesuai paket.
4.	<i>Marketer</i> bisa mengiklankan produk di facebook secara sederhana dengan sedikit menu (yang sering digunakan DMM saja) dengan informasi hasil iklan dan informasi lintasan website pada satu aplikasi <i>dashboard</i> . Dan kalau bisa disertakan <i>key performance indicator</i> (KPI) sebagai analisis efektivitas tambahan.
5.	<i>Marketer</i> bisa melakukan <i>splitting</i> semaksimal mungkin tanpa khawatir habisnya anggaran dengan monitoring otomatis.
6.	<i>Marketer</i> ketika membuat kampanye menset <i>objective</i> (Tujuan Umum Iklan) jika tidak <i>link click</i> ya <i>conversion</i> .
7.	Pada set iklan, biasanya <i>marketer</i> menset <i>segmenting</i> pada lokasi, hobi, perangkat yang digunakan, umur minimal, umur maksimal, dan jenis kelamin.
8.	Aplikasi dapat berjalan pada <i>browser mainstream</i> seperti <i>Mozilla Firefox</i> , <i>Google Chrome</i> , <i>Microsoft Edge</i> , dan <i>Opera Mini</i> .
9.	Disediakan user pengguna sebelum masuk ke <i>marketer</i> , agar bisa membaca secara lengkap cara menggunakan aplikasi dan informasi aplikasi.

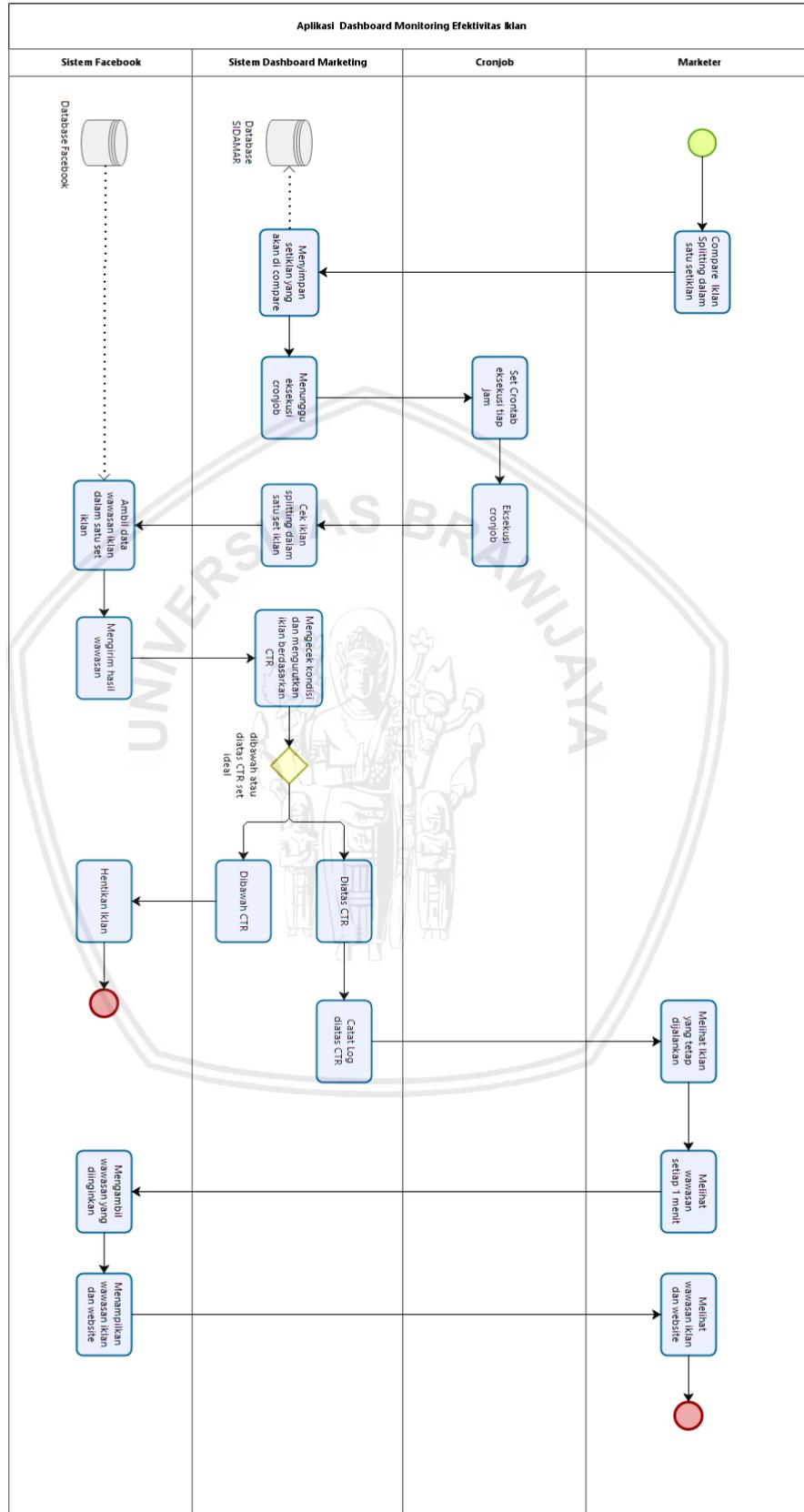


Gambar 4.1 Pembuatan Iklan dan Info Wawasan Sebelumnya





Gambar 4.2 Pembuatan Pixel dan Info Wawasan Website sebelumnya



Gambar 4.3 Aplikasi Dashboard Monitoring Efektifitas Iklan

4.3 Identifikasi Aktor

Identifikasi aktor bertujuan mendeskripsikan aktor yang akan menggunakan aplikasi. Aktor-aktor ini diperoleh dengan cara mengambil suatu sampel studi kasus yang telah digali menggunakan metode wawancara. Adapun aktor yang didapatkan adalah pengunjung, pengguna dan marketer. Identifikasi lengkap ditunjukkan pada Tabel 4.2.

Tabel 4.2 Identifikasi Aktor Sistem

No	Nama Aktor	Deskripsi
1.	Pengunjung	Aktor pengunjung merupakan aktor yang dapat membuka halaman <i>landing page</i> , <i>register</i> dan <i>login</i> .
2.	Pengguna	Pengguna adalah aktor yang sudah terdaftar dalam sistem tetapi peran akun iklan facebooknya masih belum teridentifikasi.
3.	Marketer	Marketing adalah aktor yang dapat membuat iklan, melihat wawasan, dan menghentikan iklan.

4.4 Kebutuhan Fungsional

Merupakan kebutuhan utama sebuah dari sistem yang didapatkan dari penggalan kebutuhan aktor. Kebutuhan fungsional ini akan dideskripsikan secara lengkap beserta spesifikasi kebutuhan yang menjelaskan secara mendetail kebutuhan tersebut. Kebutuhan fungsional yang akan digunakan oleh sistem berjumlah 52 kebutuhan beserta spesifikasinya yang secara mendetail ditunjukkan Tabel 4.3.

Tabel 4.3 Kebutuhan Fungsional

No	Kode	Aktor	Deskripsi Kebutuhan
1	SIDAMAR_F_0100	Pengunjung	Sistem harus menyediakan fungsionalitas <i>login</i> untuk dapat mengakses sistem sebagai pengguna, atau <i>marketer</i> .
1.1	SIDAMAR_F_0101	Pengunjung	Fungsionalitas <i>login</i> terdiri dari formulir <i>username</i> dan formulir <i>password</i> dengan karakter minimal enam huruf atau angka.
2	SIDAMAR_F_0200	Pengunjung	Sistem menyediakan fitur lupa <i>password</i> yang digunakan untuk mereset <i>password</i> .

No	Kode	Aktor	Deskripsi Kebutuhan
2.1	SIDAMAR_F_0201	Pengunjung	Sistem harus menyediakan form lupa <i>password</i> dengan masukan email yang telah terdaftar.
2.2	SIDAMAR_F_0202	Pengunjung	Sistem harus mengirimkan tautan <i>reset password</i> baru kepada email pengguna yang telah diinputkan dan telah terdaftar.
2.3	SIDAMAR_F_0203	Pengunjung	Sistem harus menyediakan halaman <i>reset password</i> yang terdiri dari <i>password</i> baru dan ulangi <i>password</i> yang digunakan untuk mengubah <i>password</i> ketika tautan dari email di klik.
3	SIDAMAR_F_0300	Pengunjung	Sistem harus menyediakan fungsi untuk pendaftaran.
3.1	SIDAMAR_F_0301	Pengunjung	Form untuk pendaftaran terdiri dari <i>full name, username, password, re type password, dan email</i> .
4	SIDAMAR_F_0400	Pengguna	Sistem harus dapat menampilkan menu <i>preface</i> yang berisi informasi awal sistem SIDAMAR.
5	SIDAMAR_F_0500	Pengguna	Sistem harus dapat menampilkan menu <i>3 Part of FB Ads</i> yang berisi informasi tentang kampanye, set iklan, dan iklan.
6	SIDAMAR_F_0600	Pengguna	Sistem harus dapat menampilkan menu <i>Introduce ROAS & ROMI</i> yang berisi informasi tentang <i>ROAS & ROMI</i> .
7	SIDAMAR_F_0700	Pengguna	Sistem harus bisa menampilkan <i>toggle</i> menu <i>About SIDAMAR</i> yang berisi informasi tentang SIDAMAR.
8	SIDAMAR_F_0800	Pengguna	Sistem harus dapat menampilkan menu <i>Complete Documentation</i> yang berisi informasi tentang dokumentasi cara menggunakan sistem.
9	SIDAMAR_F_0900	Pengguna	Sistem harus memiliki fungsionalitas <i>login</i> menggunakan Facebook

No	Kode	Aktor	Deskripsi Kebutuhan
			sehingga pengguna dapat mengakses sistem sebagai <i>marketer</i> .
10	SIDAMAR_F_1000	Marketer	Sistem harus menyediakan fitur untuk menambah kampanye
10.1	SIDAMAR_F_1001	Marketer	Pada langkah membuat kampanye, sistem harus menyediakan formulir yang berisi inputan nama kampanye, dan pilihan tujuan beriklan.
11	SIDAMAR_F_1100	Marketer	Sistem harus bisa menampilkan seluruh data kampanye yang telah dibuat.
12	SIDAMAR_F_1200	Marketer	Sistem harus menyediakan fitur untuk mengedit kampanye.
12.1	SIDAMAR_F_1201	Marketer	Data kampanye yang dapat diedit adalah nama kampanye, dan tujuan beriklan.
13	SIDAMAR_F_1300	Marketer	Sistem harus menyediakan fitur untuk menghapus kampanye yang telah terdaftar.
14	SIDAMAR_F_1400	Marketer	Sistem harus menyediakan fitur untuk memulai kampanye.
15	SIDAMAR_F_1500	Marketer	Sistem harus menyediakan fitur untuk menjeda kampanye.
16	SIDAMAR_F_1600	Marketer	Sistem harus menyediakan fitur untuk menambah set iklan
16.1	SIDAMAR_F_1601	Marketer	Pada langkah membuat set iklan, sistem harus menyediakan formulir yang berisi nama set iklan, pilihan kampanye, tujuan optimasi, bayar berdasarkan, harga setiap 1000, Anggaran harian, tanggal mulai set iklan dan tanggal berakhir set iklan, serta targeting (ketertarikan, kebiasaan, lokasi geografi, jenis kelamin, umur minimal, dan umur maksimal).

No	Kode	Aktor	Deskripsi Kebutuhan
17	SIDAMAR_F_1700	Marketer	Sistem harus bisa menampilkan seluruh data set iklan yang telah dibuat.
18	SIDAMAR_F_1800	Marketer	Sistem harus menyediakan fitur untuk mengedit set iklan.
18.1	SIDAMAR_F_1801	Marketer	Data set iklan yang dapat diedit adalah nama set iklan, tujuan optimasi, harga setiap 1000, anggaran harian, tanggal mulai set iklan dan tanggal berakhir set iklan.
19	SIDAMAR_F_1900	Marketer	Sistem harus menyediakan fitur untuk melihat detail set iklan.
20	SIDAMAR_F_2000	Marketer	Sistem harus menyediakan fitur untuk menghapus set iklan.
21	SIDAMAR_F_2100	Marketer	Sistem harus menyediakan fitur untuk memulai set iklan.
22	SIDAMAR_F_2200	Marketer	Sistem harus menyediakan fitur untuk menjeda set iklan.
23	SIDAMAR_F_2300	Marketer	Sistem harus menyediakan fitur untuk menambah iklan.
23.1	SIDAMAR_F_2301	Marketer	Pada langkah membuat iklan, sistem harus menyediakan formulir yang berisi nama iklan, pilihan set iklan, dan pilihan <i>ad creative</i> .
24	SIDAMAR_F_2400	Marketer	Sistem harus bisa menampilkan seluruh data iklan yang telah dibuat.
25	SIDAMAR_F_2500	Marketer	Sistem harus menyediakan fitur untuk memulai iklan.
26	SIDAMAR_F_2600	Marketer	Sistem harus menyediakan fitur untuk menjeda iklan.
27	SIDAMAR_F_2700	Marketer	Sistem harus menyediakan fitur untuk mengedit iklan.
27.1	SIDAMAR_F_2701	Marketer	Data iklan yang dapat diedit adalah nama, dan pilihan <i>ad creative</i> .
28	SIDAMAR_F_2800	Marketer	Sistem harus menyediakan fitur untuk menghapus iklan yang telah terdaftar.

No	Kode	Aktor	Deskripsi Kebutuhan
29	SIDAMAR_F_2900	Marketer	Sistem harus menyediakan fitur untuk menambah <i>ad creative</i> .
29.1	SIDAMAR_F_2901	Marketer	Pada langkah membuat <i>ad creative</i> , sistem harus menyediakan formulir yang berisi nama <i>ad creative</i> , pilihan halaman facebook, pilihan pixel facebook, <i>hash image</i> , <i>link url</i> , pilihan <i>call to action</i> dan deskripsi iklan.
30	SIDAMAR_F_3000	Marketer	Sistem harus menyediakan fitur untuk menampilkan data seluruh <i>ad creative</i> yang telah dibuat.
31	SIDAMAR_F_3001	Marketer	Sistem harus menyediakan fitur untuk memotong gambar sesuai dengan ukuran gambar ideal facebook.
32	SIDAMAR_F_3200	Marketer	Sistem harus menyediakan fitur untuk melihat <i>preview ad creative</i> .
32.1	SIDAMAR_F_3201	Marketer	Tipe <i>preview ad creative</i> terdiri dari: kabar facebook, artikel instan facebook, kolom kanan facebook, <i>marketplace</i> facebook, dan <i>stories</i> facebook.
33	SIDAMAR_F_3300	Marketer	Sistem harus bisa mengedit <i>ad creative</i> .
33.1	SIDAMAR_F_3301	Marketer	Data <i>ad creative</i> yang dapat diedit adalah nama <i>ad creative</i> .
34	SIDAMAR_F_3400	Marketer	Sistem harus menyediakan fitur untuk menghapus <i>ad creative</i> yang telah terdaftar.
35	SIDAMAR_F_3500	Marketer	Sistem harus menyediakan fitur untuk menambah <i>ad pixel</i> .
35.1	SIDAMAR_F_3501	Marketer	Pada langkah membuat <i>ad pixel</i> , sistem harus menyediakan formulir yang berisi nama <i>ad pixel</i> .
36	SIDAMAR_F_3600	Marketer	Sistem harus menyediakan fitur untuk menampilkan data seluruh <i>ad pixel</i> yang telah dibuat.

No	Kode	Aktor	Deskripsi Kebutuhan
37	SIDAMAR_F_3700	Marketer	Sistem harus bisa mengedit <i>ad pixel</i> .
37.5	SIDAMAR_F_3701	Marketer	Data <i>ad pixel</i> yang dapat diedit adalah nama <i>ad pixel</i> .
38	SIDAMAR_F_3800	Marketer	Sistem harus menyediakan fitur untuk set halaman website.
38.1	SIDAMAR_F_3801	Marketer	Data set halaman website yang harus diisi adalah formulir harga setiap konversi, persen net konversi, dan halaman website yang dituju.
39	SIDAMAR_F_3900	Marketer	Sistem harus menyediakan fitur untuk mengedit set halaman website.
39.1	SIDAMAR_F_3901	Marketer	Data set halaman website yang dapat diedit adalah formulir harga setiap konversi, persen net konversi, dan halaman website yang dituju.
40	SIDAMAR_F_4000	Marketer	Sistem harus menyediakan fitur untuk <i>generate</i> kode pixel sesuai dengan inputan aktor.
41	SIDAMAR_F_4100	Marketer	Sistem harus menyediakan fitur <i>copy to clipboard</i> untuk kode pixel yang telah <i>di generate</i> .
42	SIDAMAR_F_4200	Marketer	Sistem harus menyediakan fitur untuk mengganti akun iklan.
43	SIDAMAR_F_4300	Marketer	Sistem harus memiliki fungsionalitas untuk melihat sisa anggaran yang tersedia pada akun iklan.
44	SIDAMAR_F_4400	Marketer	Sistem harus bisa menampilkan data semua iklan yang telah dibuat pada menu <i>dashboard</i> .
45	SIDAMAR_F_4500	Marketer	Sistem harus bisa menampilkan data wawasan yang terdiri dari wawasan iklan, wawasan <i>website</i> , wawasan <i>website</i> dalam bentuk grafik, serta wawasan ROAS dan ROMI dalam bentuk grafik untuk setiap iklan.

No	Kode	Aktor	Deskripsi Kebutuhan
45.1	SIDAMAR_F_4501	Marketer	Wawasan iklan yang ditampilkan terdiri dari <i>impression</i> , <i>click</i> , <i>cost</i> , dan <i>ctr</i> .
45.2	SIDAMAR_F_4502	Marketer	Wawasan <i>website</i> yang ditampilkan adalah jumlah konversi, <i>page view</i> , dan jumlah <i>revenue</i> .
45.3	SIDAMAR_F_4503	Marketer	Wawasan <i>website</i> dalam bentuk grafik yang ditampilkan adalah konversi dan <i>page view</i> .
46	SIDAMAR_F_4600	Marketer	Sistem harus menyediakan fitur untuk melihat status iklan terakhir.
47	SIDAMAR_F_4700	Marketer	Sistem harus menyediakan fitur untuk menampilkan semua set iklan pada menu <i>dashboard</i> tab <i>compare</i> .
48	SIDAMAR_F_4800	Marketer	Sistem harus menyediakan fitur untuk melakukan <i>compare</i> otomatis dalam satu set iklan.
48.1	SIDAMAR_F_4801	Marketer	Waktu <i>compare</i> otomatis adalah setiap satu jam.
49	SIDAMAR_F_4900	Marketer	Sistem harus menyediakan fitur untuk melakukan hapus <i>compare</i> dalam satu set iklan.
50	SIDAMAR_F_5000	Marketer	Sistem harus menyediakan fitur untuk melihat log setiap set iklan.
50.1	SIDAMAR_F_5001	Marketer	Ketika melihat log setiap set iklan maka user harus memilih tanggal yang digunakan untuk melihat log.
51	SIDAMAR_F_5100	Marketer	Sistem harus menyediakan fitur notifikasi untuk memberitahu jika ada iklan yang diberhentikan.
51.1	SIDAMAR_F_5101	Marketer	Bentuk notifikasi adalah <i>icon</i> lonceng, angka notifikasi, dan detail (nama iklan, id iklan, hasil <i>ctr</i>).
52	SIDAMAR_F_5200	Pengguna, Marketer	Sistem harus bisa mengeluarkan Aktor Pengguna dan <i>Marketer</i> dari sistem.

4.5 Kebutuhan Non-Fungsional

Merupakan suatu kebutuhan yang bersifat non-fungsional memiliki arti terkait dengan kehandalan yang dimiliki oleh suatu aplikasi, contohnya dalam dunia nyata seperti *air conditioner* harus menyala 1x24 jam pada ATM. Pada Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* ditunjukkan Tabel 4.4 dibawah ini.

Tabel 4.4 Kebutuhan Non Fungsional

No	Kode	Parameter	Deskripsi Kebutuhan
1	SIDAMAR_NF_01	<i>Compatibility</i>	Sistem harus berjalan dengan baik ketika dijalankan di peramban yang berbeda-beda.

4.6 Diagram Use Case

Diagram *usecase* adalah suatu pemodelan UML untuk menunjukkan apa yang bisa dilakukan oleh *actor marketer*, pengunjung dan pengguna terhadap sistem yang ditunjukkan Gambar 4.4. Pada *use case* set halaman website terdapat *include* *men-generate* ad pixel yang dimaksudkan *use case* *men-generate* ad pixel tidak bisa dilakukan ketika *use case* set halaman website belum di lakukan. Dan pada *use case* *men-generate* ad pixel juga terdapat relasi *include* pada *use case* *copy to clipboard* yang berarti *use case* *copy to clipboard* tidak bisa dilakukan jika *use case* *men-generate* ad pixel tidak dilakukan.

Tabel 4.5 Skenario *Use Case Login*

<i>Actor</i>	Pengunjung
<i>Objective</i>	Pengunjung dapat masuk kedalam sistem dengan cara melalui autentikasi <i>login</i> dan menjadi pengguna.
<i>Pre-Condition</i>	Aktor membuka halaman <i>login</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Pengunjung mengetikkan <i>username</i> dan <i>password</i> pada <i>form</i> yang disediakan dengan lengkap. 2. Pengunjung menekan tombol "<i>sign in</i>". 3. Sistem akan memeriksa data <i>username</i> dan <i>password</i> pengunjung pada <i>database</i>. 4. Sistem akan memperlihatkan <i>page</i> pada <i>browser</i>, yaitu halaman pengunjung.
<i>Alternative Flow</i>	Jika ternyata data <i>username</i> atau <i>password</i> tidak ada pada <i>database</i> , maka aplikasi akan memperlihatkan <i>exception</i> berupa pesan <i>error</i> " <i>User/Password</i> tidak cocok".
<i>Post-Condition</i>	Sistem akan mengalihkan halaman <i>login</i> menuju halaman <i>dashboard</i> dengan status <i>role</i> baru yaitu pengguna.

Tabel 4.6 Skenario *Use Case Lupa Password*

<i>Actor</i>	Pengunjung
<i>Objective</i>	Aktor bisa me-reset kata sandi akun yang telah didaftarkan ketika lupa dengan <i>password</i> yang dimilikinya.
<i>Pre-Condition</i>	Aktor membuka halaman <i>login</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tulisan <i>Lupa Password</i>. 2. Sistem mengeluarkan modal <i>lupa password</i>. 3. Aktor mengisikan alamat email yang sebelumnya digunakan untuk mendaftar pada form. 4. Aktor kemudian menekan tombol "<i>Reset Password</i>". 5. Sistem akan mengirimkan <i>link reset password</i> pada alamat email yang telah diisikan dan terdaftar pada sistem.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika form kolom email kosong dan aktor menekan tombol "<i>Reset Password</i>", sistem akan memperlihatkan <i>exception error</i> "<i>Email Tidak Terdaftar!</i>". 2. Saat aktor menekan tombol "<i>X</i>", maka modal <i>lupa password</i> akan ditutup oleh sistem.

	3. Jika aktor memasukkan email yang tidak terdaftar sebelumnya kemudian ditekan tombol “Reset Password”, maka sistem akan memperlihatkan <i>exception error</i> “Email Tidak Terdaftar!”.
<i>Post-Condition</i>	Sistem akan mengirimkan <i>password</i> pada alamat e-mail pengguna dan kemudian akan mengalihkan pada <i>page login</i> dengan menampilkan pesan “Silahkan Cek Email Kamu!”.

Tabel 4.7 Skenario Use Case Pendaftaran

<i>Actor</i>	Pengunjung
<i>Objective</i>	Aktor belum memiliki autentikasi dan ingin mendaftar ke dalam sistem.
<i>Pre-Condition</i>	Aktor membuka halaman <i>login</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tulisan <i>sign up now</i>. 2. Sistem mengalihkan ke <i>page register</i>. 3. Aktor mengetikkan informasi <i>full name</i>, <i>username</i>, <i>password</i>, <i>re-type password</i>, dan e-mail. 4. Aktor menekan tombol “Submit”.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika informasi <i>full name</i>, <i>username</i>, <i>password</i>, <i>re-type password</i>, dan e-mail tidak diisi semua maka sistem akan menampilkan <i>exception error</i> “Please fill out this field”. 2. Jika salah satu dari data <i>full name</i>, <i>username</i>, <i>password</i>, <i>re-type password</i>, dan email tidak diisi maka akan memperlihatkan <i>exception error</i>. 3. Jika data <i>password</i> kurang dari enam karakter maka akan memperlihatkan <i>exception error</i>. 4. Jika data <i>password</i> dan data <i>re-type password</i> tidak sama maka akan memperlihatkan <i>exception error</i>.
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page login</i> dan mengeluarkan pesan “Sukses Terdaftar”.

Tabel 4.8 Skenario Use Case Melihat Preface

<i>Actor</i>	Pengguna
<i>Objective</i>	Aktor sudah memiliki autentikasi dan dapat masuk ke dalam sistem.
<i>Pre-Condition</i>	Aktor sudah berhasil teridentifikasi aktor kemudian sudah masuk ke sistem.



<i>Main Flow</i>	Sistem mengalihkan ke <i>page</i> pengunjung.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page</i> pengunjung.

Tabel 4.9 Skenario Use Case Melihat 3 Komponen Iklan Facebook

<i>Actor</i>	Pengguna
<i>Objective</i>	Aktor sudah memiliki autentikasi dan dapat masuk ke dalam sistem.
<i>Pre-Condition</i>	Aktor telah berhasil teridentifikasi sebagai aktor dan masuk sebagai <i>role</i> pengguna.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan pilihan menu <i>3 part of fb ads</i>. 2. Sistem menampilkan menu kampanye, set iklan, dan iklan. 3. Aktor menekan salah satu dari kampanye, set iklan, dan iklan. 4. Sistem mengalihkan ke <i>page</i> sesuai dengan pilihan aktor.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page</i> sesuai dengan pilihan aktor.

Tabel 4.10 Skenario Use Case Melihat ROAS dan ROMI

<i>Actor</i>	Pengguna
<i>Objective</i>	Aktor sudah memiliki autentikasi dan <i>role</i> telah teridentifikasi sebagai pengguna.
<i>Pre-Condition</i>	Aktor telah berhasil teridentifikasi sebagai aktor dan masuk ke sistem.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tulisan <i>introduce ROAS &ROMI</i>. 2. Sistem mengalihkan ke <i>page measure</i>.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page measure</i> .

Tabel 4.11 Skenario Use Case Melihat About SIDAMAR

<i>Actor</i>	Pengguna
<i>Objective</i>	Aktor sudah memiliki autentikasi dan memiliki status <i>role</i> sebagai pengguna.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai aktor dan masuk ke sistem.
<i>Main Flow</i>	1. Aktor menekan <i>about</i> SIDAMAR. 2. Sistem mengalihkan ke <i>page about</i> .
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page about</i> .

Tabel 4.12 Skenario Use Case Melihat Dokumentasi

<i>Actor</i>	Pengguna
<i>Objective</i>	Aktor sudah memiliki autentikasi dan memiliki status <i>role</i> sebaga pengguna.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	1. Aktor menekan menu <i>Complete Documentation</i> . 2. Sistem mengalihkan ke <i>page</i> dokumentasi.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page</i> dokumentasi.

Tabel 4.13 Skenario Use Case Login Facebook

<i>Actor</i>	Pengguna
<i>Objective</i>	Aktor agar memiliki hak akses <i>marketer</i> sesuai peran akun iklan facebook dengan cara melalui verifikasi <i>login</i> facebook.
<i>Pre-Condition</i>	Aktor teridentifikasi dan masuk ke sistem sebagai <i>role</i> pengguna.
<i>Main Flow</i>	1. Aktor menekan “Masuk Dengan Facebook”. 2. Sistem memanggil api facebook <i>login</i> dan mengalihkan ke <i>page</i> masuk facebook. 3. Aktor mengisikan <i>username</i> akun serta <i>password</i> facebooknya. 4. Sistem memperlihatkan page verifikasi perizinan untuk menggunakan data yang diminta.

	<p>5. Aktor menekan "Izinkan".</p> <p>6. Sistem akan mengalihkan ke page <i>dashboard</i> dengan <i>role</i> sebagai <i>marketer</i>.</p>
<i>Alternative Flow</i>	Ketika salah dalam memasukkan <i>username</i> atau <i>password</i> maka akan mengeluarkan <i>exception error</i> pada halaman masuk facebook.
<i>Post-Condition</i>	Sistem akan mengalihkan ke page <i>dashboard</i> dengan <i>role</i> sebagai <i>marketer</i> .

Tabel 4.14 Skenario Use Case Menambah Kampanye

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat membuat kampanye berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan pilihan menu kampanye. 3. Sistem mengalihkan ke <i>page</i> kampanye. 4. Aktor menekan tombol "Tambah Kampanye". 5. Sistem mengeluarkan modal tambah kampanye. 6. Aktor mengetikkan nama kampanye serta memilih data tujuan beriklan. 7. Aktor menekan "Save Campaign". 8. Sistem menampilkan <i>alert</i> "Buat Kampanye Berhasil". 9. Aktor menekan "OK".
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika menekan "close" maka modal akan tertutup dan batal tambah kampanye. 2. Ketika aktor menekan "X" mengakibatkan modal tertutup dan batal tambah kampanye. 3. Jika aktor membiarkan form nama kampanye kosong kemudian menekan tombol "save campaign" maka memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
<i>Post-Condition</i>	Sistem memperlihatkan kampanye terbaru yang telah ditambahkan.



Tabel 4.15 Skenario Use Case Melihat Kampanye

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat keseluruhan kampanye berdasarkan tiap <i>account</i> iklan
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu kampanye. 3. Sistem mengalihkan ke <i>page</i> kampanye. 4. Sistem menampilkan seluruh kampanye berdasarkan pilihan akun iklan pada tabel.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan seluruh kampanye pada tabel berdasarkan pilihan akun iklan.

Tabel 4.16 Skenario Use Case Mengedit Kampanye

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengedit kampanye berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu kampanye. 3. Sistem mengalihkan ke <i>page</i> kampanye. 4. Sistem menampilkan seluruh kampanye berdasarkan pilihan akun iklan dan ditampilkan pada tabel. 5. Aktor menekan tombol edit pada kampanye yang ingin diedit. 6. Sistem menampilkan modal edit kampanye. 7. Aktor merubah data nama kampanye atau data pilihan tujuan beriklan. 8. Aktor menekan tombol "<i>Save changes</i>". 9. Sistem memperlihatkan <i>alert</i> "Edit Kampanye Berhasil". 10. Aktor menekan tombol "OK".
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol "<i>close</i>" maka modal tertutup dan batal edit kampanye. 2. Jika aktor mengosongkan data form nama kampanye, maka sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".



<i>Post-Condition</i>	Sistem memperlihatkan data kampanye terbaru pada tabel.
-----------------------	---

Tabel 4.17 Skenario Use Case Menghapus Kampanye

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menghapus kampanye berdasarkan tiap <i>account</i> iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu kampanye. 3. Sistem mengalihkan ke <i>page</i> kampanye. 4. Sistem menampilkan seluruh kampanye berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol "<i>delete</i>" untuk kampanye yang ingin dihapus. 6. Sistem memperlihatkan <i>alert</i> "Delete Kampanye". 7. Aktor menekan tombol "Ya, Tentu!".
<i>Alternative Flow</i>	Ketika aktor menekan tombol "Tidak Jadi" saat sistem memperlihatkan <i>alert</i> , maka kampanye tidak jadi dihapus.
<i>Post-Condition</i>	Kampanye dihapus sehingga tidak tampil kembali pada tabel.

Tabel 4.18 Skenario Use Case Memulai Kampanye

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengaktifkan kampanye berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu kampanye. 3. Sistem mengalihkan ke <i>page</i> kampanye. 4. Sistem menampilkan seluruh kampanye berdasarkan pilihan akun iklan dan ditampilkan pada tabel. 5. Aktor menekan tombol "Mulai Kampanye".
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Status kampanye menjadi <i>active</i> .

Tabel 4.19 Skenario *Use Case* Menjeda Kampanye

<i>Actor</i>	Marketer
<i>Objective</i>	Mengizinkan aktor untuk menjeda kampanye berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan pilihan kampanye pada menu. 3. Sistem mengalihkan ke <i>page</i> kampanye. 4. Sistem menunjukkan seluruh data kampanye berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol "Jeda Kampanye". 6. Sistem memperlihatkan <i>alert</i> "Pause Kampanye". 7. Aktor menekan tombol "Ya, Tentu!".
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Status kampanye menjadi <i>paused</i> .

Tabel 4.20 Skenario *Use Case* Menambah Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat membuat set iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu set iklan. 3. Sistem memperlihatkan <i>page</i> set iklan. 4. Aktor menekan tombol "Tambah Set Iklan". 5. Sistem menampilkan modal tambah set iklan. 6. Aktor mengisi data nama set iklan, memilih kampanye, memilih tujuan optimasi, menentukan biaya impresi, menentukan anggaran harian, menentukan tanggal mulai set iklan, menentukan tanggal berakhir iklan, dan mengisi data targeting set iklan. 7. Aktor menekan tombol "Save adset". 8. Sistem menampilkan <i>alert</i> "Buat Set Iklan Berhasil". 9. Aktor menekan tombol "OK".
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol "close" maka modal tertutup dan batal tambah set iklan. 2. Jika aktor membiarkan form nama set iklan kosong, dan tidak memilih tujuan optimasi maka sistem

	memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
<i>Post-Condition</i>	Sistem memperlihatkan set iklan terbaru yang telah ditambahkan.

Tabel 4.21 Skenario Use Case Melihat Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat data seluruh set iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan set iklan pada menu. 3. Sistem memperlihatkan <i>page</i> setiklan. 4. Sistem menampilkan seluruh set iklan berdasarkan pilihan akun iklan pada tabel.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan seluruh set iklan pada tabel berdasarkan pilihan akun iklan.

Tabel 4.22 Skenario Use Case Mengedit Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengedit set iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu set iklan. 3. Sistem memperlihatkan <i>page</i> setiklan. 4. Sistem menampilkan seluruh set iklan berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol "edit" set iklan yang ingin diedit. 6. Sistem menampilkan modal edit set iklan. 7. Aktor merubah data nama set iklan, tujuan optimasi, biaya impresi, anggaran harian, mulai iklan, dan akhir iklan. 8. Aktor menekan tombol "Save changes". 9. Sistem memperlihatkan <i>alert</i> "Edit Set Iklan Berhasil". 10. Aktor menekan tombol "OK".



<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika aktor menekan tombol “close” maka modal akan tertutup dan batal edit set iklan. 2. Jika aktor mengosongkan data form nama set iklan, tujuan optimasi, biaya impresi, anggaran harian, mulai iklan, dan akhir iklan maka sistem akan menampilkan pesan error “Wajib diisi, tidak boleh kosong”.
<i>Post-Condition</i>	Sistem memperlihatkan set iklan pada tabel sesuai data yang telah dirubah.

Tabel 4.23 Skenario Use Case Melihat Detail Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat bagaimana detail set iklan berdasarkan tiap set iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu set iklan. 3. Sistem memperlihatkan <i>page</i> set iklan. 4. Sistem menampilkan seluruh set iklan berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol “detail set iklan” pada set iklan. 6. Sistem menampilkan modal “detail set iklan”.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan modal detail set iklan.

Tabel 4.24 Skenario Use Case Menghapus Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menghapus set iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu set iklan. 3. Sistem memperlihatkan <i>page</i> set iklan. 4. Sistem menampilkan seluruh set iklan berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol “Delete” pada set iklan yang ingin dihapus. 6. Sistem memperlihatkan <i>alert</i> “Delete Set Iklan”. 7. Aktor menekan tombol “Ya, Tentu!”.



<i>Alternative Flow</i>	Ketika aktor menekan tombol “Tidak Jadi” saat sistem memperlihatkan <i>alert</i> , maka set iklan tidak jadi dihapus.
<i>Post-Condition</i>	Kampanye dihapus kemudian sudah tidak ada pada tabel.

Tabel 4.25 Skenario *Use Case* Memulai Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengaktifkan set iklan berdasarkan set iklan yang diinginkan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu set iklan. 3. Sistem memperlihatkan page set iklan. 4. Sistem menampilkan seluruh set iklan berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol “Mulai set iklan”.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Status set iklan menjadi <i>active</i> .

Tabel 4.26 Skenario *Use Case* Menjeda Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menjeda set iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu set iklan. 3. Sistem memperlihatkan <i>page</i> set iklan. 4. Sistem menampilkan seluruh set iklan berdasarkan pilihan akun iklan pada tabel. 5. Aktor menekan tombol “Jeda set iklan”. 6. Sistem memperlihatkan <i>alert</i> “Pause Set Iklan”. 7. Aktor menekan tombol “Ya, Tentu!”.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Status set iklan menjadi <i>paused</i> .



Tabel 4.27 Skenario Use Case Menambah Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menambahkan iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai marketer.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu iklan. 3. Sistem mengalihkan ke <i>page</i> iklan. 4. Aktor menekan tombol "Tambah Iklan". 5. Sistem memperlihatkan modal tambah iklan. 6. Aktor mengetikkan data nama iklan, memilih set iklan, dan memilih ad creative. 7. Aktor menekan tombol "Save Ad". 8. Sistem memperlihatkan <i>alert</i> "Buat Iklan Berhasil". 9. Aktor menekan tombol "OK".
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol "Close" mengakibatkan modal "tambah iklan" tertutup dan batal tambah iklan. 2. Jika aktor membiarkan <i>form</i> nama iklan kosong kemudian menekan tombol "save ad" maka sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
<i>Post-Condition</i>	Sistem akan memperlihatkan iklan terbaru yang telah ditambahkan.

Tabel 4.28 Skenario Use Case Melihat Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat seluruh iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Aktor teridentifikasi sebagai <i>marketer</i>. 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol iklan. 2. Sistem memperlihatkan semua iklan yang telah ditambahkan.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan seluruh iklan pada tabel berdasarkan pilihan akun iklan.



Tabel 4.29 Skenario *Use Case* Memulai Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengaktifkan iklan berdasarkan iklan yang diinginkan.
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Aktor teridentifikasi sebagai <i>marketer</i>. 2. Aktor memilih akun iklan yang diinginkan. 3. Aktor telah membuka halaman iklan dengan menekan menu iklan. 3. Status iklan <i>paused</i>.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol “mulai iklan”. 2. Sistem menjalankan iklan dan mengubah status menjadi <i>active</i>.
<i>Alternative Flow</i>	Jika status iklan <i>active</i> maka tombol “mulai iklan” tidak tersedia.
<i>Post-Condition</i>	Sistem akan menjalankan iklan dan status iklan menjadi <i>active</i> .

Tabel 4.30 Skenario *Use Case* Menjeda Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor bisa menjeda iklan berdasarkan iklan yang diinginkan.
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Aktor berhasil teridentifikasi sebagai <i>marketer</i>. 2. Aktor memilih akun iklan yang diinginkan. 3. Aktor telah membuka halaman iklan dengan menekan menu iklan. 4. Status iklan <i>active</i>.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol “jeda iklan”. 2. Sistem memperlihatkan <i>alert</i> “PAUSE IKLAN”. 3. Aktor menekan tombol “Ya, Tentu!”. 4. Sistem akan menjeda iklan dan mengubah status menjadi <i>paused</i>.
<i>Alternative Flow</i>	Jika status iklan <i>paused</i> maka tombol “jeda iklan” tidak tersedia.
<i>Post-Condition</i>	Sistem akan menjeda iklan dengan mengubah status menjadi <i>paused</i> .

Tabel 4.31 Skenario *Use Case* Mengedit Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor bisa mengedit iklan yang telah tersimpan.
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Aktor teridentifikasi sebagai <i>marketer</i>. 2. Aktor memilih akun iklan yang diinginkan. 3. Aktor telah membuka halaman iklan dengan menekan menu iklan.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol "Edit Iklan". 2. Sistem menampilkan modal edit iklan. 3. Aktor mengedit pada <i>form</i> iklan. 4. Aktor menekan tombol "Save changes". 5. Sistem akan menyimpan iklan yang telah diubah dan menampilkan semua iklan pada halaman iklan.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika aktor mengosongkan salah satu <i>form</i> pada modal "edit iklan" maka akan muncul pesan <i>error</i> "Wajib diisi, tidak boleh kosong". 2. Jika aktor menekan tombol "close" pada modal edit iklan maka modal akan tertutup dan batal edit iklan.
<i>Post-Condition</i>	Sistem menyimpan data iklan yang telah diubah dan menampilkan semua iklan pada halaman iklan.

Tabel 4.32 Skenario *Use Case* Menghapus Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor bisa menghapus iklan pada data iklan yang berada di halaman iklan.
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Aktor teridentifikasi sebagai <i>marketer</i>. 2. Aktor memilih akun iklan yang diinginkan. 3. Aktor telah membuka halaman iklan dengan menekan menu iklan.
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol "Delete". 2. Sistem akan memperlihatkan <i>alert</i> "Delete Iklan". 3. Aktor menekan tombol "Ya, Tentu!". 4. Sistem akan menghapus iklan dan kemudian memperlihatkan <i>page</i> iklan.

<i>Alternative Flow</i>	Jika aktor menekan tombol “Tidak Jadi” saat sistem menampilkan <i>alert</i> , maka iklan tidak jadi dihapus.
<i>Post-Condition</i>	Sistem akan menghapus iklan dan kemudian memperlihatkan <i>page</i> iklan.

Tabel 4.33 Skenario Use Case Menambah Ad Creative

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat membuat set iklan berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu “<i>ad creative</i>”. 3. Sistem memperlihatkan <i>page adcreative</i>. 4. Aktor menekan tombol “Tambah Adcreative”. 5. Sistem memperlihatkan <i>form</i> tambah <i>ad creative</i>. 6. Aktor mengisi data <i>creative name</i>, memilih halaman facebook, memilih ad pixel, mengisi data <i>hash image</i>, mengisi data <i>link url</i>, memilih <i>call to action</i>, mengisi data deskripsi. 7. Aktor menekan tombol “<i>Save Creative</i>”. 8. Sistem menampilkan <i>alert</i> “Buat Iklan Kreatif Berhasil”. 9. Aktor menekan tombol “OK”.
<i>Alternative Flow</i>	Ketika aktor membiarkan <i>form</i> nama iklan kosong kemudian menekan tombol “ <i>Save Creative</i> ” maka sistem akan menunjukkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
<i>Post-Condition</i>	Sistem memperlihatkan <i>ad creative</i> terbaru yang telah ditambahkan pada tabel.

Tabel 4.34 Skenario Use Case Melihat Ad Creative

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat seluruh data <i>ad creative</i> berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu <i>ad creative</i> 3. Sistem memperlihatkan <i>page adcreative</i>. 4. Sistem memperlihatkan seluruh data <i>ad creative</i>.



<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem menampilkan seluruh <i>ad creative</i> pada tabel berdasarkan pilihan akun iklan.

Tabel 4.35 Skenario Use Case Memotong Gambar

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat memotong gambar sesuai yang direkomendasikan facebook.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu <i>ad creative</i>. 3. Sistem memperlihatkan <i>page adcreative</i>. 4. Aktor menekan tombol "<i>Crop Image</i>". 5. Sistem memperlihatkan modal potong gambar. 6. Aktor menekan tombol "<i>browse</i>" untuk memilih gambar yang akan dipotong. 7. Aktor menyeret foto sesuai ukuran maksimal kemudian menekan tombol "<i>Crop</i>". 8. Aktor menekan tombol "<i>Download</i>".
<i>Alternative Flow</i>	Ketika aktor menekan tombol "Close" di modal potong gambar mengakibatkan modal tertutup dan batal potong gambar.
<i>Post-Condition</i>	Sistem membuka <i>tab browser</i> baru berisi gambar setelah di <i>crop</i> .

Tabel 4.36 Skenario Use Case Melihat Preview Ad Creative

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat <i>preview</i> pada <i>ad creative</i> .
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu "<i>Ad Creative</i>". 3. Sistem memperlihatkan <i>page adcreative</i>. 4. Sistem menampilkan seluruh <i>ad creative</i>. 5. Aktor menekan tombol "<i>Preview</i>". 6. Sistem menampilkan modal <i>preview ad creative</i>. 7. Aktor memilih tipe <i>ad creative</i> yang ingin di <i>preview</i>. 8. Sistem menampilkan <i>preview</i> sesuai dengan tipe <i>ad creative</i> yang dipilih.

<i>Alternative Flow</i>	Ketika aktor menekan tombol “Close” pada modal <i>preview ad creative</i> maka modal akan tertutup dan batal <i>preview ad creative</i> .
<i>Post-Condition</i>	Sistem memperlihatkan <i>preview</i> yang sesuai dengan tipe <i>ad creative</i> yang dipilih oleh aktor.

Tabel 4.37 Skenario Use Case Mengedit Ad Creative

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengedit <i>ad creative</i> berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu “Ad Creative”. 3. Sistem memperlihatkan <i>page adcreative</i>. 4. Sistem menampilkan seluruh <i>ad creative</i>. 5. Aktor menekan tombol icon “/” pada <i>ad creative</i> yang ingin diedit. 6. Sistem menampilkan modal edit <i>ad creative</i>. 7. Aktor merubah data nama <i>ad creative</i>. 8. Aktor menekan tombol “Save changes”. 9. Sistem memperlihatkan <i>alert</i> “Edit Ad Creative Berhasil”. 10. Aktor menekan tombol “OK”.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol “close” maka modal akan tertutup dan batal edit <i>ad creative</i>. 2. Jika aktor mengosongkan data form nama <i>ad creative</i> maka sistem akan menampilkan pesan <i>error</i> “Wajib diisi, tidak boleh kosong”.
<i>Post-Condition</i>	Sistem menampilkan <i>ad creative</i> pada tabel sesuai data yang telah dirubah.

Tabel 4.38 Skenario Use Case Menghapus Ad Creative

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor bisa menghapus <i>ad creative</i> berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu “Ad Creative”.

	<ol style="list-style-type: none"> 3. Sistem memperlihatkan page <i>adcreative</i>. 4. Sistem menampilkan seluruh <i>ad creative</i>. 5. Aktor menekan tombol icon “X” pada <i>ad creative</i> yang ingin dihapus. 6. Sistem menampilkan <i>alert “Delete Ad Creative”</i>. 7. Aktor menekan tombol “Ya, Tentu”.
<i>Alternative Flow</i>	Ketika aktor menekan tombol “Tidak Jadi” saat sistem memperlihatkan <i>alert</i> , maka <i>ad creative</i> tidak jadi dihapus.
<i>Post-Condition</i>	Ad creative berhasil dihapus dan tidak ada pada tabel.

Tabel 4.39 Skenario Use Case Menambah Ad Pixel

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat membuat ad pixel berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu “Ad Pixel”. 3. Sistem memperlihatkan page <i>adpixel</i>. 4. Aktor menekan tombol “Tambah Ad Pixel”. 5. Sistem menampilkan modal tambah ad pixel. 6. Aktor mengisi data nama ad pixel. 7. Aktor menekan tombol “Save Ad Pixel”. 8. Sistem menampilkan <i>alert “Tambah Ad Pixel Berhasil”</i>. 9. Aktor menekan tombol “OK”.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol “close” mengakibatkan modal tambah Ad Pixel tertutup dan batal tambah Ad Pixel. 2. Jika aktor membiarkan form nama Ad Pixel kosong kemudian menekan tombol “save ad” maka sistem menunjukkan <i>exception error “Wajib diisi, tidak boleh kosong”</i>. 3. Jika tipe akun iklan yang dipilih merupakan akun pribadi maka terdapat <i>alert “Akun Iklan Pribadi Hanya 1 Pixel”</i>.
<i>Post-Condition</i>	Sistem akan menampilkan ad pixel terbaru yang telah ditambahkan.

Tabel 4.40 Skenario Use Case Melihat Ad Pixel

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat seluruh data ad pixel berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu "Ad Pixel". 3. Sistem memperlihatkan <i>page</i> adpixel. 4. Sistem memperlihatkan seluruh ad pixel
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem menampilkan seluruh ad pixel pada tabel berdasarkan pilihan akun iklan.

Tabel 4.41 Skenario Use Case Mengedit Ad Pixel

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat untuk mengedit ad pixel berdasarkan tiap akun iklan.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu "Ad Pixel". 3. Sistem memperlihatkan <i>page</i> adpixel. 4. Sistem menampilkan seluruh ad pixel berdasarkan pilihan akun iklan dan ditampilkan pada tabel. 5. Aktor menekan tombol "Edit" pada ad pixel yang ingin diedit. 6. Sistem menampilkan modal edit ad pixel. 7. Aktor merubah data nama ad pixel. 8. Aktor menekan tombol "Save changes". 9. Sistem memperlihatkan <i>alert</i> "Edit Pixel Berhasil". 10. Aktor menekan tombol "OK".
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol "<i>close</i>" mengakibatkan modal tertutup dan batal edit ad pixel. 2. Jika aktor mengosongkan data form nama ad pixel maka sistem akan menunjukkan <i>exception</i> error "Wajib diisi, tidak boleh kosong".
<i>Post-Condition</i>	Sistem menampilkan ad pixel pada tabel sesuai data yang telah dirubah.

Tabel 4.42 Skenario Use Case Set Halaman Website

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menset halaman website setiap ad pixel.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu "Ad Pixel". 3. Sistem memperlihatkan page adpixel. 4. Sistem menampilkan seluruh ad pixel berdasarkan pilihan akun iklan dan ditampilkan pada tabel. 5. Aktor menekan tombol "Set Halaman" pada ad pixel yang ingin diset halamannya. 6. Sistem menampilkan modal set halaman. 7. Aktor mengisi data harga konversi, persen net konversi, dan halaman konversi. 8. Aktor menekan tombol "Set Halaman". 9. Sistem memperlihatkan <i>alert</i> "Halaman Sudah Diset". 10. Aktor menekan tombol "OK".
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol "close" mengakibatkan modal set halaman tertutup dan batal set website. 2. Jika aktor membiarkan salah satu form kosong kemudian menekan tombol "Set Halaman" maka sistem akan menunjukkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
<i>Post-Condition</i>	Sistem memperlihatkan tombol "Generate Pixel" dan button "Edit Halaman".

Tabel 4.43 Skenario Use Case Mengedit Halaman Website

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengedit Halaman <i>Website</i> setiap ad pixel.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i>
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih akun iklan yang diinginkan. 2. Aktor menekan menu "Ad Pixel". 3. Sistem memperlihatkan <i>page</i> adpixel. 4. Sistem menampilkan seluruh ad pixel berdasarkan pilihan akun iklan dan ditampilkan pada tabel. 5. Aktor menekan tombol "Edit Halaman" pada ad pixel yang ingin edit halamannya.



	<p>6. Sistem menampilkan modal edit halaman.</p> <p>7. Aktor mengedit data keuntungan konversi, persen net konversi, dan halaman konversi.</p> <p>8. Aktor menekan tombol “Save changes”.</p> <p>9. Sistem memperlihatkan alert “Halaman Berhasil Dirubah”.</p> <p>10. Aktor menekan tombol “OK”.</p>
<i>Alternative Flow</i>	<p>1. Ketika aktor menekan tombol “close” maka modal akan tertutup dan batal edit halaman.</p> <p>2. Jika aktor membiarkan salah satu form kosong kemudian menekan tombol “Save changes” maka sistem akan menunjukkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.</p>
<i>Post-Condition</i>	Sistem memperlihatkan perubahan <i>page</i> dan nilai pada tabel sesuai data yang telah dirubah.

Tabel 4.44 Skenario Use Case Menggenerate Ad Pixel

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat <i>menggenerate</i> ad pixel setiap ad pixel.
<i>Pre-Condition</i>	<p>1. Aktor teridentifikasi sebagai <i>marketer</i></p> <p>2. Aktor memilih akun iklan yang diinginkan.</p> <p>3. Aktor menekan menu “Ad Pixel”.</p> <p>4. Sistem memperlihatkan <i>page</i> ad pixel.</p> <p>5. Aktor telah menekan tombol “Set Halaman”.</p>
<i>Main Flow</i>	<p>1. Aktor menekan tombol “Generate Pixel” pada ad pixel yang ingin di <i>generate</i> kode pixelnya.</p> <p>2. Sistem menampilkan modal <i>generate ad pixel code</i>.</p> <p>3. Sistem <i>menggenerate</i> dan menampilkan kode pixel sesuai dengan set halaman.</p>
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem <i>menggenerate</i> dan memperlihatkan kode pixel sesuai dengan set halaman.

Tabel 4.45 Skenario Use Case Copy to Clipboard

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menyalin kode ad pixel di <i>clipboard</i> .
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i>



	<ol style="list-style-type: none"> 2. Aktor memilih akun iklan yang diinginkan. 3. Aktor menekan menu "Ad Pixel". 4. Sistem memperlihatkan page ad pixel. 5. Aktor telah menekan tombol "Set Halaman".
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Ketika aktor menekan tombol "Generate Pixel" pada ad pixel yang ingin di <i>generate</i> kode pixelnya. 2. Sistem menampilkan modal <i>generate</i> ad pixel <i>code</i>. 3. Sistem <i>menggenerate</i> dan menampilkan kode pixel sesuai dengan set halaman. 4. Aktor menekan logo <i>copy</i> pada pojok kanan atas.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem berhasil menyalin seluruh kode pixel pada <i>clipboard</i> .

Tabel 4.46 Skenario Use Case Mengganti Akun Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat mengganti akun iklan lain yang dimilikinya.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> .
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Sistem menampilkan semua akun iklan yang dimiliki aktor. 2. Aktor memilih akun iklan yang ingin digunakan. 3. Sistem menampilkan hak akses untuk akun iklan diatas pilihan akun iklan.
<i>Alternative Flow</i>	Jika aktor tidak memiliki akun iklan maka aktor bisa menekan pilihan buat akun iklan pribadi untuk membuat akun iklan.
<i>Post-Condition</i>	Sistem menampilkan hak akses untuk akun iklan diatas pilihan akun iklan.

Tabel 4.47 Skenario Use Case Melihat Sisa Anggaran

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat sisa uang anggaran yang dimilikinya.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i>
<i>Main Flow</i>	<ol style="list-style-type: none"> 1. Sistem menampilkan semua akun iklan yang dimiliki aktor. 2. Aktor memilih akun iklan yang ingin digunakan.

	3. Sistem menampilkan sisa uang anggaran.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Sistem memperlihatkan sisa uang anggaran.

Tabel 4.48 Skenario Use Case Melihat Iklan Pada Dashboard

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat iklan pada menu <i>dashboard</i> .
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan tombol menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab iklan. 4. Sistem menampilkan semua data iklan yang telah ditambahkan.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan semua data iklan yang telah ditambahkan.

Tabel 4.49 Skenario Use Case Melihat Grafik Wawasan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat grafik dan statistik yang berjalan pada iklan aktif beserta halaman <i>website</i> yang telah di set pada iklan yang akan di <i>select</i> .
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan tombol menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab iklan. 4. Sistem menampilkan semua data iklan yang telah ditambahkan. 5. Aktor menekan tombol " <i>Lihat Grafik</i> ". 6. Sistem memperlihatkan grafik dan statistik dari iklan yang berjalan serta wawasan halaman <i>website</i> yang telah di set pada iklan yang dipilih.
<i>Alternative Flow</i>	-

<i>Post-Condition</i>	Sistem memperlihatkan grafik beserta statistik dari iklan yang berjalan serta wawasan halaman website yang telah di set pada iklan yang dipilih.
-----------------------	--

Tabel 4.50 Skenario Use Case Melihat Status Terakhir

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat status terakhir iklan yang berjalan.
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan tombol menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab iklan. 4. Sistem menampilkan semua data iklan yang telah ditambahkan. 5. Aktor menekan tombol "Lihat Status Akhir". 6. Sistem memperlihatkan modal status iklan saat ini.
<i>Alternative Flow</i>	Ketika aktor menekan tombol " <i>close</i> " maka modal menjadi tertutup.
<i>Post-Condition</i>	Sistem menampilkan modal status iklan saat ini.

Tabel 4.51 Skenario Use Case Melihat Set Iklan Pada Dashboard

<i>Actor</i>	Marketer
<i>Objective</i>	Mengizinkan aktor untuk melihat daftar set iklan yang berada pada menu <i>dashboard</i> .
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan tombol menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab "Bandingkan". 4. Sistem menampilkan semua set iklan yang telah ditambahkan.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan semua data set iklan yang telah ditambahkan.



Tabel 4.52 Skenario Use Case Compare Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melakukan <i>compare</i> otomatis untuk setiap iklan dalam satu set iklan.
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab " <i>Bandingkan</i> ". 4. Sistem menampilkan semua set iklan yang telah ditambahkan. 5. Aktor menekan tombol " <i>Compare</i> ". 6. Sistem memperlihatkan modal pengaturan <i>compare</i> . 7. Aktor memasukkan data <i>compare</i> mulai cek impresi, Jumlah iklan diambil, dan set CTR ideal. 6. Sistem akan melakukan <i>compare</i> otomatis setiap 1 jam dan kolom status ditambahkan dengan status <i>compared</i> .
<i>Alternative Flow</i>	Jika status set iklan <i>compared</i> maka button " <i>Compare</i> " tidak tersedia.
<i>Post-Condition</i>	Sistem akan menambahkan kolom status dengan status <i>compared</i> .

Tabel 4.53 Skenario Use Case Hapus Compare Set Iklan

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat menghapus <i>compare</i> set iklan.
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab " <i>Bandingkan</i> ". 4. Sistem menampilkan semua data set iklan yang telah ditambahkan. 5. Aktor menekan tombol " <i>Hapus Compare</i> ". 6. Sistem akan menghentikan <i>compare</i> otomatis setiap 1 jam dan kolom status <i>compared</i> akan terhapus.
<i>Alternative Flow</i>	Jika status set iklan tidak terdapat <i>compared</i> maka tombol " <i>Hapus Compare</i> " tidak tersedia.
<i>Post-Condition</i>	Sistem akan menghapus kolom status <i>compared</i> .

Tabel 4.54 Skenario *Use Case* Melihat Log

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor dapat melihat log untuk setiap set iklan ketika status set iklan <i>dashboard</i> terdapat <i>compared</i> .
<i>Pre-Condition</i>	1. Aktor teridentifikasi sebagai <i>marketer</i> 2. Aktor memilih akun iklan yang diinginkan.
<i>Main Flow</i>	1. Aktor menekan tombol menu " <i>Dashboard</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Aktor menekan tab " <i>Bandingkan</i> ". 4. Sistem menampilkan semua data set iklan yang telah ditambahkan. 5. Aktor menekan tombol " <i>Lihat Log</i> ". 6. Sistem akan menampilkan modal pilih tanggal log. 7. Aktor memilih tanggal log yang ingin dilihat. 8. Aktor menekan tombol " <i>Get Log</i> " untuk mengetahui log. 9. Sistem akan menampilkan log berdasarkan set iklan.
<i>Alternative Flow</i>	1. Jika aktor menekan tombol " <i>Close</i> " maka modal menjadi tertutup.
<i>Post-Condition</i>	Sistem akan menampilkan log berdasarkan set iklan

Tabel 4.55 Skenario *Use Case* Melihat Notifikasi

<i>Actor</i>	Marketer
<i>Objective</i>	Aktor bisa melihat notifikasi jika terdapat iklan yang di jeda secara otomatis.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i>
<i>Main Flow</i>	1. Aktor menekan tombol " <i>Masuk Dengan facebook</i> ". 2. Sistem memperlihatkan <i>page dashboard</i> . 3. Sistem menampilkan notifikasi pada <i>icon</i> lonceng jika terdapat iklan yang dijeda secara otomatis.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem memperlihatkan notifikasi pada <i>icon</i> lonceng jika terdapat iklan yang dijeda secara otomatis.

Tabel 4.56 Skenario *Use Case Logout*

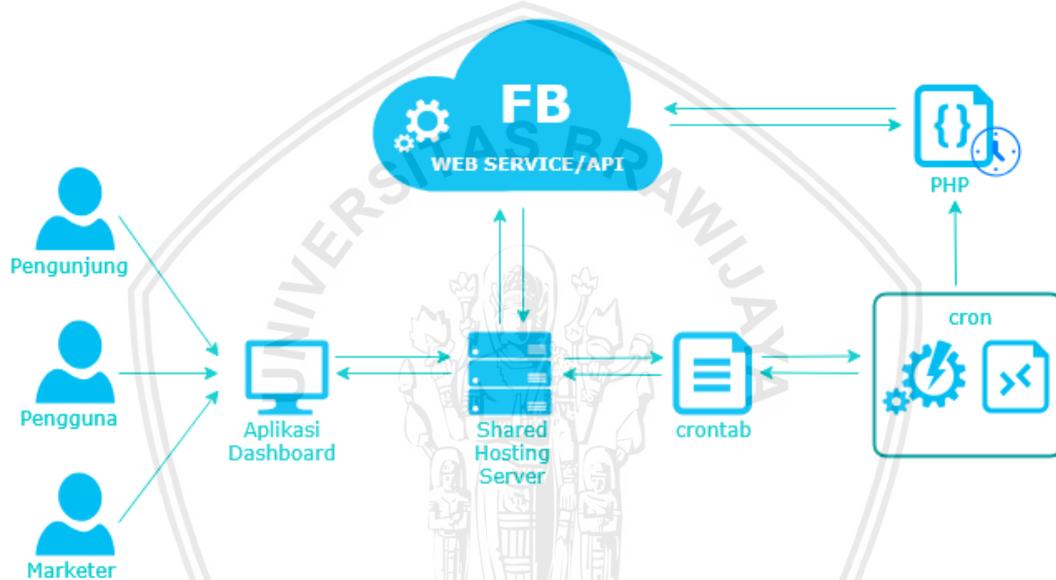
<i>Actor</i>	Pengguna dan Marketer
<i>Objective</i>	Aktor dapat keluar dari sistem kemudian data hak akses aktor akan dihapus dari sistem.
<i>Pre-Condition</i>	Aktor teridentifikasi sebagai <i>marketer</i> atau pengguna
<i>Main Flow</i>	<ol style="list-style-type: none">1. Aktor menekan tombol "<i>Logout</i>".2. Sistem menghapus hak akses aktor.3. Sistem mengalihkan ke <i>page login</i>.
<i>Alternative Flow</i>	-
<i>Post-Condition</i>	Sistem mengalihkan ke <i>page login</i> .



BAB 5 PERANCANGAN DAN IMPLEMENTASI

5.1 Perancangan

Pada subbab ini akan membahas tentang perancangan dan implementasi dari aplikasi *dashboard* untuk analisa efektivitas *marketing online* yang sebelumnya telah ditentukan kebutuhan-kebutuhan dari sistem pada bab 4. Untuk perancangan sistem aplikasi secara umum diperlihatkan oleh Gambar 5.1. Kemudian tahap perancangan sistem selanjutnya adalah perancangann arsitektur yang berisikan *sequence diagram* dan *class diagram*, perancangan basis data, perancangan komponen, kemudian terakhir perancangan antrmuka.



Gambar 5.1 Diagram Perancangan Sistem Aplikasi

5.1.1 Perancangan Arsitektur

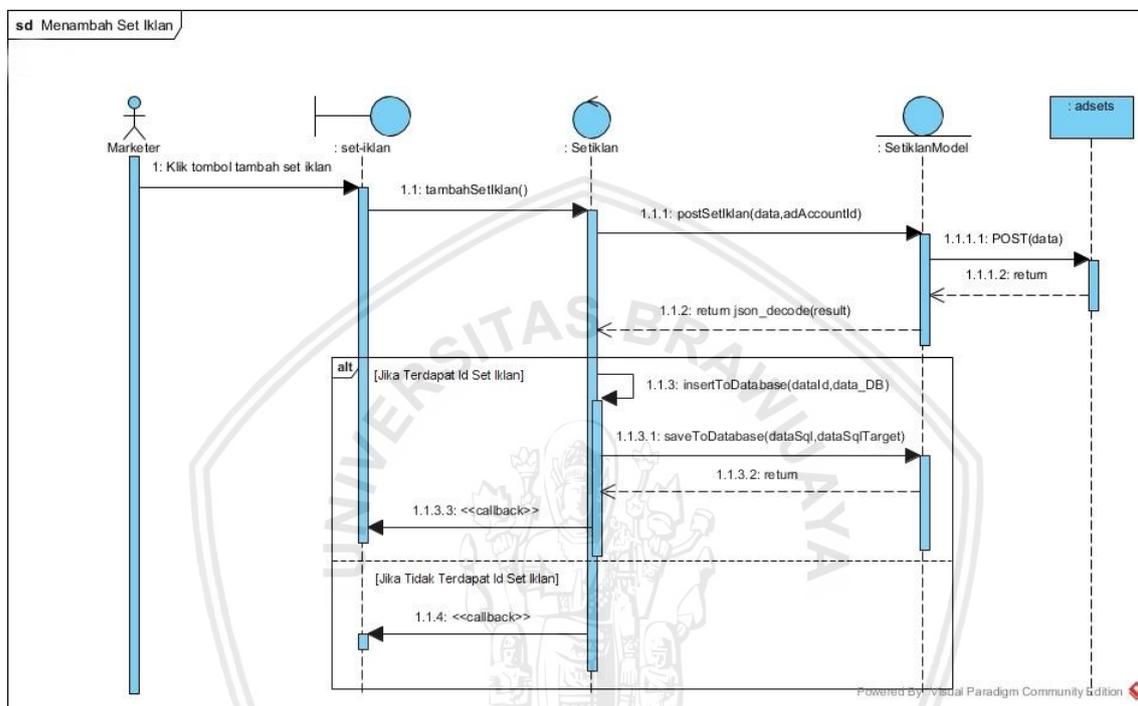
Perancangan arsitektur adalah perancangan dasar yang memberikan gambaran keseluruhan dari sebuah sistem yang akan dibuat. Ibarat sebuah rumah yang menggambarkan keseluruhan rumah seperti kamar-kamar, ruangan, ukuran dan bentuknya, serta hubungannya dengan satu sama lain. Pada perancangan arsitektur akan dibagi menjadi dua bagian yaitu perancangan *sequence diagram* dan perancangan diagram kelas.

5.1.1.1 Perancangan *Sequence Diagram*

Perancangan *sequence diagram* adalah salah satu UML pemodelan perancangan yang menjelaskan representasi alur ketika terjadi suatu interaksi objek yang berjalan berurutan. Seluruh objek didapatkan ketika pendefinisian spesifikasi kebutuhan. *Sequence diagram* sendiri memiliki pengertian sekuensial atau diagram yang menunjukkan interaksi berurutan.

a. Sequence Diagram Menambah Set Iklan

Sequence diagram menambah set iklan seperti pada gambar 5.2 menggambarkan interaksi antar objek untuk menambahkan data set iklan dengan melakukan POST pada API facebook yang jika mendapatkan *return* id dari adsets maka disimpan kepada database lokal dengan memanggil fungsi *insertToDatabase*.

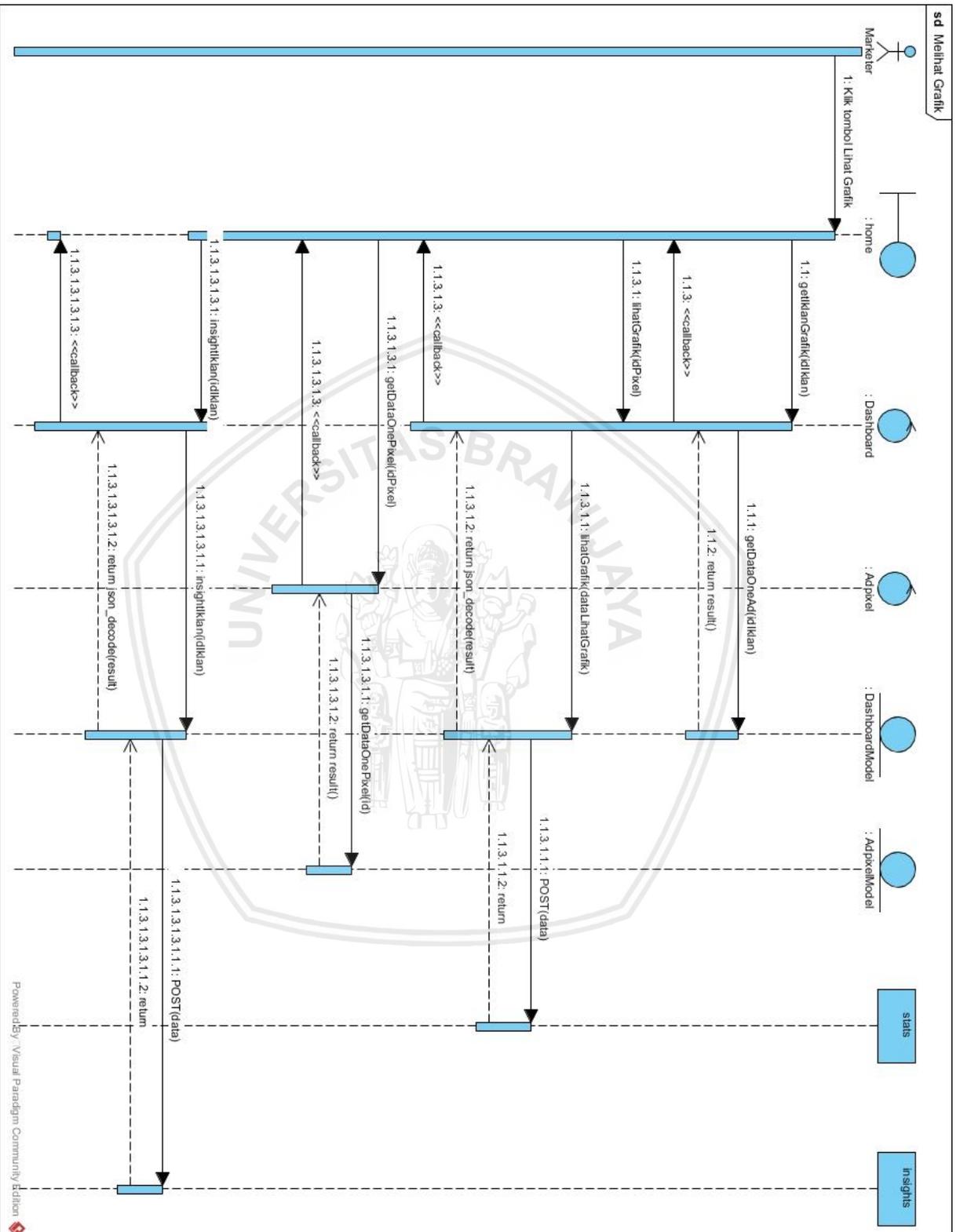


Gambar 5.2 Sequence Diagram Menambah Set Iklan

b. Sequence Diagram Melihat Grafik

Sequence diagram melihat grafik seperti ditunjukkan pada gambar 5.3 menunjukkan interaksi antar objek untuk melihat hasil wawasan iklan yang berjalan serta wawasan halaman website yang berupa trafik dan jumlah konversi barang dengan cara melakukan POST pada API facebook dengan *end point stats* dan *insights* yang kemudian mendapatkan *return* data wawasan iklan yang berjalan seperti *impression*, *click*, *cost*, dan *ctr* serta wawasan halaman website yang terdiri dari *page view*, *revenue*, dan *conversion*.

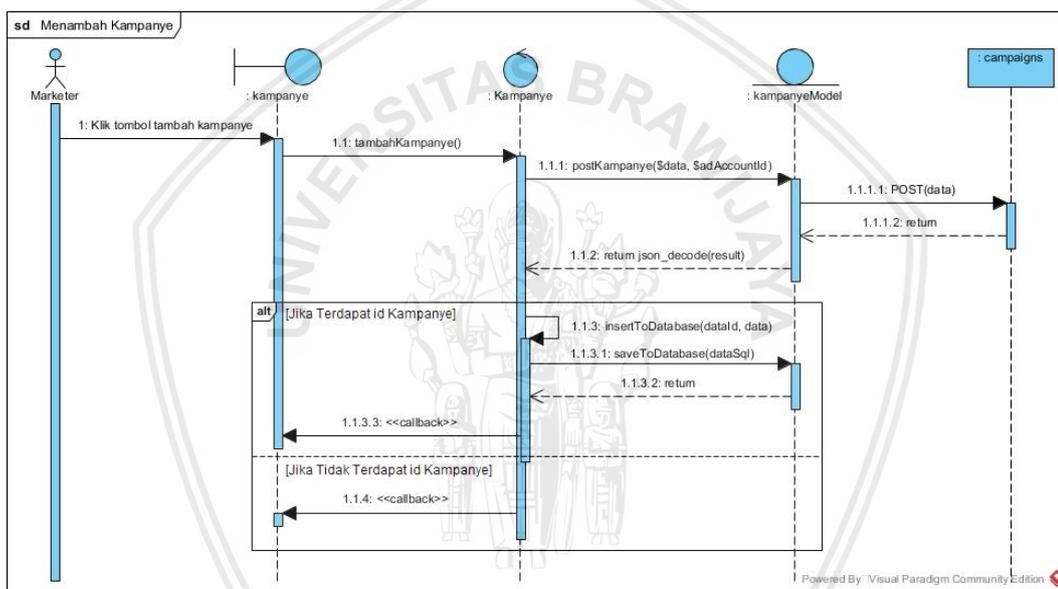




Gambar 5.3 Sequence Diagram Melihat Grafik

c. Sequence Diagram Menambah Kampanye

Sequence diagram menambah kampanye seperti pada gambar 5.4 menggambarkan interaksi antar objek untuk menyimpan kampanye yang telah dibuat dengan mengirimkan kepada API facebook dengan *end point campaign* menggunakan post, jika berhasil maka akan mendapatkan balikan berupa id kampanye dan akan disimpan kedalam database dengan memanggil fungsi *insertToDatabase* kemudian mencetak status *true* yang di encode menggunakan *json_encode* agar bisa terbaca oleh ajax dari *view* kampanye yang meminta *callback* berupa json dan kemudian akan ditampilkan *alert* berhasil yang menandakan penambahan kampanye telah berhasil dan ketika diklik tombol ok, maka modal akan tertutup dan *datatables* akan ter-*refresh*.



Gambar 5.4 Sequence Diagram Menambah Kampanye

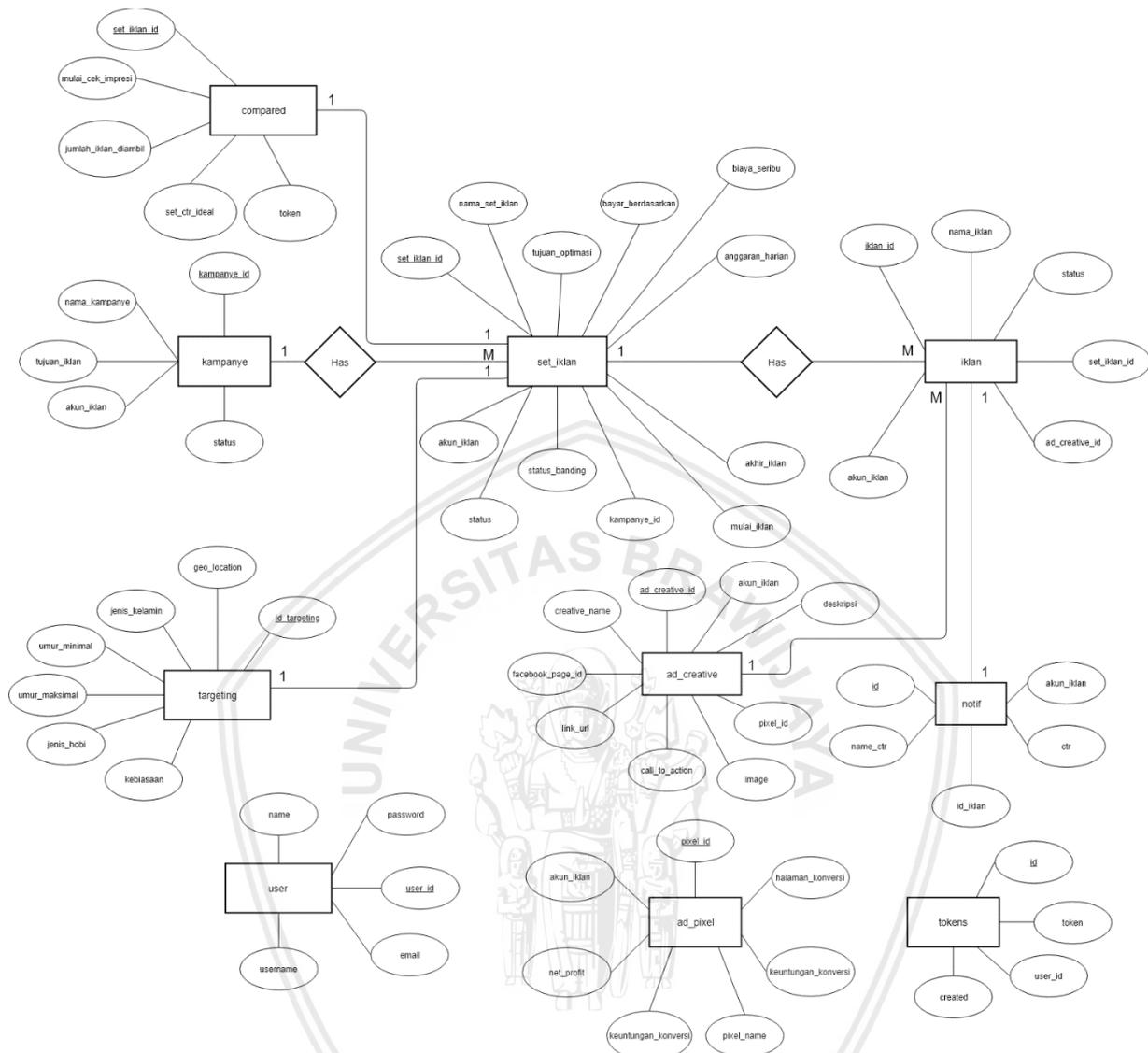
Pada gambar 5.5 terlihat perancangan *class diagram* yang telah diperoleh dari proses perancangan arsitektur sebelumnya yaitu *sequence diagram*, *message-message* dari *sequence diagram* akan digambarkan sebagai *method* pada setiap kelas dan kelas tersebut akan saling berhubungan satu dengan kelas yang lain. Pemodelan perancangan *class diagram* mengacu pada konsep yang diterapkan pada *framework* MVC. *Controller* sebagai pengatur dan jembatan antara *view* dan *model*. *Class controller* terdiri dari: Dashboard, Kampanye, Iklan, Setiklan, Adpixel, Welcome, Dokumentasi, Auth, Privacypolicy, Adcreative, Reset, Pengunjung dan User yang keseluruhan *controller* memiliki hubungan *generalisasi* pada *Class* CI_Controller. *Class model* terdiri dari: DashboardModel, KampanyeModel, SetiklanModel, IklanModel, AdpixelModel dan AdcreativeModel yang keseluruhan *model* memiliki hubungan *generalisasi* dengan *class* CI_Model.

5.1.2 Perancangan Basis Data

Perancangan basis data pada bab ini dibuat secara konseptual menggunakan *Entity Relationship Diagram* (ERD) dan *Physical Data Model* (PDM). Setidaknya ada sembilan entitas yaitu user, kampanye, set_iklan, iklan, ad_creative, ad pixel, notif, targeting, dan tokens. Perancangan basis data dari aplikasi *dashboard* untuk Analisa efektifitas *marketing online* berbasis website diilustrasikan menggunakan diagram pada Gambar 5.6 dan berakhir pada Gambar 5.7.

5.1.2.1 Entity Relationship Diagram (ERD)

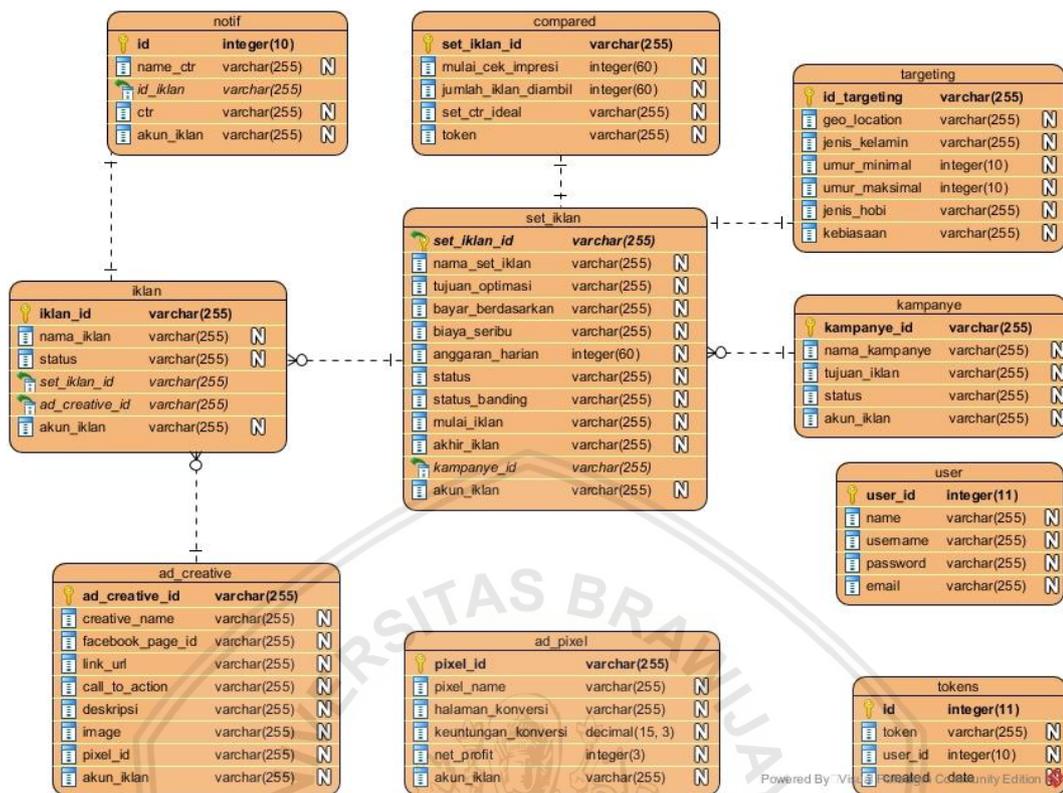
Entity Relationship Diagram (ERD) mendeskripsikan sebuah relasi antara penyimpanan (database) berdasarkan pada persepsi dunia nyata yaitu berkaitan dengan objek (*entity*) yang saling berhubungan atau berelasi antar objek-objek. Pada ERD seperti terlihat pada Gambar 5.6 tabel kampanye memiliki relasi *one to many* dengan tabel set_iklan yang artinya untuk satu kampanye dapat memiliki beberapa set iklan, untuk tabel set_iklan juga mempunyai relasi *one to many* dengan tabel iklan yang menandakan bahwa setiap satu set_iklan bisa memiliki beberapa iklan. Tabel set_iklan memiliki hubungan *one to one* dengan tabel *compared* dan *targeting* yang artinya setiap satu set_iklan memiliki satu *compared* dan satu *targeting*. tabel iklan juga memiliki relasi dengan tabel notif dengan hubungan *one to one* yang artinya satu iklan hanya memiliki satu notif pada tabel notif. Tabel iklan juga memiliki relasi dengan tabel ad_creative dengan hubungan *many to one* yang artinya banyak iklan bisa menggunakan satu ad_creative. Tabel tokens digunakan untuk menyimpan token saat ada yang meminta lupa *password*, tabel ad_pixel digunakan untuk menyimpan data ad_pixel untuk mendapatkan wawasan *traffic* dari website. Tabel user berisikan data user yang masukkan setelah ada user yang melakukan *register* yang kemudian akan digunakan autentikasi saat akan masuk ke dalam aplikasi.



Gambar 5.6 Entity Relationship Diagram

5.1.2.2 Physical Data Model (PDM)

PDM merupakan pemodelan sebuah basis data yang merepresentasikan keterkaitan antar tabel yang bersifat fisik karena setiap tabel akan memiliki beberapa kolom yang memiliki keunikan nama sendiri-sendiri, terdapat kolom sebagai *primary key* dan *foreign key* sebagai penghubung antar tabel jika hubungannya terdapat relasi yang saling berkaitan. Pada Gambar 5.7 terdapat 10 tabel yaitu: kampanye, set_iklan, iklan, notif, ad_pixel, ad_creative, tokens, user, compared, dan targeting yang akan didefinisikan mana kolom yang digunakan sebagai *primary key* dan mana yang digunakan sebagai kolom *foreign key*.



Gambar 5.7 Physical Data Model

5.1.3 Perancangan Komponen

Pada subbab perancangan komponen ini akan menerangkan tentang penggambaran fungsi atau unit pada setiap komponen yang terdapat pada perangkat lunak dengan menggunakan algoritme *pseudocode*. Pada perancangan komponen ini akan diambil tiga contoh fungsi algoritme *pseudocode* dari sistem.

5.1.3.1 Algoritme Fungsi tambahSetIklan

Algoritme pada fungsi tambahSetIklan adalah algoritme yang mendefinisikan suatu proses ketika aktor sudah teridentifikasi sebagai *marketer* dan melakukan penambahan set iklan baru pada aplikasi, kemudian data set iklan baru akan dikirimkan ke facebook API dengan menggunakan metode POST melauai curl yang kemudian akan mendapatkan balikan objek berupa id set iklan yang kemudian disimpan dalam *database* lokal.

Nama Class : Setiklan

Nama Function: tambahSetIklan

Algoritme :

1	Mulai
2	Mengecek autentikasi sesi marketing
3	Mendapatkan sesi akun iklan yang tersedia
4	Mengambil post input name dan memasukkan ke variabel nameSetIklan
5	

6	Mengambil post input tujuan dan memasukkan ke variabel optimizationGoal
7	Mengambil post input bayar dan memasukkan ke variabel billingEvent
8	Mengambil post input perseribu dan memasukkan ke variabel bidAmount
9	Mengambil post input pilih-kampanye dan memasukkan ke variabel campaignId
10	Mengambil post input anggaran-harian dan memasukkan ke variabel dailyBudget
11	Mengambil post input latitude dan memasukkan ke variabel latitude
12	Mengambil post input longitude dan memasukkan ke variabel longitude
13	Mengambil post input geolocation dan memasukkan ke variabel geolocation
14	Mengambil post input radiusModalSetIklan dan memasukkan ke variabel radius
15	Mengambil post input pilihanHobi dan memasukkan ke variabel pilihanHobi
16	
17	Memecah string jika terdapat dari variabel pilihanHobi dan menjadikannya array dan memasukkan ke variabel arrPilihanHobi
18	Mengambil post input kebiasaan dan memasukkan ke variabel kebiasaan
19	
20	Memecah string jika terdapat dari variabel kebiasaan dan menjadikannya array dan memasukkan ke variabel arrKebiasaan
21	
22	Mengambil post input jenis-kelamin dan memasukkan ke variabel a
23	Mengambil post input umur-min dan memasukkan ke variabel b
24	Mengambil post input umur-max dan memasukkan ke variabel c
25	Pengisian nilai jenisKelamin
26	Pengisian nilai umurMin
27	Pengisian nilai umurMax
28	Pengisian nilai jenisHobi
29	Pengisian nilai variabel biasaan
30	Jika variabel a tidak sama dengan null maka
31	Masukkan 'genders':\$a, kedalam variabel jenisKelamin
32	Jika variabel b tidak sama dengan null maka
33	Masukkan 'age_min':\$b, kedalam variabel umurMin
34	Jika variabel c tidak sama dengan null maka
35	Masukkan 'age_max':\$c, kedalam variabel umurMax
36	Jika variabel pilihanHobi tidak sama dengan null maka
37	Masukkan 'interest'[{ 'id':\$arrPilihanHobi[1], 'name': '\$arrPilihanHobi[0]'}], kedalam variabel jenisHobi
38	Jika variabel kebiasaan tidak sama dengan null maka
39	Masukkan 'behaviors'[{ 'id':\$arrKebiasaan[1], 'name': '\$arrKebiasaan[0]'}], kedalam variabel biasaan
40	Mengambil post input ad-start dan memasukkan ke variabel startTime
41	Mengambil post input ad-end dan memasukkan ke variabel endTime
42	Inisialisai variabel data array
43	Inisialisai variabel data_db array
44	Mengirimkan data dan adAccountId ke model setIklanModel fungsi postSetIklan
45	Jika pada variabel idSetIklan terdapat id maka
46	Masukkan id ke variabel dataId
47	Kirim dataId dan data_DB ke method insertToDatabase
48	Jika tidak ditemukan id maka
	Tampilkan isi dari idSetIklan
	selesai

5.1.3.2 Algoritme Fungsi getIklanGrafik

Algoritme pada fungsi hapusKampanye adalah algoritme dengan fungsionalitas untuk menghapus kampanye berdasarkan ID yang ingin dihapus.

Nama Class : Dashboard

Nama Function: getIklanGrafik

Algoritme :

1	Mulai
2	Mengirimkan data ke dashboard model dengan method getDataOneAd
3	Jika dalam variabel hasil terdapat isi maka
4	Inisialisasi variabel dataIklan dengan array
5	Cetak status TRUE dan cetak data
6	Jika dalam variabel hasil tidak terdapat isi maka
7	Cetak ID IKLAN INCORECT
8	selesai

5.1.3.3 Algoritme Fungsi tambahKampanye

Algoritme pada fungsi tambahKampanye adalah algoritme dengan fungsionalitas aktor pada saat sudah teridentifikasi sebagai marketer dan ingin melakukan penambahan kampanye baru pada aplikasi, kemudian data kampanye baru akan dikirimkan ke facebook API dengan menggunakan metode POST melau curl yang kemudian akan mendapatkan balikan objek berupa id kampanye yang kemudian disimpan dalam database lokal.

Nama Class : Kampanye

Nama Function: tambahKampanye

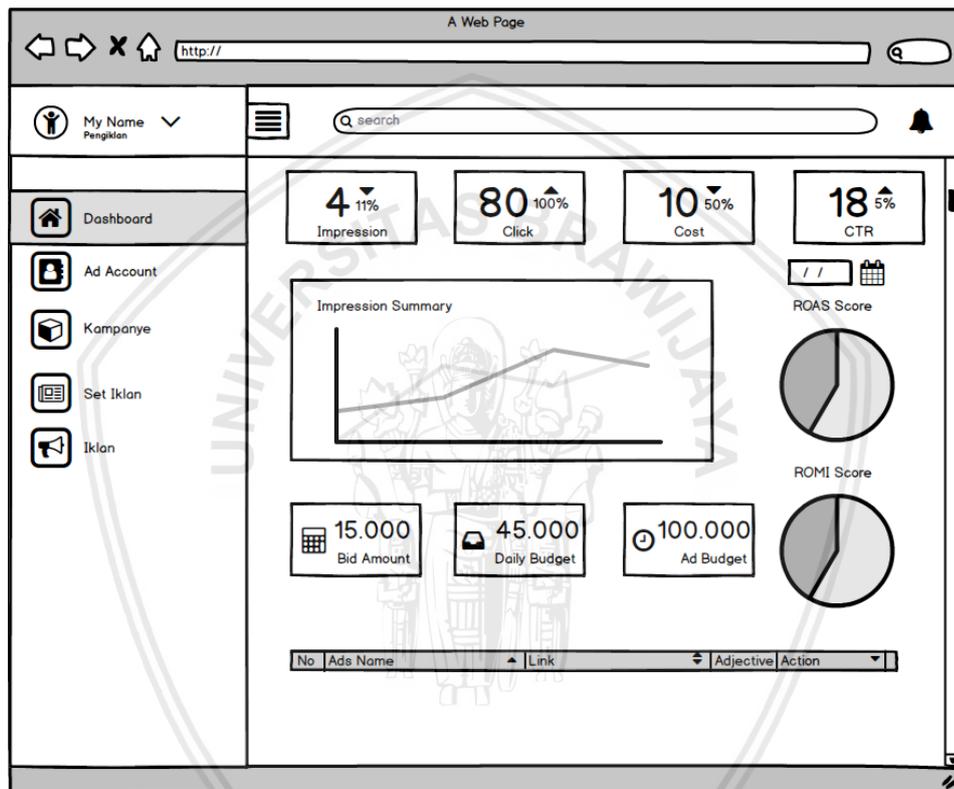
Algoritme :

1	Mulai
2	Mengecek autentikasi sesi marketing dan memasukkan ke variabel token
3	
4	mengambil post input akun iklan dan memasukkan ke variabel akunIklanForm
5	
6	Membuat array data
7	mengambil post input name dan memasukkan ke array name
8	mengambil post input objective dan memasukkan ke array objective
9	Inisialisasi variabel status dengan isi string "PAUSED"
10	Inisialisasi variabel access_token berisi dari token
11	Mengirimkan data dan adAccountId ke kampanyeModel
12	Jika pada variabel idKampanye terdapat id maka
13	Kirim dataId dan data ke method insertToDatabase
14	Jika tidak ditemukan id maka
15	Cetak variabel idKampanye
16	selesai

5.1.4 Perancangan Antarmuka

Subbab perancangan antarmuka, akan digambarkan sebagai suatu sketsa dari aplikasi yang akan dibuat. Pembuatan sketsa atau *mockup* ini dibantu oleh aplikasi balsamiq dan tidak ada kode program sama sekali didalamnya. Untuk perancangan antarmuka ini akan diambil tiga sampel contoh, yaitu: perancangan antarmuka dashboard, kampanye, dan iklan yang akan ditunjukkan pada Gambar 5.8 hingga Gambar 5.10.

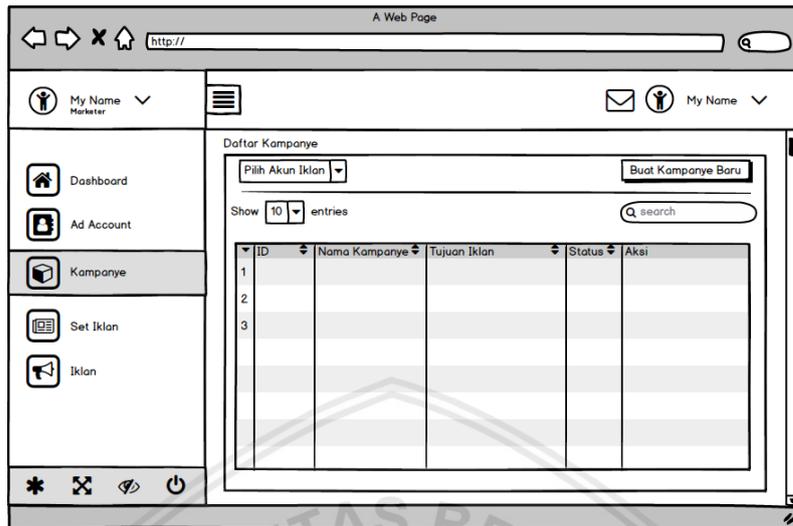
5.1.4.1 Perancangan Antarmuka Dashboard



Gambar 5.8 Perancangan Antarmuka Dashboard

Pada gambar 5.8 adalah merupakan tampilan antarmuka perancangan pada halaman *dashboard* yang terdiri dari *main page* dan *header* yang berisi menu. Bagian *header* terdapat nama dan foto profil pemilik facebook. Kemudian bawahnya terdapat pilihan akun iklan, karena setiap pemilik facebook bisa memiliki banyak akun iklan dan jika dipilih salah satu maka akan menampilkan status kepemilikan sebuah akun iklan (Admin, Pengiklan dan Analis), kemudian bawahnya terdapat menu *dashboard* yang berisi kumpulan wawasan dari iklan yang berjalan dan juga wawasan website baik *page view* maupun konversi, bawahnya lagi terdapat menu kampanye yang berisi tujuan iklan secara garis besar, bawahnya terdapat set iklan yang berfungsi untuk membuat *targeting* dan *budgeting* dari sebuah iklan, yang terakhir ada iklan yang berfungsi untuk membuat iklan.

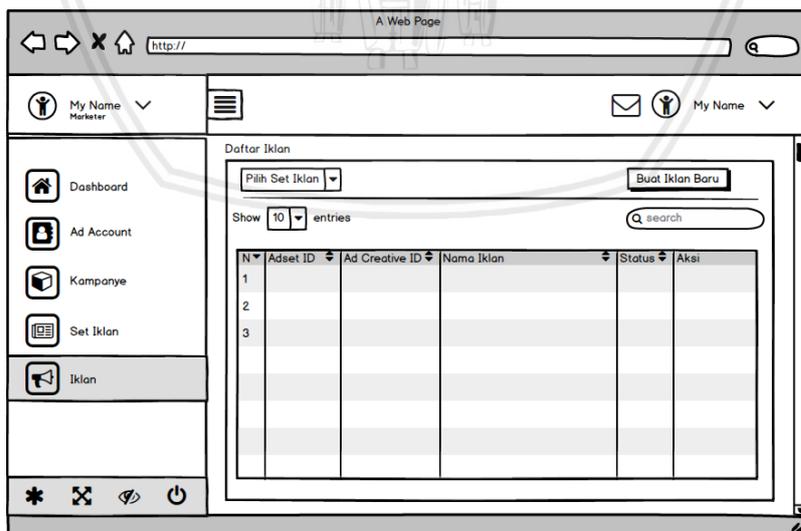
5.1.4.2 Perancangan Antarmuka Kampanye



Gambar 5.9 Perancangan Antarmuka Kampanye

Pada gambar 5.9 adalah merupakan tampilan antarmuka untuk melihat seluruh kampanye yang dimiliki akun iklan. Terlihat tombol untuk membuat kampanye baru pada pojok kanan atas tabel, kemudian bawahnya terdapat kolom pencarian yang digunakan untuk *men-sorting* atau mencari informasi yang diinginkan, sebelah pojok kiri atas terdapat jumlah *row* atau baris yang ingin ditampilkan. Dan bawahnya terdapat tabel yang menampilkan informasi kampanye yang sudah pernah dibuat.

5.1.4.3 Perancangan Antarmuka Iklan



Gambar 5.10 Perancangan Antarmuka Iklan

Gambar 5.10 adalah merupakan tampilan antarmuka pada fungsionalitas melihat seluruh iklan yang telah dibuat yang dimiliki oleh akun iklan. Terlihat tombol untuk membuat iklan baru pada pojok kanan atas tabel, kemudian bawahnya terdapat kolom pencarian yang digunakan untuk men-*sorting* atau mencari informasi yang diinginkan, sebelah pojok kiri atas terdapat jumlah *row* atau baris yang ingin ditampilkan. Dan bawahnya terdapat tabel yang menampilkan informasi iklan yang telah dibuat.

5.2 Implementasi

Subbab implementasi akan dijelaskan proses implementasi sistem yang hendak dikerjakan berdasarkan hasil dari fase sebelumnya, yaitu fase perancangan. Implementasi berisi kode program yang dibuat berdasarkan algoritme yang telah dibuat pada fase perancangan.

5.2.1 Spesifikasi Sistem

Subbab spesifikasi sistem menjelaskan spesifikasi perangkat yang akan dipakai pada pengembangan sistem. Spesifikasi sistem akan dibagi menjadi empat subbab lagi, yaitu spesifikasi perangkat keras, spesifikasi perangkat lunak, spesifikasi *operating system* dan spesifikasi arsitektur sistem. Di bawah ini akan dijelaskan pada beberapa sub bab terkait dengan spesifikasi sistem. Pada tabel 5.1 merupakan penjelasan mengenai spesifikasi *hardware* yang akan dipakai, kemudian di tabel 5.2 merupakan penjelasan mengenai perangkat lunak yang diandalkan dalam pembangunan aplikasi, tabel 5.3 menjelaskan mengenai spesifikasi sistem operasi dan pada tabel 5.4 menjelaskan tentang spesifikasi arsitektur sistem yang digunakan dalam pengembangan aplikasi *dashboard* untuk analisa efektivitas *marketing online*.

5.2.1.1 Spesifikasi Perangkat Keras

Pembangunan aplikasi *dashboard* untuk Analisa efektivitas *marketing online* didukung oleh beberapa perangkat keras yang lebih jelasnya akan dijabarkan pada Tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
System Model	ASUSTek COMPUTER INC. X 455LF
Processor	Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz (4 CPUs)
Memory (RAM)	8192MB RAM
Display	Intel(R) HD Graphics 5500 NVIDIA GeForce 930M

Nama Komponen	Spesifikasi
Hardisk	465GB Hitachi HGST HTS545050A7E680 (SATA)

5.2.1.2 Spesifikasi Perangkat Lunak

Menjabarkan daftar perangkat lunak yang dipakai untuk mempermudah pembangunan aplikasi seperti perangkat lunak untuk *text editor*, *design editor*, dan *documentation editor* dapat dilihat lebih lengkap pada Tabel 5.2.

Tabel 5.2 Spesifikasi Perangkat Lunak

Nama Komponen	Spesifikasi
Editor Dokumentasi	Microsoft Word 2016
Editor Perancangan	Visual Paradigm 15.2, Draw.io Desktop
Editor Pemrograman	Visual Studio Code
Bahasa Pemrograman	PHP, Javascript
Framework	Codeigniter, Bootstrap
DBMS	Mysql
Browser	Mozilla Firefox Developer Edition
Hosting	1GB

5.2.1.3 Spesifikasi Sistem Operasi

Tabel 5.3 Spesifikasi Sistem Operasi

Nama Komponen	Spesifikasi
Sistem Operasi	Windows 10 Pro

5.2.1.4 Spesifikasi Server

Detail *server* yang akan dipakai oleh aplikasi *dashboard* untuk analisa efetifitas *marketing online* untuk dilakukan implementasi pada *shared hosting*.

Tabel 5.4 Spesifikasi Server

Nama Komponen	Spesifikasi
CPU	1 Core
RAM (GB)	1GB
SSD Disk Space	Unlimited

5.2.1.5 Batasan Implementasi

Terdapat batasan-batasan pada saat tahap implementasi yang akan dilakukan. Yaitu sebagai berikut.

1. Aplikasi akan dikembangkan menggunakan PHP, HTML, CSS, Javascript, dan AJAX.
2. Aplikasi akan dibangun pada platform *website*.
3. MySQL digunakan sebagai Relational Database Management System (RDBMS) untuk membangun aplikasi.

5.2.2 Implementasi Basis Data

Hasil dari perancangan pada subbab 5.1.2 yaitu *Entity Relationship Diagram* (ERD) pada Gambar 5.8 dan *Physical Data Model* (PDM) yang ditunjukkan Gambar 5.9 akan dibuat implementasi yang menghasilkan *Data Definition Language* (DDL) dengan contoh tabel kampanye pada urutan satu, tabel set_iklan pada urutan kedua dan tabel iklan pada urutan ketiga seperti pada Tabel 5.5. diambil sampel tabel kampanye, set_iklan, dan iklan yang merupakan penggambaran bagaimana pembuatan sebuah basis data beserta tabel-tabelnya atau DDL dari aplikasi *dashboard* untuk analisa efektifitas *marketing online*.

Tabel 5.5 Implementasi Basis Data Dengan DDL

No	Data Definition Language
1.	Create table kampanye (kampanye_id varchar(255) primary key, nama_kampanye varchar(255), tujuan_iklan varchar(255), status varchar(255), akun_iklan varchar(255))
2.	Create table set_iklan (set_iklan_id varchar(255) primary key, nama_set_kampanye varchar(255), tujuan_optimasi varchar(255), bayar_berdasarkan varchar(255), biaya_seribu varchar(255), anggaran_harian int(60), status varchar(255), status_banding varchar(255), mulai_iklan varchar(255), akhir_iklan varchar(255), kampanye_id varchar(255) foreign key, akun_iklan varchar(255))
3.	Create table iklan (iklan_id varchar(255) primary key, nama_iklan varchar(255), status varchar(255), set_iklan_id varchar(255) foreign key, ad_creative_id varchar(255) foreign key, akun_iklan varchar(255))

5.2.3 Implementasi Kode Program

Hasil dari algoritme perncangan komponen pada subbab 5.1.3 akan diubah pada subbab 5.2.3 menjadi bahasa yang lebih teknis atau bahasa komputer yaitu dengan bahasa PHP. Selanjutnya akan ditunjukkan 3 contoh sampel kode program dengan bahasa PHP yaitu pada fungsi tambahSetIklan, fungsi getIklanGrafik, dan fungsi tambahKampanye.

5.2.3.1 Kode Program Fungsi tambahSetIklan

Fungsi tambahSetIklan adalah anggota *function* dari *class* Setiklan yang akan berfungsi ketika *marketer* melakukan tambah set iklan. Fungsi tambahSetIklan dijabarkan dalam Tabel 5.1 sebagai berikut.

Nama *Class* : Setiklan

Nama Fungsi : tambahSetIklan

Sourcecode :

```

1  $token = $this->session->fb_access_token;
2      $adAccountId = $this->session->pilihanAdAccount;
3
4      $nameSetIklan = $this->input->post('name');
5      $optimizationGoal = $this->input->post('tujuan');
6      $billingEvent = $this->input->post('bayar');
7      $bidAmount = $this->input->post('perseribu');
8      $campaignId = $this->input->post('pilih-kampanye');
9      $dailyBudget = $this->input->post('anggaran-harian');
10
11     $latitude = $this->input->post('latitude');
12     $longitude = $this->input->post('longitude');
13     $geolocation = $this->input->post('geolocation');
14     $radius = $this->input->post('radiusModalSetIklan');
15
16     $pilihanHobi = $this->input->post('pilihanHobi');
17     $arrPilihanHobi = explode('|', $pilihanHobi);
18     $kebiasaan = $this->input->post('kebiasaan');
19     $arrKebiasaan = explode('|', $kebiasaan);
20
21     $a = $this->input->post('jenis-kelamin');
22     $b = $this->input->post('umur-min');
23     $c = $this->input->post('umur-max');
24
25     $jenisKelamin = "";
26     $umurMin = "";
27     $umurMax = "";
28     $jenisHobi = "";
29     $biasaan = "";
30

```

```

31     if ($a != ""){
32         $jenisKelamin = "'genders': $a,";
33     }if ($b != ""){
34         $umurMin = "'age_min': $b,";
35     }if ($c != ""){
36         $umurMax = "'age_max': $c,";
37     }if ($pilihanHobi != ""){
38
39         $jenisHobi =
40         "'interests':[{'id':$arrPilihanHobi[1],'name':'$arrPilihanHobi[0]'"
41         "}],";
42     }if ($kebiasaan != ""){
43         $biasaan =
44         "'behaviors':[{'id':$arrKebiasaan[1],'name':'$arrKebiasaan[0]'}]",
45         ";";
46     $startTime = $this->input->post('ad-start');
47     $endTime = $this->input->post('ad-end');
48
49     $data = array(
50         'name' => $nameSetIklan,
51         'optimization_goal' => $optimizationGoal,
52         'billing_event' => $billingEvent,
53         'bid_amount' => $bidAmmount,
54         'campaign_id' => $campaignId,
55         'daily_budget' => $dailyBudget,
56         'targeting' => "{
57             $umurMin
58             $umurMax
59             $jenisKelamin
60             'geo_locations':{
61                 'custom_locations':[
62                 {'address_string': '$geolocation', 'radius': '$radius',
63                 'latitude': '$latitude',
64                 'longitude': '$longitude',
65                 'radius': '$radius',
66                 'distance_unit': 'kilometer'
67                 }
68                 ],
69                 'location_types': ['recent', 'home']
70             },
71             $jenisHobi
72             $biasaan
73             'publisher_platforms':[
74                 'facebook'
75             ]
76         }",
77         'start_time' => strtotime($startTime.'+07:00'),
78         'end_time' => strtotime($endTime.'+07:00'),
79         'status' => "PAUSED",
80         'access_token' => $token
81     );
82     $data_DB = array(
83         'adAccountId' => $adAccountId,
84         'campaign_id' => $campaignId,
85         'name' => $nameSetIklan,
86         'optimization_goal' => $optimizationGoal,
87         'billing_event' => $billingEvent,
88         'bid_amount' => $bidAmmount,
89         'daily_budget' => $dailyBudget,
90         'targeting' => $geolocation.'.'.$radius,
91         'startTime' => $startTime,
92         'jenisKelamin' => $a,

```

```

100     'umurMin' => $b,
101         'umurMax' => $c,
102         'jenisHobi' => $arrPilihanHobi[0],
103         'kebiasaan' => $arrKebiasaan[0],
104         'endTime' => $endTime
105     );
106
107     $idSetIklan = $this->setiklan_m->postSetIklan($data,
108 $adAccountId);
109
110     if(isset($idSetIklan->id)){
111         $dataId = $idSetIklan->id;
112         $this->insertToDatabase($dataId, $data_DB);
113     }else{
114         echo json_encode($idSetIklan);
115     }

```

Kode Program 5.1 Kode Program Fungsi tambahSetIklan

5.2.3.2 Kode Program Fungsi getIklanGrafik

getIklanGrafik adalah fungsi dari kelas Kampanye yang berfungsi ketika *marketer* melakukan tambah kampanye. Implementasi dari fungsi getIklanGrafik bisa diamati dalam Program Kode 5.2 sebagai berikut.

Nama Class : Dashboard

Nama Fungsi : getIklanGrafik

Sourcecode :

```

1     $hasil = $this->dashboard_m->getDataOneAd($idIklan);
2
3     if(isset($hasil)){
4         $dataIklan = array(
5             'namaIklan' => $hasil[0]->nama_iklan,
6             'pixelId' => $hasil[0]->pixel_id
7         );
8         echo json_encode(array("status" => TRUE, "data" =>
9 $dataIklan));
10        }else{
11            echo "ID IKLAN INCORECT";

```

Kode Program 5.2 Kode Program Fungsi getIklanGrafik

5.2.3.3 Kode Program Fungsi tambahKampanye

Fungsi tambahKampanye adalah bagian daripada *class* Kampanye yang berfungsi ketika *marketer* melakukan tambah kampanye. Fungsi tambahKampanye dapat diamati dalam Kode Program 5.3.

Nama Class : Kampanye

Nama Fungsi : tambahKampanye

Sourcecode :

```
1     $token = $this->session->fb_access_token;
2     $adAccountId = $this->input->post('akunIklanForm');
3     $data = array(
4         'name' => $this->input->post('nameKampanye'),
5         'objective' => $this->input->post('tujuan'),
6         'status' => "PAUSED",
7         'access_token' => $token,
8         'akun_iklan' => $adAccountId
9     );
10    $idKampanye = $this->kampanye_m->postKampanye($data,
11    $adAccountId);
12
13    if(isset($idKampanye->id)){
14        $dataId = $idKampanye->id;
15        $this->insertToDatabase($dataId,$data);
16    }else{
17        echo json_encode($idKampanye);}

```

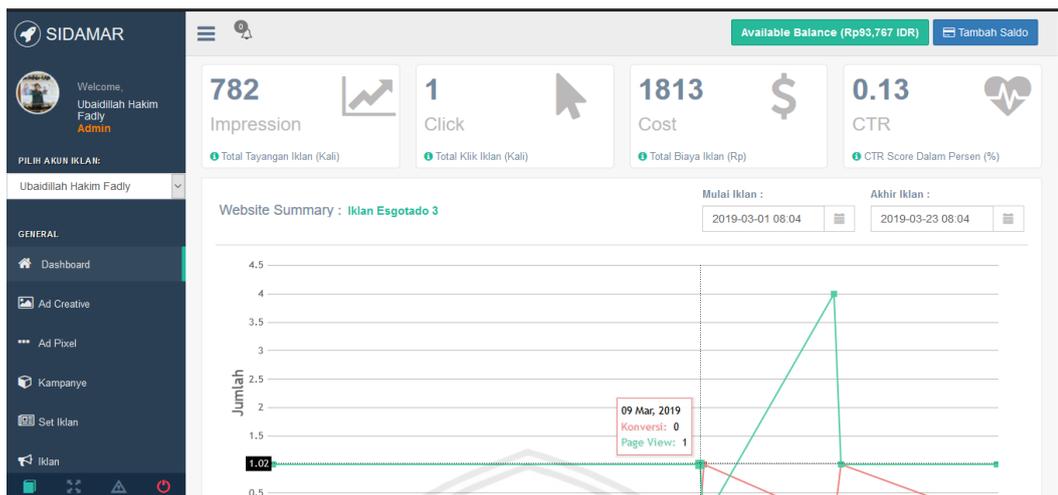
Kode Program 5.3 Kode Program Fungsi tambahKampanye

5.2.4 Implementasi Antarmuka

Implementasi antarmuka pada aplikasi *dashboard* untuk analisa efektivitas *marketing online* bisa dilihat di Gambar 5.11 sampai dengan 5.16. Implementasi antarmuka adalah hasil penerapan dari perancangan antarmuka pada bab 5. Berikut adalah contoh implementasi yang diantaranya adalah implementasi antarmuka melihat dashboard, menambah kampanye, menambah set iklan, melihat ad creative, melihat ad pixel, dan menambah iklan.

5.2.4.1 Implementasi Antarmuka Melihat Dashboard

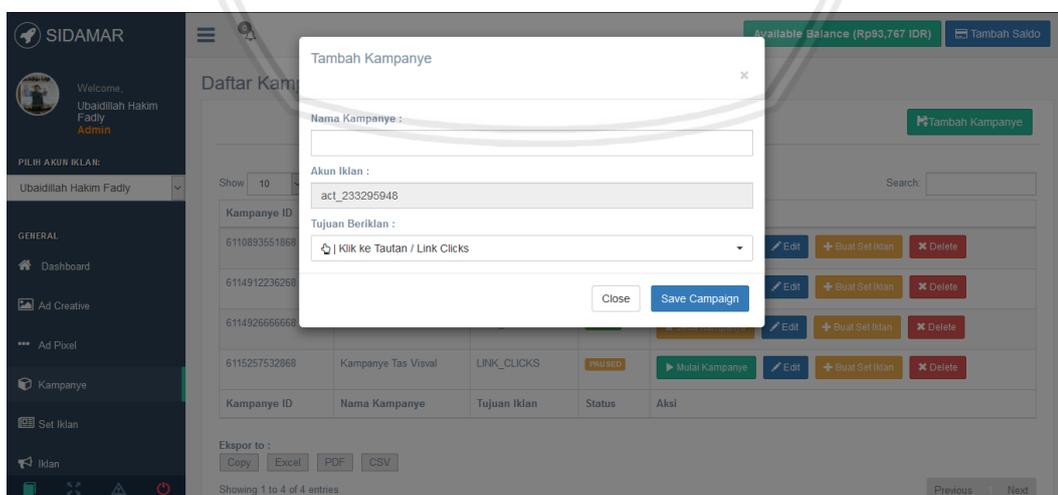
Implementasi antarmuka melihat dashboard seperti di ditunjukkan oleh Gambar 5.11 terlihat beberapa wawasan yang ditampilkan seperti wawasan iklan yang terdiri dari impresi yaitu informasi berapa kali iklan ditayangkan, kemudian terdapat klik yang memberikan informasi berapa kali iklan di klik, *cost* yang menunjukkan biaya yang telah dikurangi atau total *cost* yang dikeluarkan hingga pada saat tombol "Lihat Grafik" di klik, kemudian ada informasi *CTR* yang berisi informasi perhitungan jumlah klik yang dibagi dengan impresi dan dikalikan dengan 100. Kemudian dibawahnya terdapat wawasan website yang ditampilkan dalam bentuk grafik yang berisi informasi berapa orang yang telah mengunjungi website dan berapa yang telah melakukan *purchase*. Kemudian terdapat informasi keuntungan yang dihitung dari perkiraan keuntungan bersih tiap barang yang dikalikan dengan jumlah barang yang masuk *purchase*. Setelahnya terdapat informasi tentang metrik ROAS dan ROMI yang terdiri dari 3 golongan yaitu rendah, bagus dan sangat bagus.



Gambar 5.11 Implementasi Antarmuka Melihat Dashboard

5.2.4.2 Implementasi Antarmuka Menambah Kampanye

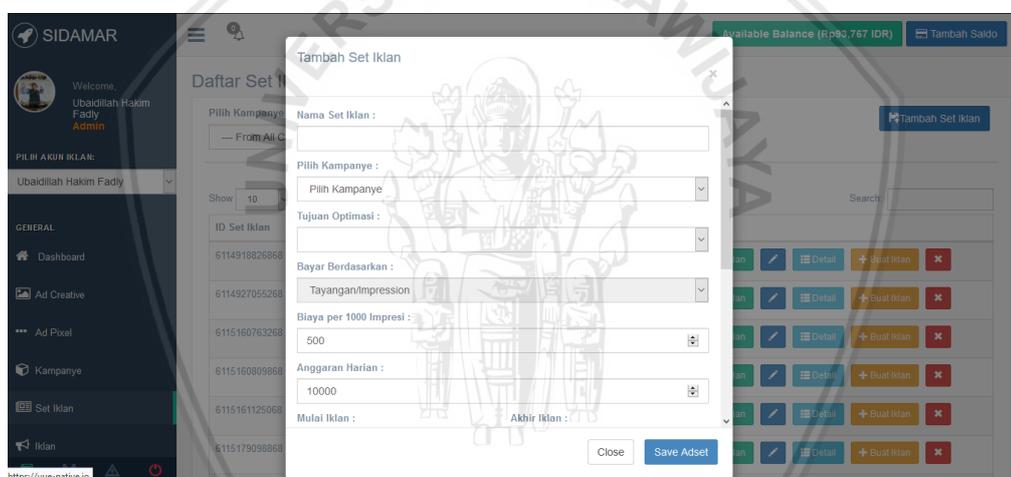
Implementasi antarmuka menambah kampanye seperti terlihat di Gambar 5.12 terlihat pilihan kampanye pada menu kemudian akan terlihat halaman kampanye yang menampilkan tabel kampanye yang berada di *list*. Kemudian pada pojok kanan atas terlihat tombol “Tambah Kampanye” yang jika diklik akan menampilkan *bootstrap modal* yang berisi *form* untuk menambahkan kampanye yang terdiri dari nama kampanye yang digunakan untuk memasukkan nama kampanye yang diinginkan, kemudian terdapat *form* akun iklan yang berisi ID akun iklan yang secara *default* tidak bisa dirubah. Selanjutnya terdapat form tujuan beriklan yang berisi tujuan besar atau *goal* secara umum dari iklan itu dibuat. Terakhir terdapat tombol “Save Campaign” yang digunakan untuk menyimpan iklan.



Gambar 5.12 Implementasi Antarmuka Menambah Kampanye

5.2.4.3 Implementasi Antarmuka Menambah Set Iklan

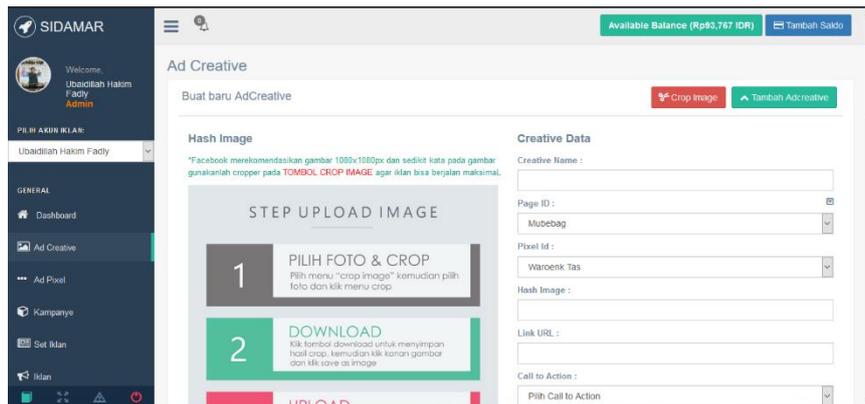
Implementasi antarmuka menambah set iklan seperti pada Gambar 5.13 terlihat *bootstrap modal* yang berisi *form* nama set iklan yang digunakan untuk mengisi nama iklan, kemudian *dropdown* pilihan kampanye untuk memilih kampanye yang digunakan untuk set iklan, kemudian terdapat tujuan optimasi yaitu tujuan yang lebih spesifik ketika beriklan, selanjutnya ada bayar berdasarkan yang digunakan untuk penagihan pembayaran yang di set secara default menjadi tayangan/impresi, kemudian terdapat biaya per 1000 tayangan/impresi yang digunakan untuk menentukan berapa yang akan dibayarkan ke facebook ketika mencapai setiap 1000 tayangan, kemudian terdapat anggaran harian yang menunjukkan perkiraan anggaran harian tiap harinya, kemudian terdapat *form targeting* yang digunakan untuk spesifikasi target market yang terdiri dari ketertarikan, kebiasaan, daerah cakupan, jenis kelamin, umur minimal, dan umur maksimal.



Gambar 5.13 Implementasi Antarmuka Menambah Set Iklan

5.2.4.4 Implementasi Antarmuka Melihat Ad Creative

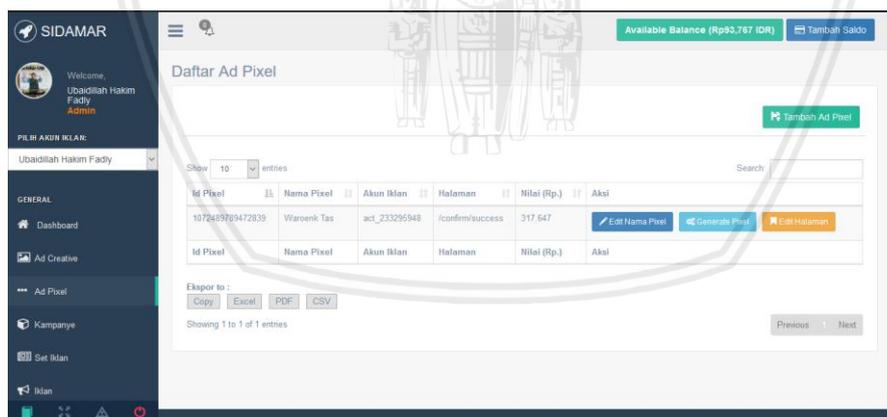
Implementasi antarmuka melihat ad creative seperti terlihat pada gambar 5.14, dimuat sebuah halaman yang akan ditampilkan setelah aktor menekan tombol ad creative pada menu. Pada halaman ad creative terlihat tabel yang menampilkan ad creative dari setiap akun iklan yang dipilih.



Gambar 5.14 Implementasi Antarmuka Melihat Ad Creative

5.2.4.5 Implementasi Antarmuka Melihat Ad Pixel

Implementasi antarmuka melihat ad pixel seperti terlihat pada Gambar 5.15, tersedia halaman ad pixel yang ditampilkan ketika aktor menekan tombol ad pixel pada menu, ditampilkan seluruh ad pixel berdasarkan akun iklan yang dipilih. Untuk akun pribadi hanya diperbolehkan memuat 1 ad pixel. Pada gambar terlihat tombol edit nama pixel yang digunakan untuk merubah nama ad pixel, kemudian *generate* kode yang digunakan untuk men-*generate* kode sesuai dengan informasi yang dimasukkan pada saat tambah ad pixel, Kode ini nantinya akan ditanamkan pada web yang ingin dilihat wawasannya, kemudian terdapat tombol edit halaman web yang digunakan untuk mengedit halaman website.

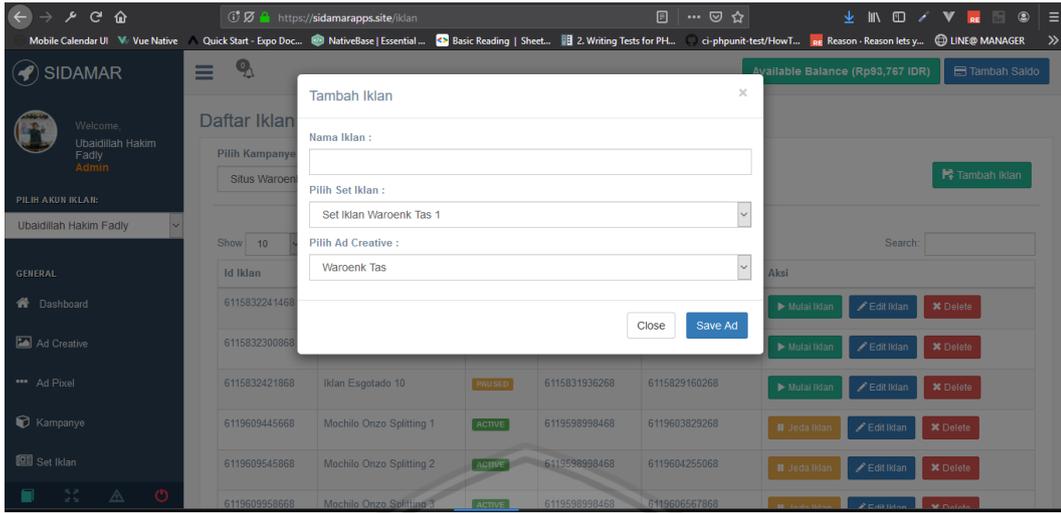


Gambar 5.15 Implementasi Antarmuka Melihat Ad Pixel

5.2.4.6 Implementasi Antarmuka Menambah Iklan

Implementasi antarmuka menambah iklan pada Gambar 5.16, Terlihat *form* tambah iklan yang berisi *form* nama iklan yang digunakan untuk memasukkan nama iklan yang diinginkan kemudian terdapat *dropdown* set iklan yang digunakan untuk memilih set iklan yang digunakan untuk iklan, dan *dropdown* pilih ad creative yang digunakan untuk memilih konten yang akan digunakan dalam iklan yang akan dibuat.





Gambar 5.16 Implementasi Antarmuka Menambah Iklan



BAB 6 PENGUJIAN

Pada bab ini, yaitu bab pengujian atau *testing* akan berisi tahapan terkait pengujian terhadap aplikasi *dashboard* untuk analisa efektivitas *marketing online* dan hasil yang diperoleh analis dari pengujian sistem. Pengujian yang akan digunakan adalah *unit testing*, *integration testing* dan *validation testing*.

6.1 Pengujian Unit

Dilaksanakan untuk menguji logika dalam satu unit yang diperoleh dari hasil algoritme pada bab 5 perancangan sistem. Pengujian unit ini dilakukan dengan memanfaatkan *basis path testing*. Pengujian unit akan dilaksanakan untuk tiga fungsi yang diambil sebagai contoh. Yaitu fungsi *tambahSetIklan* pada kelas *SetIklan*, fungsi *getIklanGrafik* pada kelas *Dashboard*, dan fungsi *tambahKampanye* pada kelas *Dashboard*.

6.1.1 Pengujian Unit Klas *SetIklan* pada Fungsi *tambahSetIklan()*

1. Pseudocode

1	START	1
2	INIT token = session(fb_access_token)	
3	INIT adAccountId = session(pilihanAdAccount)	
4	INIT namaSetIklan = POST[name]	
5	INIT optimizationGoal = POST[tujuan]	
6	INIT billingEvent = POST[bayar]	
7	INIT bidAmmount = POST[perseribu]	
8	INIT campaignId = POST[pilih-kampanye]	
9	INIT dailyBudget = POST[anggaran-harian]	
10	INIT latitude = POST[latitude]	
11	INIT longitude = POST[longitude]	
12	INIT geolocation = POST[geolocation]	
13	INIT radius = POST[radiusModalSetIklan]	
14	INIT pilihanHobi = POST[pilihanHobi]	2
15	INIT arrPilihanHobi = explode[pilihanHobi]	
16	INIT kebiasaan = POST[kebiasaan]	
17	INIT arrKebiasaan = explode[kebiasaan]	
18	INIT a = POST[jenis-kelamin]	
19	INIT b = POST[umur-min]	
20	INIT c = POST[umur-max]	
21	INIT jenisKelamin = null	
22	INIT umurMin = null	
23	INIT umurMax = null	
24	INIT jenisHobi = null	
25	INIT biasaan = null	
26		
27	IF (a != null) THEN	3
28	INIT jenisKelamin = "'genders':a,"	4
29	IF (b != null) THEN	5
30	INIT umurMin = "'age_min':b,"	6
31	IF (c != null) THEN	7
32	INIT umurMax = "'age_max':c,"	8
33	IF (pilihanHobi != null) THEN	9
34	INIT jenisHobi =	
	"'interests':[{ 'id':arrPilihanHobi[1], 'name': 'arrPilihanHobi[0] }],"	10
35	IF (kebiasaan != null) THEN	11

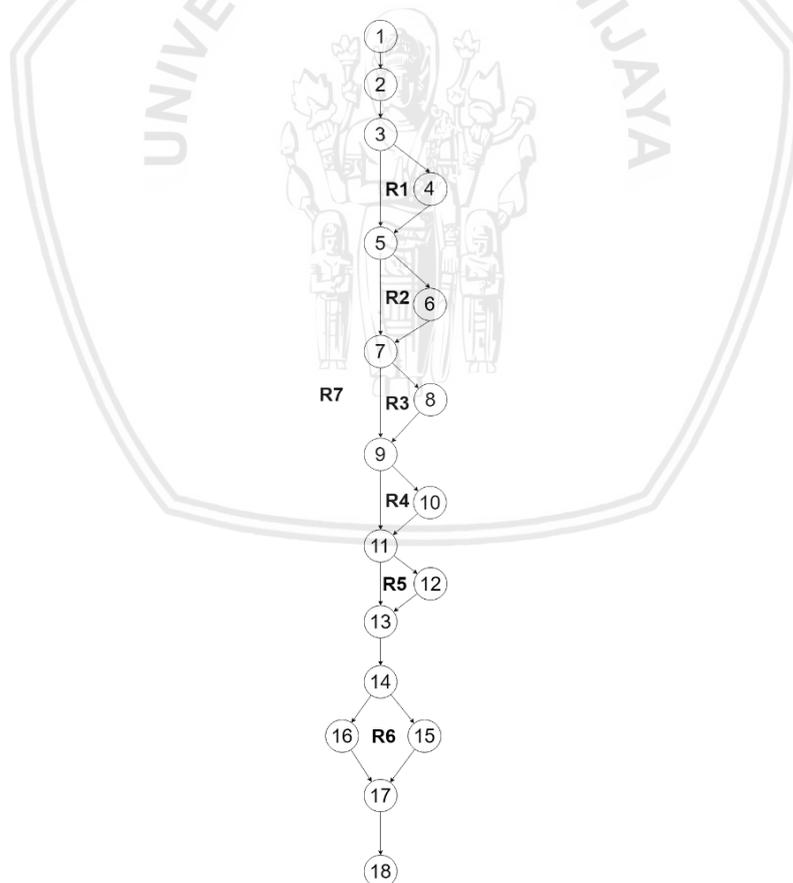
```

36     INIT biasanya =
37     "'behaviors':[{'id':arrKebiasaan[1],'name':'arrKebiasaan[0]'}],"; } 12
38
39
40     INIT startTime = POST[ad-start]
41     INIT endTime = POST[ad-end]
42
43     INIT data = array
44     INIT data_DB = array
45
46     INIT idSetIklan = kirim data dan adAccountId ke database } 13
47
48     IF (terdapat id) THEN _____ 14
49         INIT dataId = id
50         CALL : insertToDatabase } 15
         ELSE } 16
         PRINT idSetIklan
         ENDIF _____ 17
         END _____ 18
    
```

Kode Program 6.1 Pseudocode tambahSetIklan

2. Basis Path Testing

a. Flow Graph



Gambar 6.1 Flow Graph tambahSetIklan()

b. Cyclomatic Complexity

- $V(G) = 7$, ada 7 region = R1, R2, R3, R4, R5, R6, R7

- $V(G) = 23 \text{ edges} - 18 \text{ nodes} + 2 = 7$

- $V(G) = 6 \text{ predicate nodes} + 1 = 7$

c. Independen Path

- Jalur 1 = 1-2-3-5-7-9-11-13-14-15-17-18

- Jalur 2 = 1-2-3-5-7-9-11-13-**14-16-17**-18

- Jalur 3 = 1-2-3-**4-5**-7-9-11-13-14-**15-17**-18

- Jalur 4 = 1-2-3-**5-6-7**-9-11-13-14-15-17-18

- Jalur 5 = 1-2-3-5-**7-8-9**-11-13-14-15-17-18

- Jalur 6 = 1-2-3-5-7-**9-10**-11-13-14-15-17-18

- Jalur 7 = 1-2-3-5-7-**9-11-12-13**-14-15-17-18

Tabel 6.1 Hasil Pengujian Unit Pada Fungsi tambahSetIklan

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Variabel geolocation = 1, Variabel a = null, Variabel b = null, Variabel c = null, Variabel pilihanHobi = null, Variabel kebiasaan = null, Dan id = 1	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Valid
2	2	Variabel geolocation = 0, Variabel a = null,	Memperlihatkan <i>exception</i> "Error".	Memperlihatkan <i>exception</i> "Error".	Valid



No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
		Variabel b = null, Variabel c = null, Variabel pilihanHobi = null, Variabel kebiasaan = null, Dan id = 0			
3	3	Variabel geolocation = 1, Variabel a = 1, Variabel b = null, Variabel c = null, Variabel pilihanHobi = null, Variabel kebiasaan = null, Dan id = 1	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Valid
4	4	Variabel geolocation = 1, Variabel a = null, Variabel b = 1, Variabel c = null, Variabel pilihanHobi = null,	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Valid

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
		Variabel kebiasaan = null, Dan id = 1			
5	5	Variabel geolocation = 1, Variabel a = null, Variabel b = null, Variabel c = 1, Variabel pilihanHobi = null, Variabel kebiasaan = null, Dan id = 1	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Valid
6	6	Variabel geolocation = 1, Variabel a = null, Variabel b = null, Variabel c = null, Variabel pilihanHobi = 1, Variabel kebiasaan = null, Dan id = 1	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan berhasil ditambah.	Valid
7	7	Variabel geolocation = 1,	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set iklan	Menampilkan pesan "Buat Set Iklan Berhasil" dan data set	Valid

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
		Variabel a = null, Variabel b = null, Variabel c = null, Variabel pilihanHobi = null, Variabel kebiasaan = 1, Dan id = 1	berhasil ditambah.	iklan berhasil ditambah.	

6.1.2 Pengujian Unit Klas Dashboard pada Fungsi getIklanGrafik()

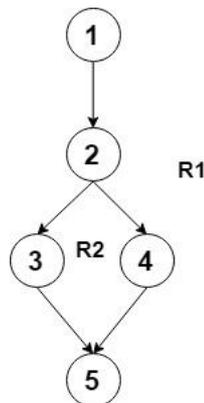
1. Pseudocode

```

1  START
2  INIT hasil = kirim idIklan ke database dashboard model _____ 1
3  IF variabel hasil terdapat isi THEN _____ 2
4      INIT array dataIklan = (namaIklan => $hasil[0]->nama_iklan,
5                          pixelId => $hasil[0]->pixel_id)
6      PRINT array("status" => TRUE, "data"=> dataIklan) } 3
7  ELSE
8      PRINT "ID IKLAN INCORRECT" } 4
9  ENDIF _____ 5
END
    
```

2. Basis Path Testing

a. Flow Graph



Gambar 6.2 Flow Graph getIklanGrafik()



b. Cyclomatic Complexity

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

c. Independen Path

- Jalur 1 = 1-2-4-5
- Jalur 2 = 1-2-3-5

Tabel 6.2 Hasil Pengujian Unit Pada Fungsi getIklanGrafik

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Variabel hasil = 0	Mencetak "ID IKLAN INCORECT".	Mencetak "ID IKLAN INCORECT".	Valid
2	2	Variabel hasil = 1	Mencetak status = true	Mencetak status = true	Valid

6.1.3 Pengujian Unit Klas Dashboard pada Fungsi tambahKampanye()

1. Pseudocode

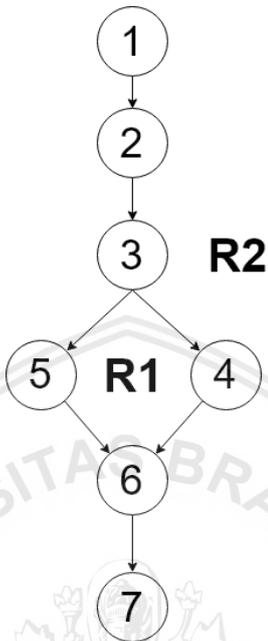
```

1  INIT sessDataFb = session(fb_access_token)
2  INIT adAccountId = ambil input dari post akunIklanForm } 1
3
4      INIT array data = (name =
5                          POST[namaKampanye],
6                          Objective = POST[tujuan],
7                          Status = "PAUSED",
8                          Access_token = token) } 2
9  INIT idKampanye = mengirimkan data ke database } 3
10 IF (terdapat id) THEN
11     INIT dataId = ambil id dari idKampanye } 4
12     Masukkan dataId dan data ke database
13 ELSE } 5
14     PRINT isi variabel idKampanye
15 END IF } 6
    END } 7
    
```



2. Basis Path Testing

a. Flow Graph



Gambar 6.3 Flow Graph tambahKampanye()

b. Cyclomatic Complexity

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

c. Independent Path

- Jalur 1 = 1-2-3-4-6-7
- Jalur 2 = 1-2-3-5-6-7

Tabel 6.3 Hasil Pengujian Unit Pada Fungsi tambahKampanye

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	variable id berisi set 1 atau true	Menampilkan pesan "Buat Kampanye Berhasil" dan data kampanye	Menampilkan pesan "Buat Kampanye Berhasil" dan data kampanye	Valid

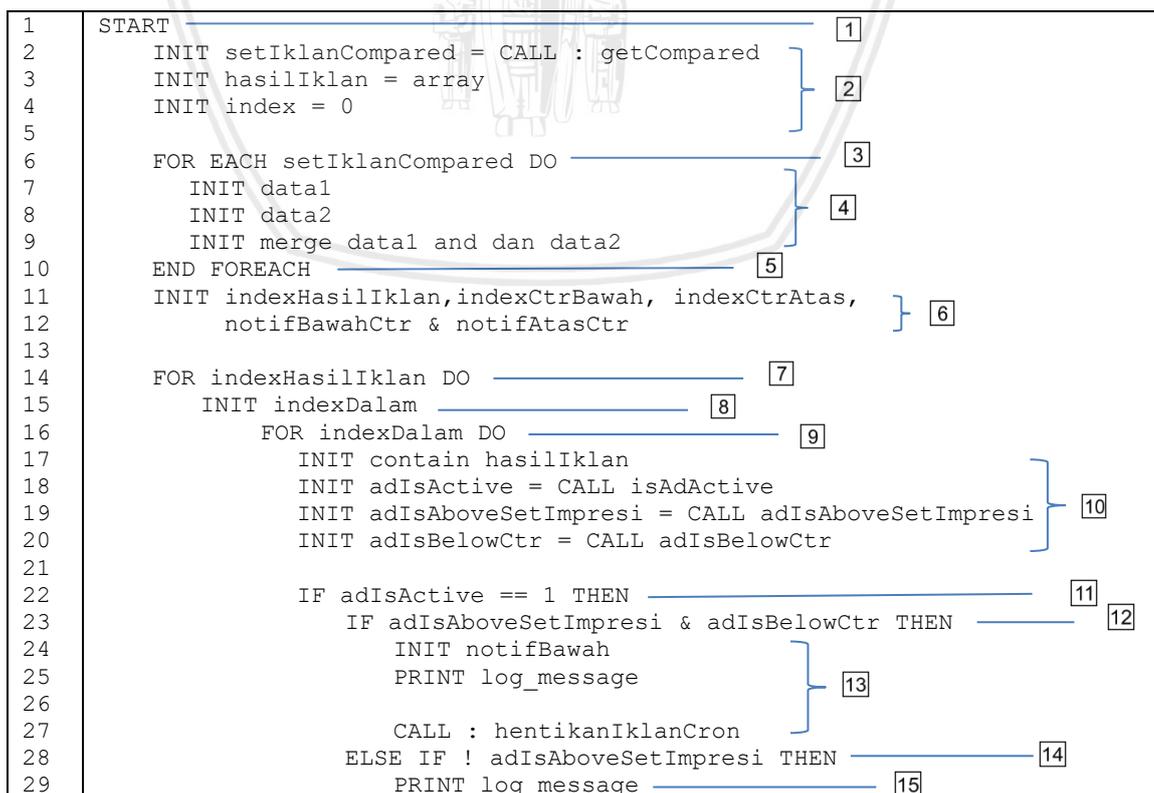
No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
			berhasil ditambah.	berhasil ditambah.	
2	2	variable id berisi set 0 atau false	Mencetak variabel idKampanye pesan kesalahan	Mencetak variabel idKampanye pesan kesalahan	Valid

6.2 Pengujian Integrasi

Pada subbab 6.2 akan dilakukan pengujian integrasi yaitu mekanisme pengujian yang dilaksanakan untuk men-test interaksi antar kelas. Untuk dapat melakukan pengujian integrasi dibutuhkan kelas lain yang dapat menjalankan atau memanggil kelas yang dimana fungsinya ingin diuji, Metode integrasi yang dipakai ialah dengan menggunakan *basis path testing* melalui *approachment top-down*. *Method* yang digunakan untuk pengujian integrasi adalah method `ctrCheck` pada kelas Dashboard.

6.2.1 Pengujian Integrasi Method `ctrCheck()` Klas Dashboard

1. Pseudocode



```

30         ELSE THEN
31             INIT notifAtas
32             PRINT log_message
33         ENDIF
34     ELSE THEN
35         PRINT log_message
36     ENDIF
37 ENDFOR
38 SEND notif(notifBawahCtr)
39 INIT notifAtasCtr = CALL sortCtrAbove(notifAtasCtr)
40 CALL getAdWinner(notifAtasCtr)
41 END

```

Diagram annotations in the code block:

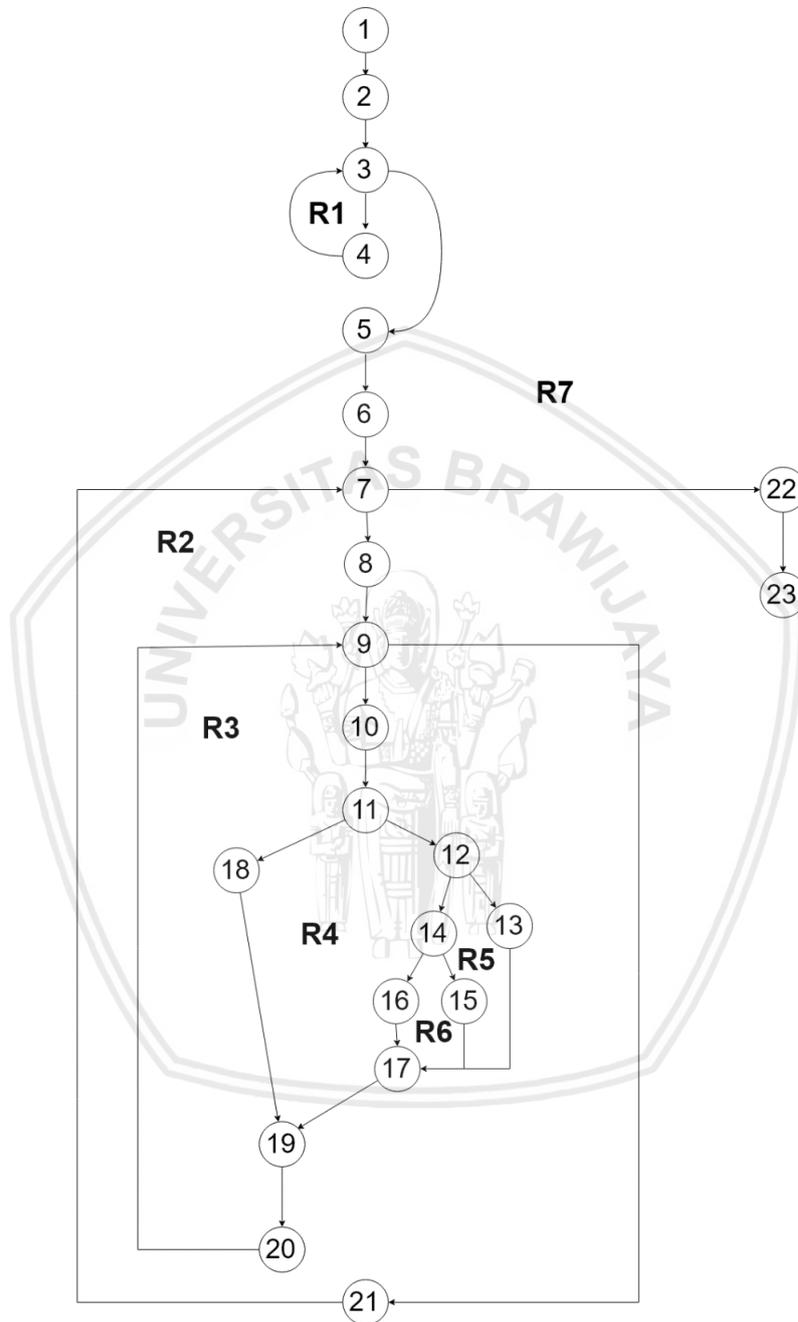
- Line 31: } 16
- Line 32: } 17
- Line 33: } 18
- Line 34: } 19
- Line 35: } 20
- Line 36: } 21
- Line 38: } 22
- Line 40: } 23

Kode Program 6.2 Pseudocode ctrCheck



2. Basis Path Testing

a. Flow Graph



Gambar 6.4 Flow Graph ctrCheck()

b. Cyclomatic Complexity

- $V(G) = 7$, ada 7 region = R1, R2, R3, R4, R5, R6, R7
- $V(G) = 28 \text{ edges} - 23 \text{ nodes} + 2 = 7$
- $V(G) = 6 \text{ predicate nodes} + 1 = 7$

c. Independen Path

- Jalur 1 = 1-2-3-5-6-7-22-23
- Jalur 2 = 1-2-3-4-3-5-6-7-22-23
- Jalur 3 = 1-2-3-4-3-5-6-7-8-9-21-7-22-23
- Jalur 4 = 1-2-3-4-3-5-6-7-8-9-10-11-18-19-20-9-21-7-22-23
- Jalur 5 = 1-2-3-4-3-5-6-7-8-9-10-11-12-13-17-19-20-9-21-7-22-23
- Jalur 6 = 1-2-3-4-3-5-6-7-8-9-10-11-12-14-15-17-19-20-9-21-7-22-23
- Jalur 7 = 1-2-3-4-3-5-6-7-8-9-10-11-12-14-16-17-19-20-9-21-7-22-23

Tabel 6.4 Hasil Pengujian Integrasi Pada Fungsi tambahSetIklan

No. Jalur	No	Prosedur Uji	Expected Result	Result	Status
1	1	Membuat method stub_sortCtrAbove dengan return null	Sistem mengambil data dari setIklanCompare d dan setiap masuk perulangan akan langsung keluar dari perulangan	Sistem mengambil data dari setIklanCompare d dan setiap masuk perulangan akan langsung keluar dari perulangan	Valid
	2	Memanggil method getAdWinner			
	3	Memanggil method ctrCheck dengan isi variabel setIklanCompared = null			
2	1	Membuat method stub_sortCtrAbove dengan return null	Sistem mengambil data dari setIklanCompare d dan masuk ke perulangan cek set iklan kemudian keluar dari perulangan.	Sistem mengambil data dari setIklanCompare d dan masuk ke perulangan cek set iklan kemudian keluar dari perulangan.	Valid
	2	Memanggil method getAdWinner			
	3	Memanggil method ctrCheck dengan isi variabel setIklanCompared = array tapi tidak tersedia iklan			
3	1	Membuat method stub_sortCtrAbove dengan return null	Sistem mengambil data dari	Sistem mengambil data dari	Valid



No. Jalur	No	Prosedur Uji	Expected Result	Result	Status
	2	Memanggil method getAdWinner	setIklanCompare d dan masuk ke perulangan cek index dalam kemudian keluar dari perulangan.	setIklanCompare d dan masuk ke perulangan cek index dalam kemudian keluar dari perulangan.	
	3	Memanggil method ctrCheck dengan isi variabel setIklanCompared = array tapi tidak tersedia isi array indexDalam			
4	1	Membuat method stub_isAdActive dengan return false	Sistem mengambil data dari setIklanCompare d berupa array sampai pada balikan method isAdActive bernilai false sehingga balikan method sortCtrAbove akan mengembalikan null.	Sistem mengambil data dari setIklanCompare d berupa array sampai pada balikan method isAdActive bernilai false sehingga balikan method sortCtrAbove akan mengembalikan null.	Valid
	2	Membuat method stub_adIsAboveImpresi dengan return false			
	3	Membuat method stub_adIsBelowCtr dengan return false			
	4	Membuat method stub_sortCtrAbove dengan return null			
	5	Memanggil method getAdWinner			
	6	Memanggil method ctrCheck dengan isi variabel setIklanCompared = array			
5	1	Membuat method stub_isAdActive dengan return true	Sistem mengambil data dari setIklanCompare d berupa array sampai pada balikan method	Sistem mengambil data dari setIklanCompare d berupa array sampai pada balikan method	Valid
	2	Membuat method stub_adIsAboveImpresi dengan return true			

No. Jalur	No	Prosedur Uji	Expected Result	Result	Status
	3	Membuat method stub_adIsBelowCtr dengan return true	adIsAboveSetImpresi dan balikan method	adIsAboveSetImpresi dan balikan method	
	4	Memanggil method hentikanIklanCron	adIsBelowCtr bernilai true maka akan	adIsBelowCtr bernilai true maka akan	
	5	Membuat method stub_sortCtrAbove dengan return null	memanggil method hentikanIklanCron untuk	memanggil method hentikanIklanCron untuk	
	6	Memanggil method getAdWinner	menghentikan iklan.	menghentikan iklan.	
	7	Memanggil method ctrCheck dengan isi variabel setIklanCompared = array			
6	1	Membuat method stub_isAdActive dengan return true	Sistem mengambil data dari	Sistem mengambil data dari	Valid
2	Membuat method stub_adIsAboveImpresi dengan return false	setIklanCompared berupa array sampai pada balikan method adIsAboveSetImpresi bernilai false maka akan	setIklanCompared berupa array sampai pada balikan method adIsAboveSetImpresi bernilai false maka akan		
3	Membuat method stub_adIsBelowCtr dengan return false	menambahkan log "Ad Below Set Impresi".	menambahkan log "Ad Below Set Impresi".		
4	Membuat method stub_sortCtrAbove dengan return null				
5	Membuat method getAdWinner				
6	Memanggil method ctrCheck dengan isi variabel setIklanCompared = array				



No. Jalur	No	Prosedur Uji	Expected Result	Result	Status
7	1	Membuat method stub_isAdActive dengan return true	Sistem mengambil data dari setIklanCompare d berupa array sampai pada balikan method adIsAboveSetImp resi bernilai true dan pada balikan method adIsBelowCtr bernilai false sehingga balikan method sortCtrAbove bernilai array.	Sistem mengambil data dari setIklanCompare d berupa array sampai pada balikan method adIsAboveSetImp resi bernilai true dan pada balikan method adIsBelowCtr bernilai false sehingga balikan method sortCtrAbove bernilai array.	Valid
	2	Membuat method stub_adIsAboveImp resi dengan true			
	3	Membuat method stub_adIsBelowCtr dengan return false			
	4	Membuat method stub_sortCtrAbove dengan return array			
	5	Memanggil method getAdWinner			
	6	Memanggil method ctrCheck dengan isi variabel setIklanCompared = array			

6.3 Pengujian Validasi

Pengujian validasi adalah pengujian yang dipakai memvalidasi bahwa fungsionalitas sistem dapat berjalan sesuai dengan harapan dan bisa berjalan dengan baik. Dalam melakukan pengujian validasi akan diterapkan seperti pada pengujian kotak hitam yaitu apa yang di dimasukkan dan apa yang dihasilkan oleh *output* dari sebuah fungsionalitas tanpa melihat kedalam kode yang digunakan. Pengujian Validasi ini akan dilakukan terhadap 52 kebutuhan fungsional.

6.3.1 Pengujian Validasi Login

1. Tabel 6.5 memperlihatkan tabel *test case* Berhasil Login.

Tabel 6.5 Kasus Uji Berhasil Login

Nama Kasus Uji	Login
Prosedur	<ol style="list-style-type: none"> 1. Membuka <i>page</i> login dengan mengetikkan alamat <i>sidamarapps.site/auth</i> kemudian enter. 2. Mengetikkan <i>usernme</i> dan <i>password</i> dengan benar. 3. Menekan tombol login.
Hasil yang diharapkan	Sistem mengecek <i>authentication</i> aktor kemudian memperlihatkan <i>page</i> dashboard dengan <i>role</i> pengunjung.
Hasil	Sistem mengecek <i>authentication</i> aktor kemudian memperlihatkan <i>page</i> utama pengunjung.
Status	Valid

2. Tabel 6.6 memperlihatkan tabel *test case* Login Alternatif 1.

Tabel 6.6 Kasus Uji Login Alternatif 1

Nama Kasus Uji	Login Alternatif 1 : Jika data <i>username</i> atau <i>password</i> tidak valid
Prosedur	<ol style="list-style-type: none"> 1. Membuka halaman <i>login</i> dengan memasukkan alamat <i>sidamarapps.site/auth</i> kemudian enter. 2. Mengetikkan <i>username</i> dan <i>password</i> tidak sesuai sebagaimana yang didaftarkan. 3. Menekan tombol <i>login</i>.
Hasil yang diharapkan	Sistem bakal memperlihatkan <i>exception error</i> "User/Password Tidak Cocok" dan memperlihatkan <i>page login</i> .
Hasil	Sistem akan menampilkan pesan error "User/Password Tidak Cocok" dan memperlihatkan <i>page login</i> .
Status	Valid

3. Tabel 6.7 memperlihatkan tabel *test case* Login Alternatif 2.

Tabel 6.7 Kasus Uji Login Alternatif 2

Nama Kasus Uji	Login Alternatif 2 : Jika data <i>username</i> atau <i>password</i> tidak terisi
Prosedur	<ol style="list-style-type: none"> 1. Aktor membuka <i>page login</i> dengan memasukkan alamat <i>sidamarapps.site/auth</i> kemudian enter. 2. Aktor mengetikkan <i>username</i> dan tidak mengisi <i>password</i>. 3. Menekan tombol <i>login</i>.
Hasil yang diharapkan	Sistem bakal menunjukkan <i>exception error</i> "Password is required" atau "Username is required"
Hasil	Sistem menampilkan pesan <i>error</i> "Password is required" atau "Username is required"
Status	Valid

6.3.2 Pengujian Validasi Lupa Password

1. Tabel 6.8 memperlihatkan tabel *test case* Berhasil Lupa Password.

Tabel 6.8 Kasus Uji Berhasil Lupa Password

Nama Kasus Uji	Lupa Password
Prosedur	<ol style="list-style-type: none"> 1. Aktor membuka halaman login dengan memasukkan alamat <i>sidamarapps.site/auth</i> kemudian <i>enter</i>. 2. Aktor menekan tulisan Lupa Password. 3. Aktor mengisi email valid. 4. Aktor menekan tombol "RESET PASSWORD".
Hasil yang diharapkan	Sistem bakal memperlihatkan pesan "Silahkan cek email kamu!" dan akan mengirimkan <i>link reset password</i> pada alamat e-mail yang terdaftar.
Hasil	Sistem bakal memperlihatkan pesan "Silahkan cek email kamu!" dan akan mengirim <i>link reset password</i> pada alamat e-mail yang terdaftar.
Status	Valid

2. Tabel 6.9 memperlihatkan tabel *test case* Lupa Password Alternatif 1

Tabel 6.9 Kasus Uji Lupa Password Alternatif 1

Nama Kasus Uji	Lupa Password Alternatif 1: Jika form e-mail kosong dan menekan tombol dapatkan password.
Prosedur	<ol style="list-style-type: none"> 1. Membuka halaman login dengan memasukkan alamat sidamarapps.site/auth kemudian enter. 2. Menenekan tulisan Lupa Password. 3. Membiarkan kosong <i>form</i> e-mail. 4. Aktor menekan "RESET PASSWORD".
Hasil yang diharapkan	Sisttem memperlihatkan <i>exception error</i> "Email Tidak Terdaftar!".
Hasil	Sisttem memperlihatkan <i>exception error</i> "Email Tidak Terdaftar!".
Status	Valid

3. Tabel 6.10 memperlihatkan tabel *test case* Lupa Password Alternatif 2

Tabel 6.10 Kasus Uji Lupa Password Alternatif 2

Nama Kasus Uji	Lupa Password Alternatif 2: Jika menekan tombol Batal.
Prosedur	<ol style="list-style-type: none"> 1. Membuka <i>page login</i> dengan memasukkan alamat sidamarapps.site/auth kemudian enter. 2. Menekan tulisan Lupa Password. 3. Mengetikkan e-mail. 4. Menekan tombol "X".
Hasil yang diharapkan	Sisttem memperlihatkan <i>page login</i> .
Hasil	Sisttem akan memperlihatkan <i>page login</i> .
Status	Valid

4. Tabel 6.11 memperlihatkan tabel *test case* Lupa Password Alternatif 3

Tabel 6.11 Kasus Uji Lupa Password Alternatif 3

Nama Kasus Uji	Lupa Password Alternatif 2: Jika aktor memasukkan email yang tidak terdaftar sebelumnya atas sistem.
Prosedur	<ol style="list-style-type: none"> 1. Membuka <i>page login</i> dengan mengetikkan alamat <i>sidamarapps.site/auth</i> kemudian enter. 2. Menekan tulisan Lupa Pasword. 3. Mengetikkan e-mail yang tidak terdaftar. 4. Menekan tombol "RESET PASSWORD".
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> "Email Tidak Terdaftar!".
Hasil	Sistem memperlihatkan <i>exception error</i> "Email Tidak Terdaftar!".
Status	Valid

6.3.3 Pengujian Validasi Pendaftaran

1. Tabel 6.12 memperlihatkan tabel *test case* Berhasil Melakukan Pendaftaran

Tabel 6.12 Kasus Uji Berhasil Melakukan Pendaftaran

Nama Kasus Uji	Pendaftaran
Prosedur	<ol style="list-style-type: none"> 1. Membuka <i>page login</i> dengan memasukkan alamat <i>sidamarapps.site/auth</i> kemudian enter. 2. Menekan tulisan "Sign up now". 3. Mengetikkan <i>Full Name, Username, Password, Re-type Password</i> dan <i>Email</i> pada form yang disediakan. 4. Menekan tombol Submit.
Hasil yang diharapkan	Sistem berhasil menyimpan dan mengalihkan ke <i>page login</i> dengan pesan "Sukses Terdaftar".
Hasil	Sistem berhasil menyimpan dan mengalihkan ke <i>page login</i> dengan pesan "Sukses Terdaftar".
Status	Valid

2. Tabel 6.13 memperlihatkan tabel *test case* Melakukan Pendaftaran Alternatif 1

Tabel 6.13 Kasus Uji Melakukan Pendaftaran Alternatif 1

Nama Kasus Uji	Pendaftaran Alternatif 1: Jika aktor tidak mengisi form input Email.
Prosedur	<ol style="list-style-type: none"> 1. Membuka <i>page</i> login dengan memasukkan alamat <i>sidamarapps.site/auth</i> kemudian enter. 2. Menekan tombol "Sign up now". 3. Mengetikkan <i>Full Name, Username, Password, Re-type Password</i> pada form yang disediakan dan mengosongkan Email. 4. Menekan tombol "Submit".
Hasil yang diharapkan	Sistem akan menunjukkan <i>exception error</i> "Please Fill Out This Field".
Hasil	Sistem akan menunjukkan <i>exception error</i> "Please Fill Out This Field".
Status	Valid

6.3.4 Pengujian Validasi Melihat *Preface*

1. Tabel 6.14 memperlihatkan tabel *test case* Berhasil Melihat *Preface*

Tabel 6.14 Kasus Uji Berhasil Melihat *Preface*

Nama Kasus Uji	Melihat <i>Preface</i>
Prosedur	Menekan tombol " <i>preface</i> " pada menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan <i>page</i> pengunjung.
Hasil	Sistem memperlihatkan <i>page</i> pengunjung.
Status	Valid

6.3.5 Pengujian Validasi Melihat 3 Komponen Iklan Facebook

1. Tabel 6.15 memperlihatkan tabel *test case* Berhasil Melihat 3 Komponen Iklan Facebook

Tabel 6.15 Kasus Uji Berhasil Melihat 3 Komponen Iklan Facebook

Nama Kasus Uji	Melihat 3 Komponen Iklan Facebook
Prosedur	Menekan tombol " <i>3 Part of Facebook Ads</i> " pada menu <i>sidebar</i> .

Hasil yang diharapkan	Sistem memperlihatkan pilihan kampanye, set iklan serta iklan.
Hasil	Sistem memperlihatkan pilihan kampanye, set iklan serta iklan.
Status	Valid

6.3.6 Pengujian Validasi Melihat ROAS dan ROMI

1. Tabel 6.16 memperlihatkan tabel *test case* Berhasil Melihat ROAS & ROMI

Tabel 6.16 Kasus Uji Berhasil Melihat ROAS dan ROMI

Nama Kasus Uji	Melihat Informasi Melihat ROAS dan ROMI
Prosedur	Menekan tombol " <i>Introduce</i> ROAS dan ROMI" pada menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan page <i>measure</i> .
Hasil	Sistem memperlihatkan page <i>measure</i> .
Status	Valid

6.3.7 Pengujian Validasi Melihat *About* SIDAMAR

1. Tabel 6.17 memperlihatkan tabel *test case* Berhasil Melihat Informasi *About* SIDAMAR

Tabel 6.17 Kasus Uji Berhasil Melihat Informasi *About* SIDAMAR

Nama Kasus Uji	Melihat Informasi <i>About</i> SIDAMAR
Prosedur	Menekan tombol " <i>About</i> SIDAMAR" pada menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan page <i>About</i> .
Hasil	Sistem memperlihatkan page <i>About</i> .
Status	Valid

6.3.8 Pengujian Validasi Melihat Dokumentasi

1. Tabel 6.18 memperlihatkan tabel *test case* Berhasil Melihat Dokumentasi

Tabel 6.18 Kasus Uji Berhasil Melihat Dokumentasi

Nama Kasus Uji	Melihat Dokumentasi
Prosedur	Menekan tombol " <i>Complete Documentation</i> " pada menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan <i>page</i> dokumentasi.
Hasil	Sistem memperlihatkan <i>page</i> dokumentasi.
Status	Valid

6.3.9 Pengujian Validasi Login Facebook

1. Tabel 6.19 memperlihatkan tabel *test case* Berhasil Login Facebook

Tabel 6.19 Kasus Uji Berhasil Login Facebook

Nama Kasus Uji	Login Facebook
Prosedur	<ol style="list-style-type: none"> 1. Membuka <i>page</i> "<i>preface</i>" pada menu <i>sidebar</i>. 2. Menekan tombol "Masuk Dengan Facebook". 3. Mengetikkan <i>username</i> serta <i>password</i> akun facebook. 4. Menekan tombol "Masuk". 5. Menekan tombol "Izinkan".
Hasil yang diharapkan	Sistem mengecek autentifikasi kemudian mengarahkan ke dashboard dengan status sebagai marketer.
Hasil	Sistem akan mengecek autentifikasi dan mengarahkan ke dashboard dengan status sebagai marketer.
Status	Valid

2. Tabel 6.20 memperlihatkan tabel *test case* Login Facebook Alternatif 1

Tabel 6.20 Kasus Uji Login Facebook Alternatif 1

Nama Kasus Uji	Login Facebook Alternatif 1: Ketika aktor salah memasukkan <i>username</i> atau <i>password</i> .
Prosedur	1. Membuka <i>page</i> utama aktor pengunjung.

	<ol style="list-style-type: none"> 2. Menekan tombol “Login Menggunakan Facebook”. 3. Mengetikkan <i>username</i> serta <i>password</i> akun facebook yang tidak valid. 4. Menekan tombol “Masuk”.
Hasil yang diharapkan	Sistem mengecek autentifikasi kemudian memperlihatkan <i>exception error</i> .
Hasil	Sistem mengecek autentifikasi kemudian memperlihatkan <i>exception error</i> .
Status	Valid

6.3.10 Pengujian Validasi Menambah Kampanye

1. Tabel 6.21 memperlihatkan tabel *test case* Berhasil Menambah Kampanye

Tabel 6.21 Kasus Uji Berhasil Menambah Kampanye

Nama Kasus Uji	Menambah Kampanye
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol kampanye pada menu <i>sidebar</i>. 2. Memilih akun iklan yang digunakan untuk menambahkan kampanye pada <i>dropdown</i> Pilih Akun Iklan. 3. Menekan tombol “Tambah Kampanye”. 4. Mengetikkan pada form tambah kampanye yaitu nama kampanye dan memilih tujuan iklan dalam kampanye yang dibuat pada <i>dropdown</i>. 5. Menekan tombol “Save Campaign”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Kampanye Berhasil Dibuat” kemudian memperlihatkan hasil tambah kampanye pada tabel daftar kampanye.
Hasil	Sistem memperlihatkan <i>alert</i> “Kampanye Berhasil Dibuat” kemudian memperlihatkan hasil tambah kampanye pada tabel daftar kampanye.
Status	Valid

2. Tabel 6.22 memperlihatkan tabel *test case* Menambah Kampanye Alternatif 1

Tabel 6.22 Kasus Uji Menambah Kampanye Alternatif 1

Nama Kasus Uji	Menambah Kampanye Alternatif 1: Ketika terdapat form input dalam kondisi <i>empty</i> .
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol kampanye di menu <i>sidebar</i>. 2. Memilih akun iklan yang digunakan untuk menambahkan kampanye pada <i>dropdown</i> Pilih Akun Iklan. 3. Menekan tombol "Tambah Kampanye". 4. Mengosongkan form nama kampanye dan memilih <i>dropdown</i> Tujuan Beriklan. 5. Menekan tombol "Save Campaign".
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Hasil	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

3. Tabel 6.23 memperlihatkan tabel *test case* Menambah Kampanye Alternatif 2

Tabel 6.23 Kasus Uji Menambah Kampanye Alternatif 2

Nama Kasus Uji	Menambah Kampanye Alternatif 2: Jika menekan tombol "Close".
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol Kampanye di menu <i>sidebar</i>. 2. Memilih akun iklan yang digunakan untuk menambahkan kampanye pada <i>dropdown</i> Pilih Akun Iklan. 3. Menekan tombol "Tambah Kampanye". 4. Mengetikkan form tambah kampanye. 5. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal tambah kampanye yang telah ditampilkan.
Hasil	Sistem menutup modal tambah kampanye yang telah ditampilkan.
Status	Valid

4. Tabel 6.24 memperlihatkan tabel *test case* Menambah Kampanye Alternatif 3

Tabel 6.24 Kasus Uji Menambah Kampanye Alternatif 3

Nama Kasus Uji	Menambah Kampanye Alternatif 3: Jika menekan tombol "X".
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol Kampanye di menu <i>sidebar</i>. 2. Memilih akun iklan yang digunakan untuk menambahkan kampanye pada <i>dropdown</i> Pilih Akun Iklan. 3. Menekan tombol "Tambah Kampanye". 4. Mengetikkan form tambah kampanye. 5. Menekan tombol "X".
Hasil yang diharapkan	Sistem menutup modal tambah kampanye yang telah ditampilkan.
Hasil	Sistem menutup modal tambah kampanye yang telah ditampilkan.
Status	Valid

6.3.11 Pengujian Validasi Melihat Kampanye

1. Tabel 6.25 memperlihatkan tabel *test case* Berhasil Menampilkan Data Seluruh Kampanye

Tabel 6.25 Kasus Uji Berhasil Menampilkan Data Seluruh Kampanye

Nama Kasus Uji	Menampilkan Data Seluruh Kampanye.
Prosedur	Menekan tombol Kampanye di menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan seluruh kampanye.
Hasil	Sistem memperlihatkan seluruh kampanye.
Status	Valid

6.3.12 Pengujian Validasi Mengedit Kampanye

1. Tabel 6.26 memperlihatkan tabel *test case* Berhasil Mengedit Kampanye

Tabel 6.26 Kasus Uji Berhasil Mengedit Kampanye

Nama Kasus Uji	Mengedit Kampanye
Prosedur	1. Menekan tombol Kampanye di menu <i>sidebar</i> .

	<ol style="list-style-type: none"> 2. Menekan tombol “edit kampanye” pada kampanye yang diinginkan. 3. Mengedit pada form edit kampanye. 4. Menekan tombol “Save Changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> kampanye berhasil di update kemudian memperlihatkan pada daftar kampanye.
Hasil	Sistem memperlihatkan <i>alert</i> kampanye berhasil di update kemudian memperlihatkan pada daftar kampanye.
Status	Valid

2. Tabel 6.27 memperlihatkan tabel *test case* Mengedit Kampanye Alternatif 1

Tabel 6.27 Kasus Uji Mengedit Kampanye Alternatif 1

Nama Kasus Uji	Mengedit Kampanye Alternatif 1: Jika mengosongkan salah satu form edit kampanye.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol Kampanye di menu <i>sidebar</i>. 2. Menekan tombol “edit kampanye” pada kampanye yang diinginkan. 3. Mengosongkan nama kampanye pada form kampanye. 4. Menekan tombol “Save Changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Hasil	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Status	Valid

2. Tabel 6.28 memperlihatkan tabel *test case* Mengedit Kampanye Alternatif 2

Tabel 6.28 Kasus Uji Mengedit Kampanye Alternatif 2

Nama Kasus Uji	Mengedit Kampanye Alternatif 2: Ketika menekan tombol “Close”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol Kampanye di menu <i>sidenar</i>. 2. Memilih akun iklan yang digunakan untuk menambahkan kampanye pada <i>dropdown</i> Pilih Akun Iklan.

	<ol style="list-style-type: none"> 3. Menekan tombol “edit” pada tabel. 4. Mengetikkan form edit kampanye. 5. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal edit kampanye.
Hasil	Sistem menutup modal edit kampanye.
Status	Valid

6.3.13 Pengujian Validasi Menghapus Kampanye

1. Tabel 6.29 memperlihatkan tabel *test case* Berhasil Menghapus Kampanye

Tabel 6.29 Kasus Uji Berhasil Menghapus Kampanye

Nama Kasus Uji	Menghapus Kampanye
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol Kampanye di menu <i>sidebar</i>. 2. Menekan tombol “Delete” pada kampanye yang dihapus.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Kampanye berhasil dihapus”.
Hasil	Sistem memperlihatkan <i>alert</i> “Kampanye berhasil dihapus”.
Status	Valid

2. Tabel 6.30 memperlihatkan tabel *test case* Menghapus Kampanye Alternatif 1

Tabel 6.30 Kasus Uji Menghapus Kampanye Alternatif 1

Nama Kasus Uji	Menghapus Kampanye alternatif 1: menekan tombol tidak jadi.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol Kampanye di menu <i>sidebar</i>. 2. Menekan tombol “Delete” di pilihan kampanye 3. Menekan tombol “Tidak Jadi!”.
Hasil yang diharapkan	Sistem menutup <i>alert</i> hapus kampanye.
Hasil	Sistem menutup <i>alert</i> hapus kampanye.
Status	Valid

6.3.14 Pengujian Validasi Memulai Kampanye

1. Tabel 6.31 memperlihatkan tabel *test case* Berhasil Memulai Kampanye

Tabel 6.31 Kasus Uji Berhasil Memulai Kampanye

Nama Kasus Uji	Memulai Kampanye
Prosedur	1. Menekan tombol Kampanye di menu <i>sidebar</i> . 2. Menekan tombol “Mulai Kampanye” pada kampanye yang ingin diaktifkan.
Hasil yang diharapkan	Sistem merubah keterangan <i>paused</i> jadi <i>active</i>
Hasil	Sistem merubah keterangan <i>paused</i> jadi <i>active</i>
Status	Valid

6.3.15 Pengujian Validasi Menjeda Kampanye

1. Tabel 6.32 memperlihatkan tabel *test case* Berhasil Menjeda Kampanye

Tabel 6.32 Kasus Uji Berhasil Menjeda Kampanye

Nama Kasus Uji	Menjeda Kampanye
Prosedur	1. Menekan tombol Kampanye di menu <i>sidebar</i> . 2. Menekan tombol “Jeda Kampanye” pada kampanye yang ingin dijeda.
Hasil yang diharapkan	Sistem merubah keterangan <i>active</i> jadi <i>paused</i> .
Hasil	Sistem merubah keterangan <i>active</i> jadi <i>paused</i> .
Status	Valid

6.3.16 Pengujian Validasi Menambah Set Iklan

1. Tabel 6.33 memperlihatkan tabel *test case* Berhasil Menambah Set Iklan

Tabel 6.33 Kasus Uji Berhasil Menambah Set Iklan

Nama Kasus Uji	Menambah Set Iklan
Prosedur	1. Menekan tombol “Set Iklan” di menu <i>sidebar</i> . 2. Menekan tombol “Tambah Set Iklan”. 3. Mengetikkan form set iklan secara lengkap. 4. Menekan tombol “Save Adset”.
Hasil yang diharapkan	Sistem <i>memperlihatkan alert</i> “Set Iklan berhasil ditambahkan ke kampanye”.

Hasil	Sistem <i>memperlihatkan alert</i> "Set Iklan berhasil ditambahkan ke kampanye".
Status	Valid

2. Tabel 6.34 memperlihatkan tabel *test case* Menambah Set Iklan Alternatif 1

Tabel 6.34 Kasus Uji Menambah Set Iklan Alternatif 1

Nama Kasus Uji	Menambah Set Iklan Alternatif 1: Terdapat salah satu form yang kosong.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Set Iklan" pada menu. 2. Menekan tombol "Tambah Set Iklan". 3. Mengetikkan form set iklan dan mengosongkan form nama set iklan. 4. Menekan tombol "Save Adset".
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Hasil	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

3. Tabel 6.35 memperlihatkan tabel *test case* Menambah Set Iklan Alternatif 2

Tabel 6.35 Kasus Uji Menambah Set Iklan Alternatif 2

Nama Kasus Uji	Menambah Set Iklan Alternatif 2: Ketika menekan tombol "Close".
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Set Iklan" di menu <i>sidebar</i>. 2. Memilih akun iklan yang digunakan untuk menambahkan kampanye pada <i>dropdown</i> Pilih Akun Iklan. 3. Menekan tombol "Tambah Set Iklan". 4. Mengetikkan form tambah set iklan. 5. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal tambah set iklan.
Hasil	Sistem menutup modal tambah set iklan.
Status	Valid

6.3.17 Pengujian Validasi Melihat Set Iklan

1. Tabel 6.36 memperlihatkan tabel *test case* Berhasil Melihat Set Iklan.

Tabel 6.36 Kasus Uji Berhasil Melihat Set Iklan

Nama Kasus Uji	Melihat Set Iklan
Prosedur	Menekan tombol “Set Iklan” di menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan seluruh set iklan di tabel.
Hasil	Sistem memperlihatkan seluruh set iklan di tabel.
Status	Valid

6.3.18 Pengujian Validasi Mengedit Set Iklan

1. Tabel 6.37 memperlihatkan tabel *test case* Berhasil Mengedit Set Iklan.

Tabel 6.37 Kasus Uji Berhasil Mengedit Set Iklan

Nama Kasus Uji	Mengedit Set Iklan
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Set Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “/” pada set iklan yang dipilih. 3. Mengedit form set iklan. 4. Menekan tombol “Save changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Update Set Iklan Berhasil”.
Hasil	Sistem memperlihatkan <i>alert</i> “Update Set Iklan Berhasil”.
Status	Valid

2. Tabel 6.38 memperlihatkan tabel *test case* Mengedit Set Iklan Alternatif 1.

Tabel 6.38 Kasus Uji Mengedit Set Iklan Alternatif 1

Nama Kasus Uji	Mengedit Set Iklan Alternatif 1: Mengosongkan beberapa form set iklan.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Set Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “/” pada set iklan yang dipilih. 3. Mengedit form set iklan dan mengosongkan form nama set iklan. 4. Menekan tombol “Save changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.

Hasil	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

3. Tabel 6.39 memperlihatkan tabel *test case* Mengedit Set Iklan Alternatif 2.

Tabel 6.39 Kasus Uji Mengedit Set Iklan Alternatif 2

Nama Kasus Uji	Mengedit Set Iklan Alternatif 2: Jika mengklik tombol "Close".
Prosedur	1. Menekan tombol "Set Iklan" di menu <i>sidebar</i> . 2. Menekan tombol "/" pada set iklan yang dipilih. 3. Mengetikkan form Edit set iklan. 5. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal Edit set iklan.
Hasil	Sistem menutup modal Edit set iklan.
Status	Valid

6.3.19 Pengujian Validasi Melihat Detail Set Iklan

1. Tabel 6.40 memperlihatkan tabel *test case* Berhasil Melihat Detail Set Iklan.

Tabel 6.40 Kasus Uji Berhasil Melihat Detail Set Iklan

Nama Kasus Uji	Melihat Detail Set Iklan
Prosedur	1. Menekan tombol "Set Iklan" di menu <i>sidebar</i> . 2. Menekan tombol "Detail" set iklan pilihan.
Hasil yang diharapkan	Sistem memperlihatkan details set iklan.
Hasil	Sistem memperlihatkan details set iklan.
Status	Valid

6.3.20 Pengujian Validasi Menghapus Set Iklan

1. Tabel 6.41 memperlihatkan tabel *test case* Berhasil Menghapus Set Iklan.

Tabel 6.41 Kasus Uji Berhasil Menghapus Set Iklan

Nama Kasus Uji	Menghapus Set Iklan
Prosedur	1. Menekan tombol "Set Iklan" di menu <i>sidebar</i> .

	<ol style="list-style-type: none"> 2. Menekan tombol “X” pada Set Iklan yang ingin dihapus. 3. Menekan tombol “Ya, Tentu!”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Set Iklan berhasil dihapus”.
Hasil	Sistem memperlihatkan <i>alert</i> “Set Iklan berhasil dihapus”.
Status	Valid

2. Tabel 6.42 memperlihatkan tabel *test case* Menghapus Set Iklan Alternatif 1.

Tabel 6.42 Kasus Uji Menghapus Set Iklan Alternatif 1

Nama Kasus Uji	Menghapus Kampanye Alternatif 1: menekan tombol tidak jadi.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Set Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “X” pada set iklan yang ingin dihapus. 3. Menekan tombol “Tidak Jadi!”.
Hasil yang diharapkan	Sistem menutup <i>alert</i> hapus set iklan.
Hasil	Sistem menutup <i>alert</i> hapus set iklan.
Status	Valid

6.3.21 Pengujian Validasi Memulai Set Iklan

1. Tabel 6.43 memperlihatkan tabel *test case* Berhasil Memulai Set Iklan Kasus Uji.

Tabel 6.43 Kasus Uji Berhasil Memulai Set Iklan

Nama Kasus Uji	Memulai Set Iklan
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Set Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “Mulai Set Iklan” pada set iklan yang ingin diaktifkan.
Hasil yang diharapkan	Sistem merubah keterangan <i>paused</i> jadi <i>active</i> .
Hasil	Sistem merubah keterangan <i>paused</i> jadi <i>active</i> ..
Status	Valid

6.3.22 Pengujian Validasi Menjeda Set Iklan

1. Tabel 6.44 memperlihatkan tabel *test case* Berhasil Menjeda Set Iklan

Tabel 6.44 Kasus Uji Berhasil Menjeda Set Iklan

Nama Kasus Uji	Menjeda Set Iklan
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Set Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “Jeda Set Iklan” pada set iklan yang ingin dijeda. 3. Menekan tombol “Ya, Tentu!”.
Hasil yang diharapkan	Sistem merubah keterangan <i>active</i> jadi <i>paused</i> .
Hasil	Sistem merubah keterangan <i>active</i> jadi <i>paused</i> .
Status	Valid

6.3.23 Pengujian Validasi Menambah Iklan

1. Tabel 6.45 memperlihatkan tabel *test case* Berhasil Menambahkan Iklan

Tabel 6.45 Kasus Uji Berhasil Menambah Iklan

Nama Kasus Uji	Menambah Iklan
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “Tambah Iklan”. 3. Mengetikkan form tambah iklan secara lengkap. 4. Menekan tombol “Save Ad”.
Hasil yang diharapkan	Sistem memperlihatkan alert “Buat Iklan Berhasil”.
Hasil	Sistem memperlihatkan alert “Buat Iklan Berhasil”.
Status	Valid

2. Tabel 6.46 memperlihatkan tabel *test case* Menambahkan Iklan Alternatif 1

Tabel 6.46 Kasus Uji Menambah Iklan Alternatif 1

Nama Kasus Uji	Menambah Iklan Alternatif 1: Terdapat salah satu form yang kosong.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Iklan” di menu <i>sidebar</i>. 2. Menekan tombol “Tambah Iklan”. 3. Mengetikkan form tambah iklan dan mengosongkan form nama iklan.

	4. Menekan tombol "Save Ad".
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Hasil	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

3. Tabel 6.47 memperlihatkan tabel *test case* Menambahkan Iklan Alternatif 2

Tabel 6.47 Kasus Uji Menambah Iklan Alternatif 2

Nama Kasus Uji	Menambah Iklan Alternatif 2: Ketika menekan tombol "Close".
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Iklan" di menu <i>sidebar</i>. 2. Memilih akun iklan pada <i>dropdown</i> Pilih Akun Iklan. 3. Menekan tombol "Tambah Iklan". 4. Mengetikkan form tambah iklan. 5. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal tambah iklan.
Hasil	Sistem menutup modal tambah iklan.
Status	Valid

6.3.24 Pengujian Validasi Melihat Iklan

1. Tabel 6.48 memperlihatkan tabel *test case* Berhasil Melihat Iklan.

Tabel 6.48 Kasus Uji Berhasil Melihat Iklan

Nama Kasus Uji	Melihat Iklan
Prosedur	Menekan tombol "Iklan" di menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan seluruh iklan di tabel.
Hasil	Sistem memperlihatkan seluruh iklan di tabel.
Status	Valid

6.3.25 Pengujian Validasi Memulai Iklan

1. Tabel 6.49 memperlihatkan tabel *test case* Berhasil Memulai Set Iklan.

Tabel 6.49 Kasus Uji Berhasil Memulai Iklan

Nama Kasus Uji	Memulai Iklan
Prosedur	1. Menekan tombol “Iklan” di menu <i>sidebar</i> . 2. Menekan tombol “Mulai Iklan” pada iklan yang ingin diaktifkan.
Hasil yang diharapkan	Sistem merubah keterangan <i>paused</i> jadi <i>active</i> .
Hasil	Sistem merubah keterangan <i>paused</i> jadi <i>active</i> .
Status	Valid

6.3.26 Pengujian Validasi Menjeda Iklan

1. Tabel 6.50 memperlihatkan tabel *test case* Berhasil Memulai Menjeda Iklan.

Tabel 6.50 Kasus Uji Berhasil Menjeda Iklan

Nama Kasus Uji	Menjeda Iklan
Prosedur	1. Menekan tombol “Iklan” di menu <i>sidbar</i> . 2. Menekan tombol “Jeda Iklan” pada iklan yang ingin dijeda. 3. Menekan tombol “Ya, Tentu!”.
Hasil yang diharapkan	Sistem merubah keterangan <i>active</i> jadi <i>paused</i> .
Hasil	Sistem merubah keterangan <i>active</i> jadi <i>paused</i> .
Status	Valid

6.3.27 Pengujian Validasi Mengedit Iklan

1. Tabel 6.51 memperlihatkan tabel *test case* Berhasil Mengedit Iklan.

Tabel 6.51 Kasus Uji Berhasil Mengedit Iklan

Nama Kasus Uji	Mengedit Iklan
Prosedur	1. Menekan tombol “Iklan” di menu <i>sidebar</i> . 2. Menekan tombol “edit iklan” pada iklan yang dipilih. 3. Mengedit form iklan. 4. Menekan tombol “Save changes”.
Hasil yang diharapkan	Sistem memperlihatkan alert “Update Iklan Berhasil”.

Hasil	Sistem memperlihatkan alert "Update Iklan Berhasil".
Status	Valid

2. Tabel 6.52 memperlihatkan tabel *test case* Mengedit Iklan Alternatif 1.

Tabel 6.52 Kasus Uji Mengedit Iklan Alternatif 1

Nama Kasus Uji	Mengedit Iklan Alternatif 1: Mengosongkan beberapa form iklan.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Iklan" di menu <i>sidebar</i>. 2. Menekan tombol "edit iklan" pada iklan yang dipilih. 3. Mengedit form iklan dan mengosongkan form nama iklan. 4. Menekan tombol "Save changes".
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Hasil	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

3. Tabel 6.53 memperlihatkan tabel *test case* Mengedit Iklan Alternatif 2.

Tabel 6.53 Kasus Uji Mengedit Iklan Alternatif 2

Nama Kasus Uji	Mengedit Iklan Alternatif 2: Jika menekan tombol "Close".
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Iklan" di menu <i>sidebar</i>. 2. Menekan tombol "Edit Iklan" pada iklan yang dipilih. 3. Mengetikkan form Edit iklan. 5. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal Edit iklan.
Hasil	Sistem menutup modal Edit iklan.
Status	Valid

6.3.28 Pengujian Validasi Menghapus Iklan.

1. Tabel 6.54 memperlihatkan tabel *test case* Berhasil Menghapus Iklan.

Tabel 6.54 Kasus Uji Berhasil Menghapus Iklan

Nama Kasus Uji	Menghapus Iklan
Prosedur	1. Menekan tombol “Iklan” di menu <i>sidebar</i> . 2. Menekan tombol “Hapus Iklan” pada Set Iklan yang ingin dihapus.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Iklan berhasil dihapus”.
Hasil	Sistem memperlihatkan <i>alert</i> “Iklan berhasil dihapus”.
Status	Valid

2. Tabel 6.55 memperlihatkan tabel *test case* Menghapus Iklan Alternatif 1.

Tabel 6.55 Kasus Uji Menghapus Iklan Alternatif 1

Nama Kasus Uji	Menghapus Iklan Alternatif 1: menekan tombol tidak jadi.
Prosedur	1. Menekan tombol “Iklan” di menu <i>sidebar</i> . 2. Menekan tombol “X” di iklan yang ingin dihapus. 3. Menekan tombol “Tidak Jadi!”.
Hasil yang diharapkan	Sistem menutup <i>alert</i> hapus iklan.
Hasil	Sistem menutup <i>alert</i> hapus iklan.
Status	Valid

6.3.29 Pengujian Validasi Menambah Ad Creative

1. Tabel 6.56 memperlihatkan tabel *test case* Berhasil Menambah Ad Creative.

Tabel 6.56 Kasus Uji Berhasil Menambah Ad Creative

Nama Kasus Uji	Menambah Ad Creative
Prosedur	1. Menekan tombol “Ad Creative” di menu <i>sidebar</i> . 2. Menekan tombol “Tambah Ad Creative”. 3. Mengetikkan form tambah Ad Creative secara lengkap. 4. Menekan tombol “Save Creative”.

Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> "Buat Iklan Kreatif Berhasil".
Hasil	Sistem memperlihatkan <i>alert</i> "Buat Iklan Kreatif Berhasil".
Status	Valid

2. Tabel 6.57 memperlihatkan tabel *test case* Menambah Ad Creative Alternatif 1.

Tabel 6.57 Kasus Uji Menambah Ad Creative Alternatif 1

Nama Kasus Uji	Menambah Iklan Ad Creative 1: Terdapat salah satu form yang kosong.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Ad Creative" di menu <i>sidebar</i>. 2. Menekan tombol "Tambah Ad Creative". 3. Mengetikkan form tambah ad creative dan mengosongkan beberapa form ad creative. 4. Menekan tombol "Save Creative".
Hasil yang diharapkan	Sistem meperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Hasil	Sistem meperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

6.3.30 Pengujian Validasi Melihat Ad Creative

1. Tabel 6.58 memperlihatkan tabel *test* Berhasil Melihat Ad Creative.

Tabel 6.58 Kasus Uji Berhasil Melihat Ad Creative

Nama Kasus Uji	Melihat Ad Creative
Prosedur	Menekan tombol "Ad Creative" di menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan seluruh Ad Creative dalaam tabel.
Hasil	Sistem memperlihatkan seluruh Ad Creative dalaam tabel.
Status	Valid

6.3.31 Pengujian Validasi Memotong Gambar

1. Tabel 6.59 memperlihatkan tabel *test* Berhasil Memotong Gambar.

Tabel 6.59 Kasus Uji Berhasil Memotong Gambar

Nama Kasus Uji	Memotong Gambar
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Creative” di menu <i>sidebar</i>. 2. Menekan tombol “Crop Image”. 3. Menekan tombol “browse” kemudian pilih gambar yang akan dipotong. 4. Memilih area yang akan dipotong. 5. Menekan tombol “Crop”. 4. Menekan tombol “Download”.
Hasil yang diharapkan	Sistem memperlihatkan gambar terbaru sesuai dengan hasil potongan aktor.
Hasil	Sistem memperlihatkan gambar terbaru sesuai dengan hasil potongan aktor.
Status	Valid

2. Tabel 6.60 memperlihatkan tabel *test* Memotong Gambar Alternatif 1.

Tabel 6.60 Kasus Uji Memotong Gambar Alternatif 1

Nama Kasus Uji	Memotong Gambar Alternatif 1: Jika menekan tombol “close”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Creative” di menu <i>sidebar</i>. 2. Menekan tombol “Crop Image”. 3. Memilih gambar yang akan dipotong dengan menekan tombol “Browse”. 4. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal Potong Gambar.
Hasil	Sistem menutup modal Potong Gambar.
Status	Valid

6.3.32 Pengujian Validasi Melihat Preview Ad Creative

1. Tabel 6.61 memperlihatkan tabel *test* Berhasil Melihat Preview Ad Creative.

Tabel 6.61 Kasus Uji Berhasil Melihat Preview Ad Creative

Nama Kasus Uji	Melihat Preview Ad Creative
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Creative” di menu <i>sidebar</i>. 2. Menekan tombol “Preview” di ad creative yang diinginkan. 3. Memilih jenis preview yang diinginkan.
Hasil yang diharapkan	Sistem memperlihatkan <i>preview</i> ad creative sesuai dengan pilihan aktor.
Hasil	Sistem memperlihatkan <i>preview</i> ad creative sesuai dengan pilihan aktor.
Status	Valid

2. Tabel 6.62 memperlihatkan tabel *test* Melihat Preview Ad Creative Alternatif 1.

Tabel 6.62 Kasus Uji Melihat Preview Ad Creative Alternatif 1

Nama Kasus Uji	Melihat Preview Ad Creative Alternatif 1: Jika menekan tombol “close”.
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Ad Creative” pada menu. 2. Aktor menekan tombol “Preview” pada ad creative yang diinginkan. 3. Aktor memilih jenis preview yang diinginkan. 4. Aktor menekan tombol “Close”.
Hasil yang diharapkan	Sistem akan menutup modal Preview Ad Creative.
Hasil	Sistem akan menutup modal Preview Ad Creative.
Status	Valid

6.3.33 Pengujian Validasi Mengedit Ad Creative

1. Tabel 6.63 memperlihatkan tabel *test* Berhasil Mengedit Ad Creative.

Tabel 6.63 Kasus Uji Berhasil Mengedit Iklan

Nama Kasus Uji	Mengedit Ad Creative
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Creative” di menu <i>sidebar</i>. 2. Menekan tombol “/” pada ad creative yang dipilih.

	3. Merubah form yang ingin diedit. 4. Menekan tombol "Save changes".
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> "Edit Ad Creative Berhasil!".
Hasil	Sistem memperlihatkan <i>alert</i> "Edit Ad Creative Berhasil!".
Status	Valid

2. Tabel 6.64 memperlihatkan tabel *test* Mengedit Ad Creative Alternative 1.

Tabel 6.64 Kasus Uji Mengedit Mengedit Ad Creative Alternatif 1

Nama Kasus Uji	Mengedit Ad Creative Alternatif 1: Mengosongkan beberapa form ad creative.
Prosedur	1. Menekan tombol "Ad Creative" di menu <i>sidebar</i> . 2. Menekan tombol "/" di ad creative yang dipilih. 3. Mengosongkan salah satu form ad creative. 4. Menekan tombol "Save changes".
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Hasil	Sistem memperlihatkan <i>exception error</i> "Wajib diisi, tidak boleh kosong".
Status	Valid

3. Tabel 6.64 memperlihatkan tabel *test* Mengedit Ad Creative Alternative 2.

Tabel 6.65 Kasus Uji Mengedit Ad Creative Alternatif 2

Nama Kasus Uji	Mengedit Ad Creative Alternatif 2: Jika menekan tombol "Close".
Prosedur	1. Menekan tombol "Ad Creative" di menu <i>sidebar</i> . 2. Menekan tombol "/" pada ad creative yang dipilih. 3. Merubah form yang ingin diedit. 4. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal Edit Ad Creative.
Hasil	Sistem menutup modal Edit Ad Creative.
Status	Valid

6.3.34 Pengujian Validasi Menghapus Ad Creative

1. Tabel 6.66 memperlihatkan tabel *test* Berhasil Menghapus Ad Creative.

Tabel 6.66 Kasus Uji Berhasil Menghapus Ad Creative

Nama Kasus Uji	Menghapus Ad Creative
Prosedur	1. Menekan tombol “Ad Creative” di menu <i>sidebar</i> . 2. Menekan tombol “X” pada ad creative yang ingin dihapus. 3. Menekan tombol “Ya, Tentu!”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Ad Creative berhasil dihapus”.
Hasil	Sistem memperlihatkan <i>alert</i> “Ad Creative berhasil dihapus”.
Status	Valid

2. Tabel 6.67 memperlihatkan tabel *test* Menghapus Ad Creative Alternatif 1.

Tabel 6.67 Kasus Uji Menghapus Ad Creative Alternatif 1

Nama Kasus Uji	Menghapus Ad Creative Alternatif 1: menekan tombol “Tidak Jadi!”.
Prosedur	1. Menekan tombol “Ad Creative” di menu <i>sidebar</i> . 2. Menekan tombol “X” di ad creative yang ingin dihapus. 3. Menekan tombol “Tidak Jadi!”.
Hasil yang diharapkan	Sistem menutup <i>alert</i> Delete Ad Creative.
Hasil	Sistem menutup <i>alert</i> Delete Ad Creative.
Status	Valid

6.3.35 Pengujian Validasi Menambah Ad Pixel

1. Tabel 6.68 memperlihatkan tabel *test* Berhasil Menambah Ad Pixel.

Tabel 6.68 Kasus Uji Berhasil Menambah Ad Pixel

Nama Kasus Uji	Menambah Ad Pixel
Prosedur	1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i> . 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Tambah Ad Pixel”.

	<p>4. Mengetikkan form tambah Ad Pixel yaitu nama Ad Pixel.</p> <p>5. Menekan tombol “Save Ad Pixel”.</p>
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Tambah Ad Pixel Berhasil!” dan menampilkan hasil tambah Ad Pixel terbaru pada tabel.
Hasil	Sistem memperlihatkan <i>alert</i> “Tambah Ad Pixel Berhasil!” dan menampilkan hasil tambah Ad Pixel terbaru pada tabel.
Status	Valid

2. Tabel 6.69 memperlihatkan tabel *test* Menambah Ad Pixel Alternatif 1.

Tabel 6.69 Kasus Uji Menambah Ad Pixel Alternatif 1

Nama Kasus Uji	Menambah Ad Pixel Alternatif 1: Jika terdapat form input kosong.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Tambah Ad Pixel”. 4. Mengosongkan salah satu form tambah ad pixel. 5. Menekan tombol “Save Ad Pixel”.
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Hasil	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Status	Valid

3. Tabel 6.70 memperlihatkan tabel *test* Menambah Ad Pixel Alternatif 2.

Tabel 6.70 Kasus Uji Menambah Ad Pixel Alternatif 2

Nama Kasus Uji	Menambah Ad Pixel Alternatif 2: Ketika menekan tombol “Close”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Tambah Ad Pixel”. 4. Mengetikkan form tambah Ad Pixel yaitu nama Ad Pixel.



	5. Menekan tombol "Close".
Hasil yang diharapkan	Sistem menutup modal tambah ad pixel.
Hasil	Sistem menutup modal tambah ad pixel.
Status	Valid

4. Tabel 6.71 memperlihatkan tabel *test* Menambah Ad Pixel Alternatif 3.

Tabel 6.71 Kasus Uji Menambah Ad Pixel Alternatif 3

Nama Kasus Uji	Menambah Ad Pixel Alternatif 3: Ketika akun iklan yang digunakan adalah tipe pribadi.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol "Ad Pixel" di menu <i>sidebar</i>. 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol "Tambah Ad Pixel". 4. Mengetikkan form tambah Ad Pixel yaitu nama Ad Pixel. 5. Menekan tombol "Save Ad Pixel".
Hasil yang diharapkan	Sistem menutup modal tambah ad pixel dan memperlihatkan <i>exception error</i> "akun pribadi hanya 1 pixel".
Hasil	Sistem menutup modal tambah ad pixel dan memperlihatkan <i>exception error</i> "akun pribadi hanya 1 pixel".
Status	Valid

6.3.36 Pengujian Validasi Melihat Ad Pixel

1. Tabel 6.72 memperlihatkan tabel *test* Berhasil Melihat Ad Pixel.

Tabel 6.72 Kasus Uji Berhasil Melihat Ad Pixel

Nama Kasus Uji	Melihat Ad Pixel
Prosedur	Menekan tombol "Ad Pixel" di menu <i>sidebar</i> .
Hasil yang diharapkan	Sistem memperlihatkan seluruh Ad Pixel dalam tabel.
Hasil	Sistem memperlihatkan seluruh Ad Pixel dalam tabel.
Status	Valid

6.3.37 Pengujian Validasi Mengedit Ad Pixel

1. Tabel 6.73 memperlihatkan tabel *test* Berhasil Mengedit Ad Pixel.

Tabel 6.73 Kasus Uji Berhasil Mengedit Ad Pixel

Nama Kasus Uji	Mengedit Ad Pixel
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Menekan tombol “Edit Nama Pixel “ di Ad Pixel pilihan. 3. Merubah form nama ad pixel yang ingin diedit. 4. Menekan tombol “Save changes”.
Hasil yang diharapkan	Sistem memperlihatkan alert “Edit Pixel Berhasil!”.
Hasil	Sistem memperlihatkan alert “Edit Pixel Berhasil!”.
Status	Valid

2. Tabel 6.74 memperlihatkan tabel *test* Mengedit Ad Pixel Alernatif 1.

Tabel 6.74 Kasus Uji Mengedit Mengedit Ad Pixel Alternatif 1

Nama Kasus Uji	Mengedit Ad Pixel Alternatif 1: Mengosongkan beberapa form Ad Pixel.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Menekan tombol “Edit Nama Pixel “ di Ad Pixel pilihan. 3. Mengosongkan beberapa form ad pixel. 4. Menekan tombol “Save changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Hasil	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Status	Valid

3. Tabel 6.75 memperlihatkan tabel *test* Mengedit Ad Pixel Alernatif 2.

Tabel 6.75 Kasus Uji Mengedit Ad Pixel Alternatif 2

Nama Kasus Uji	Mengedit Ad Pixel Alternatif 2: Ketika menekan tombol “Close”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>.

	<ol style="list-style-type: none"> 2. Menekan tombol “Edit Nama Pixel “ di Ad Pixel pilihan. 3. Merubah form nama ad pixel yang ingin diedit. 4. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal Edit Ad Pixel.
Hasil	Sistem menutup modal Edit Ad Pixel.
Status	Valid

6.3.38 Pengujian Validasi Set Halaman Website

1. Tabel 6.76 memperlihatkan tabel *test* Berhasil Set Halaman Website.

Tabel 6.76 Kasus Uji Berhasil Set Halaman Website

Nama Kasus Uji	Set Halaman Website
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Menekan tombol “Set Halaman” di Ad Pixel pilihan. 3. Mengetikkan Seluruh form Set Halaman. 5. Menekan tombol “Set Halaman”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Halaman Sudah Diset!”.
Hasil	Sistem memperlihatkan <i>alert</i> “Halaman Sudah Diset!”.
Status	Valid

2. Tabel 6.77 memperlihatkan tabel *test* Set Halaman Website Alternatif 1.

Tabel 6.77 Kasus Uji Set Halaman Website Alternatif 1

Nama Kasus Uji	Set Halaman Website Alternatif 1: Ketika terdapat form input kosong.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Set Halaman” di Ad Pixel pilihan. 4. Mengosongkan beberapa form set halaman website. 5. Menekan tombol “Set Halaman”.
Hasil yang diharapkan	Sistem memperlihatkan exception error “Wajib diisi, tidak boleh kosong”.

Hasil	Sistem memperlihatkan exception error “Wajib diisi, tidak boleh kosong”.
Status	Valid

3. Tabel 6.77 memperlihatkan tabel *test* Set Halaman Website Alternatif 2.

Tabel 6.78 Kasus Uji Set Halaman Website Alternatif 2

Nama Kasus Uji	Set Halaman Website Alternatif 2: Ketika menekan tombol “Close”.
Prosedur	1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i> . 2. Menekan tombol “Set Halaman” di Ad Pixel pilihan. 3. Mengetikkan Seluruh form Set Halaman. 5. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal set halaman.
Hasil	Sistem menutup modal set halaman.
Status	Valid

6.3.39 Pengujian Validasi Mengedit Halaman Website

1. Tabel 6.79 memperlihatkan tabel *test* Set Berhasil Mengedit Halaman Website.

Tabel 6.79 Kasus Uji Berhasil Mengedit Halaman Website

Nama Kasus Uji	Mengedit Halaman Website
Prosedur	1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i> . 2. Menekan tombol “Edit Halaman” di Ad Pixel pilihan. 3. Merubah form edit halaman yang ingin diedit. 4. Menekan tombol “Save Changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert</i> “Halaman Berhasil Dirubah!”.
Hasil	Sistem memperlihatkan <i>alert</i> “Halaman Berhasil Dirubah!”.
Status	Valid

2. Tabel 6.80 memperlihatkan tabel *test* Set Mengedit Halaman Website Alternatif 1.

Tabel 6.80 Kasus Uji Mengedit Halaman Website Alternatif 1

Nama Kasus Uji	Mengedit Halaman Website Alternatif 1: Ketika terdapat form input kosong.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Edit Halaman” di Ad Pixel pilihan. 4. Mengosongkan beberapa form edit halaman website. 5. Menekan tombol “Save Changes”.
Hasil yang diharapkan	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Hasil	Sistem memperlihatkan <i>exception error</i> “Wajib diisi, tidak boleh kosong”.
Status	Valid

3. Tabel 6.81 memperlihatkan tabel *test* Set Mengedit Halaman Website Alternatif 2.

Tabel 6.81 Kasus Uji Mengedit Halaman Website Alternatif 2

Nama Kasus Uji	Mengedit Halaman Website Alternatif 2: Jika menekan tombol “Close”
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i>. 2. Menekan tombol “Edit Halaman” di Ad Pixel pilihan. 3. Mengetikkan Seluruh form Set Halaman. 5. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal edit halaman.
Hasil	Sistem menutup modal edit halaman.
Status	Valid

6.3.40 Pengujian Validasi Menggenerate Ad Pixel

1. Tabel 6.82 memperlihatkan tabel *test* Berhasil Menggenerate Ad Pixel.

Tabel 6.82 Kasus Uji Berhasil Menggenerate Ad Pixel

Nama Kasus Uji	Menggenerate Ad Pixel
Prosedur	1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i> . 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Generate Pixel”.
Hasil yang diharapkan	Sistem memperlihatkan kode pixel berdasarkan inputan set halaman oleh aktor.
Hasil	Sistem memperlihatkan kode pixel berdasarkan inputan set halaman oleh aktor.
Status	Valid

6.3.41 Pengujian Validasi Copy to Clipboard

1. Tabel 6.83 memperlihatkan tabel *test* Berhasil Copy to Clipboard.

Tabel 6.83 Kasus Uji Berhasil Copy to Clipboard

Nama Kasus Uji	Copy to Clipboard
Prosedur	1. Menekan tombol “Ad Pixel” di menu <i>sidebar</i> . 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Menekan tombol “Generate Pixel”. 4. Mnegklik “copy to clipboard” pada pojok kanan atas.
Hasil yang diharapkan	Sistem memperlihatkan <i>alert copied</i> yang menandakan bahwa telah tersalin.
Hasil	Sistem memperlihatkan <i>alert copied</i> yang menandakan bahwa telah tersalin.
Status	Valid

6.3.42 Pengujian Validasi Mengganti Akun Iklan

1. Tabel 6.84 memperlihatkan tabel *test* Berhasil Menghapus Ad Creative.

Tabel 6.84 Kasus Uji Berhasil Mengganti Akun Iklan

Nama Kasus Uji	Mengganti Akun Iklan
Prosedur	1. Menekan tombol <i>dropdown</i> pilih akun iklan.

	2. Memilih akun iklan yang diinginkan.
Hasil yang diharapkan	Sistem memperlihatkan isi saldo, ad pixel, ad creative, kampanye, set iklan dan iklan dari akun yang dipilih.
Hasil	Sistem memperlihatkan isi saldo, ad pixel, ad creative, kampanye, set iklan dan iklan dari akun yang dipilih.
Status	Valid

2. Tabel 6.85 memperlihatkan tabel *test* Menghapus Ad Creative Alternatif 1.

Tabel 6.85 Kasus Uji Mengganti Akun Iklan Alternatif 1

Nama Kasus Uji	Mengganti Akun Iklan Alternatif 1: Ketika tidak memiliki akun iklan sama sekali.
Prosedur	1. Menekan tombol <i>dropdown</i> pilih akun iklan. 2. Memilih pilihan buat akun iklan pribadi.
Hasil yang diharapkan	Sistem meredirect ke facebook selama 5 detik, kemudian kembali ke halaman dashboard dengan akun iklan pribadi yang telah tersedia.
Hasil	Sistem meredirect ke facebook selama 5 detik, kemudian kembali ke halaman dashboard dengan akun iklan pribadi yang telah tersedia.
Status	Valid

6.3.43 Pengujian Validasi Melihat Sisa Anggaran

1. Tabel 6.86 memperlihatkan tabel *test* Berhasil Melihat Sisa Anggaran.

Tabel 6.86 Kasus Uji Berhasil Melihat Sisa Anggaran

Nama Kasus Uji	Melihat Sisa Anggaran
Prosedur	1. Telah masuk menggunakan login facebook. 2. Memilih akun iklan pada <i>dropdown</i> pilih akun iklan. 3. Melihat sisa anggaran pada pojok kanan atas layar.
Hasil yang diharapkan	Sistem memperlihatkan jumlah sisa anggaran pada pojok kana atas layar.



Hasil	Sistem memperlihatkan jumlah sisa anggaran pada pojok kanan atas layar.
Status	Valid

6.3.44 Pengujian Validasi Melihat Iklan Pada Dashboard

1. Tabel 6.87 memperlihatkan tabel *test* Berhasil Melihat Iklan Pada Dashboard.

Tabel 6.87 Kasus Uji Berhasil Melihat Iklan Pada Dashboard

Nama Kasus Uji	Melihat Iklan Pada Dashboard
Prosedur	1. Menekan tombol “Dashboard” di menu <i>sidebar</i> . 2. Melihat seluruh iklan pada tabel.
Hasil yang diharapkan	Sistem memperlihatkan seluruh iklan dalam tabel pada menu dashboard.
Hasil	Sistem memperlihatkan seluruh iklan dalam tabel pada menu dashboard.
Status	Valid

6.3.45 Pengujian Validasi Melihat Grafik Wawasan

1. Tabel 6.88 memperlihatkan tabel *test* Berhasil Melihat Grafik Wawasan.

Tabel 6.88 Kasus Uji Berhasil Melihat Grafik Wawasan

Nama Kasus Uji	Melihat Grafik Wawasan
Prosedur	1. Menekan tombol “Dashboard” di menu <i>sidebar</i> . 2. Melihat seluruh iklan pada tabel. 3. Menekan tombol “Lihat Grafik” pada iklan yang ingin dilihat grafiknya.
Hasil yang diharapkan	Sistem memperlihatkan seluruh wawasan iklan baik dari wawasan web maupun wawasan iklan yang berjalan dalam bentuk grafik.
Hasil	Sistem memperlihatkan seluruh wawasan iklan baik dari wawasan web maupun wawasan iklan yang berjalan dalam bentuk grafik.
Status	Valid

6.3.46 Pengujian Validasi Melihat Status Terakhir

1. Tabel 6.89 memperlihatkan tabel *test* Berhasil Melihat Status Terakhir.

Tabel 6.89 Kasus Uji Berhasil Melihat Status Terakhir

Nama Kasus Uji	Melihat Status Terakhir
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” di menu <i>sidebar</i>. 2. Melihat seluruh iklan pada tabel. 3. Menekan tombol “Lihat Status Akhir” di iklan yang ingin dilihat status terakhirnya. 4. Melihat status akhir iklan yang telah dipilih.
Hasil yang diharapkan	Sistem memperlihatkan informasi status akhir iklan dalam bentuk modal.
Hasil	Sistem memperlihatkan informasi status akhir iklan dalam bentuk modal.
Status	Valid

2. Tabel 6.90 memperlihatkan tabel *test* Melihat Status Terakhir Alternatif 1.

Tabel 6.90 Kasus Uji Melihat Status Terakhir Alternatif 1

Nama Kasus Uji	Melihat Status Terakhir Alternatif 1: Ketika menekan tombol “Close”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” di menu <i>sidebar</i>. 2. Melihat seluruh iklan pada tabel. 3. Menekan tombol “Lihat Status Akhir” pada iklan yang ingin dilihat status terakhirnya. 4. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal status iklan saat ini.
Hasil	Sistem menutup modal status iklan saat ini.
Status	Valid

6.3.47 Pengujian Validasi Melihat Set Iklan Pada Dashboard

1. Tabel 6.91 memperlihatkan tabel *test* Berhasil Melihat Set Iklan Pada Dashboard.

Tabel 6.91 Kasus Uji Berhasil Melihat Set Iklan Pada Dashboard

Nama Kasus Uji	Melihat Set Iklan Pada Dashboard
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” di menu <i>sidebar</i>. 2. Menekan <i>tab compare</i>. 3. Melihat seluruh set iklan pada menu dashboard.
Hasil yang diharapkan	Sistem memperlihatkan seluruh set iklan dalam tabel di menu dashboard.
Hasil	Sistem memperlihatkan seluruh set iklan dalam tabel di menu dashboard.
Status	Valid

6.3.48 Pengujian Validasi Compare Set Iklan

1. Tabel 6.92 memperlihatkan tabel *test* Berhasil Compare Set Iklan Dashboard.

Tabel 6.92 Kasus Uji Berhasil Compare Set Iklan

Nama Kasus Uji	Compare Set Iklan
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” di menu <i>sidebar</i>. 2. Menekan <i>tab</i> bandingkan. 3. Menekan tombol “compare”.
Hasil yang diharapkan	Sistem menambahkan status “compared” pada kolom status.
Hasil	Sistem menambahkan status “compared” pada kolom status.
Status	Valid

2. Tabel 6.93 memperlihatkan tabel *test* Compare Set Iklan Dashboard Alternatif 1.
 - 1.

Tabel 6.93 Kasus Uji Compare Set Iklan Alternatif 1

Nama Kasus Uji	Compare Set Iklan Alternatif 1: Ketika pada kolom status terdapat “compared”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” di menu <i>sidebar</i>. 2. Menekan <i>tab</i> bandingkan.

Hasil yang diharapkan	Sistem mengganti button “compare” menjadi button “Hapus Compare”.
Hasil	Sistem mengganti button “compare” menjadi button “Hapus Compare”.
Status	Valid

6.3.49 Pengujian Validasi Hapus Compare Set Iklan

1. Tabel 6.94 memperlihatkan tabel *test* Berhasil Hapus Compare Set Iklan.

Tabel 6.94 Kasus Uji Berhasil Hapus Compare Set Iklan

Nama Kasus Uji	Hapus Compare Set Iklan
Prosedur	1. Menekan tombol “Dashboard” di menu <i>sidabar</i> . 2. Menekan tab bandingkan. 3. Menekan tombol “Hapus Compare” di set iklan yang ingin dihapus comparenya.
Hasil yang diharapkan	Sistem men- <i>delete</i> status “compared” pada kolom status.
Hasil	Sistem men- <i>delete</i> status “compared” pada kolom status.
Status	Valid

2. Tabel 6.95 memperlihatkan tabel *test* Hapus Compare Set Iklan Alternatif 1.

Tabel 6.95 Kasus Uji Hapus Compare Set Iklan Set Iklan Alternatif 1

Nama Kasus Uji	Hapus Compare Set Iklan Alternatif 1: Ketika kolom status tidak terdapat “compared”.
Prosedur	1. Menekan tombol “Dashboard” di menu <i>sidebar</i> . 2. Menekan tab bandingkan.
Hasil yang diharapkan	Sistem <i>button</i> “Hapus Compare” tidak tersedia.
Hasil	Sistem <i>button</i> “Hapus Compare” tidak tersedia.
Status	Valid

6.3.50 Pengujian Validasi Melihat Log

1. Tabel 6.96 memperlihatkan tabel *test* Berhasil Melihat Log.

Tabel 6.96 Kasus Uji Berhasil Melihat Log

Nama Kasus Uji	Melihat Log
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” di menu <i>sidebar</i>. 2. Megklik <i>tab</i> bandingkan. 3. Menekan tombol “Log Compare” di set iklan yang diinginkan. 4. Memilih tanggal untuk log yang dilihat. 5. Menekan tombol “Get Log”.
Hasil yang diharapkan	Sistem memperlihatkan seluruh daftar log pada set iklan berdasarkan tanggal yang dipilih.
Hasil	Sistem memperlihatkan seluruh daftar log pada set iklan berdasarkan tanggal yang dipilih.
Status	Valid

2. Tabel 6.97 memperlihatkan tabel *test* Melihat Log Alternatif 1.

Tabel 6.97 Kasus Uji Melihat Log Alternatif 1

Nama Kasus Uji	Melihat Log Alternatif 1: Ketika aktor menekan tombol “Close”.
Prosedur	<ol style="list-style-type: none"> 1. Menekan tombol “Dashboard” pada menu. 2. Menekan <i>tab</i> bandingkan. 3. Menekan tombol “Log Compare” di set iklan yang diinginkan. 4. Memilih tanggal untuk log yang dilihat. 5. Menekan tombol “Close”.
Hasil yang diharapkan	Sistem menutup modal pilih tanggal log.
Hasil	Sistem menutup modal pilih tanggal log.
Status	Valid

6.3.51 Pengujian Validasi Melihat Notifikasi

1. Tabel 6.98 memperlihatkan tabel *test* Berhasil Melihat Notifikasi.

Tabel 6.98 Kasus Uji Berhasil Melihat Notifikasi

Nama Kasus Uji	Melihat Notifikasi
Prosedur	1. Menekan tombol “Masuk Dengan Facebook” ketika teridentifikasi sebagai pengguna. 2. Melihat halaman menu dashboard. 3. Melihat notifikasi pada ikon lonceng.
Hasil yang diharapkan	Sistem memperlihatkan notifikasi pada ikon lonceng ketika terdapat iklan yang diberhentikan secara otomatis.
Hasil	Sistem memperlihatkan notifikasi pada ikon lonceng ketika terdapat iklan yang diberhentikan secara otomatis.
Status	Valid

6.3.52 Pengujian Validasi Logout

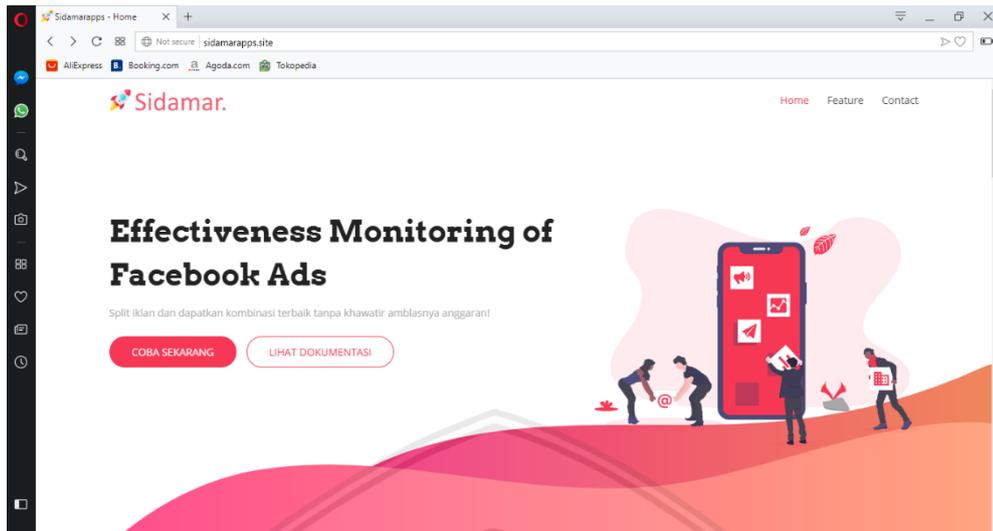
1. Tabel 6.99 memperlihatkan tabel *test* Berhasil Logout.

Tabel 6.99 Kasus Uji Berhasil Logout

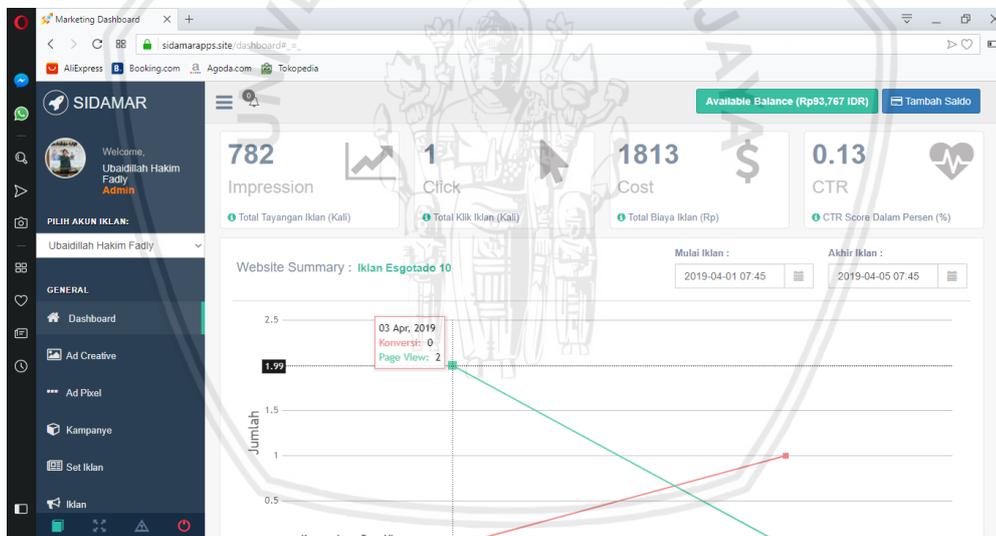
Nama Kasus Uji	Logout
Prosedur	Menekan tombol “Logout” di menu.
Hasil yang diharapkan	Sistem meredirect menuju <i>page</i> login.
Hasil	Sistem meredirect menuju <i>page</i> login.
Status	Valid

6.4 Pengujian *Compatibility*

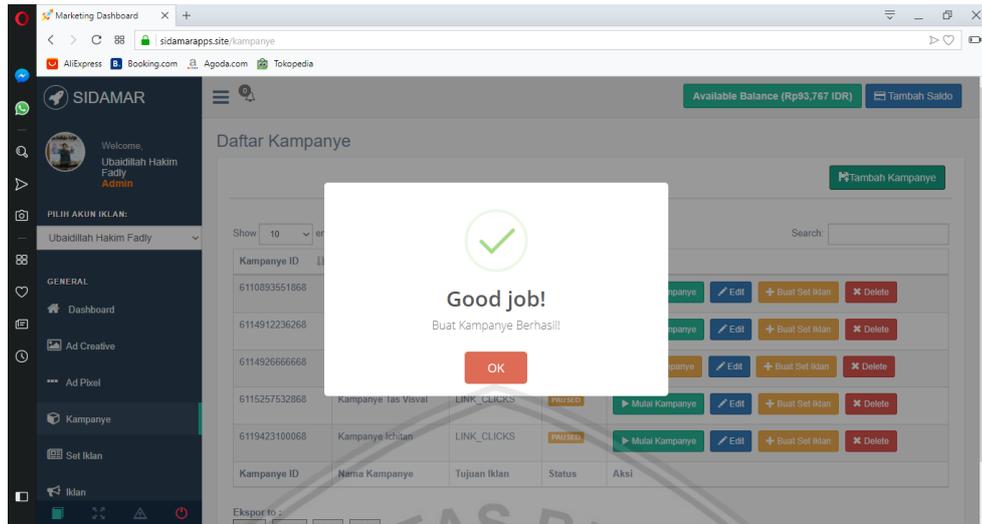
Pengujian *compatibility* akan men-*running* aplikasi di 4 peramban yang berbeda (Opera Mini, Chrome, Edge, dan Mozilla Firefox) untuk memperlihatkan fungsionalitas sistem dapat *running* pada keempat peramban yang diujikan. Rangkuman uji pada peramban Opera Mini dapat diperlihatkan di Gambar 6.5-6.8, Gambar 6.9-6.13 menunjukkan perolehan pengujian pada *browser* chrome, Gambar 6.14-6.18 menggambarkan perolehan dari pengujian pada *browser* edge, dan Gambar 6.19-6.23 menggambarkan perolehan dari pengujian di peramban Mozilla.



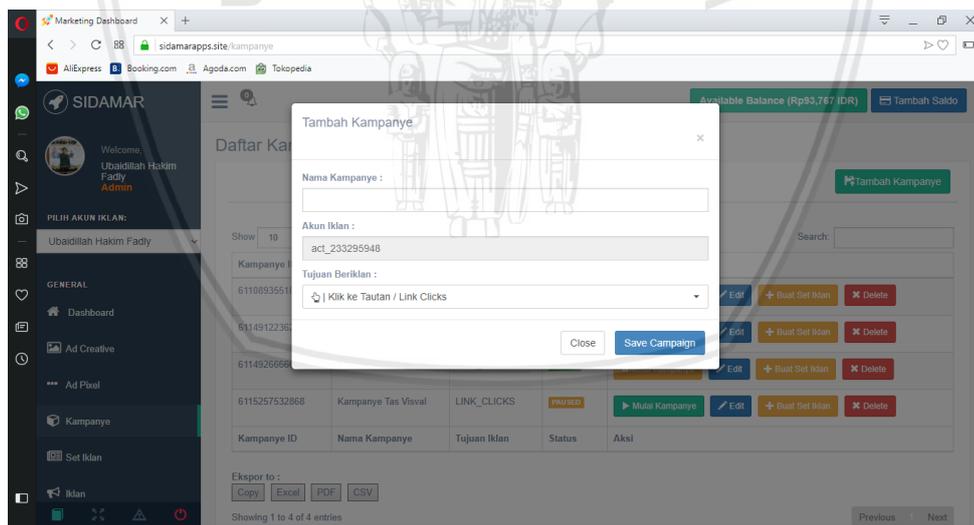
Gambar 6.5 Pengujian Compatibility pada Peramban Opera 1



Gambar 6.6 Pengujian Compatibility pada Peramban Opera 2

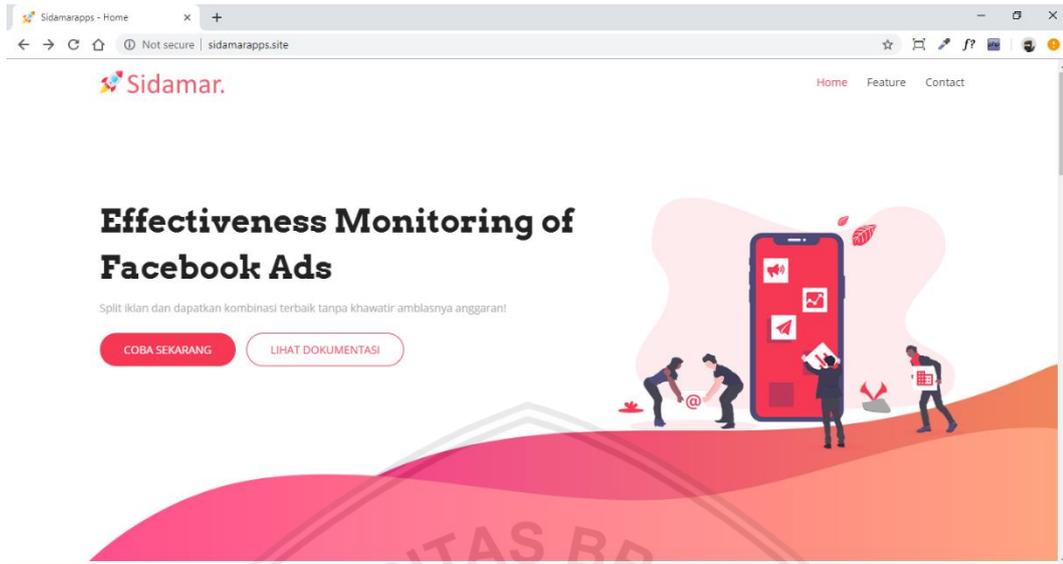


Gambar 6.7 Pengujian Compatibility pada Peramban Opera 3

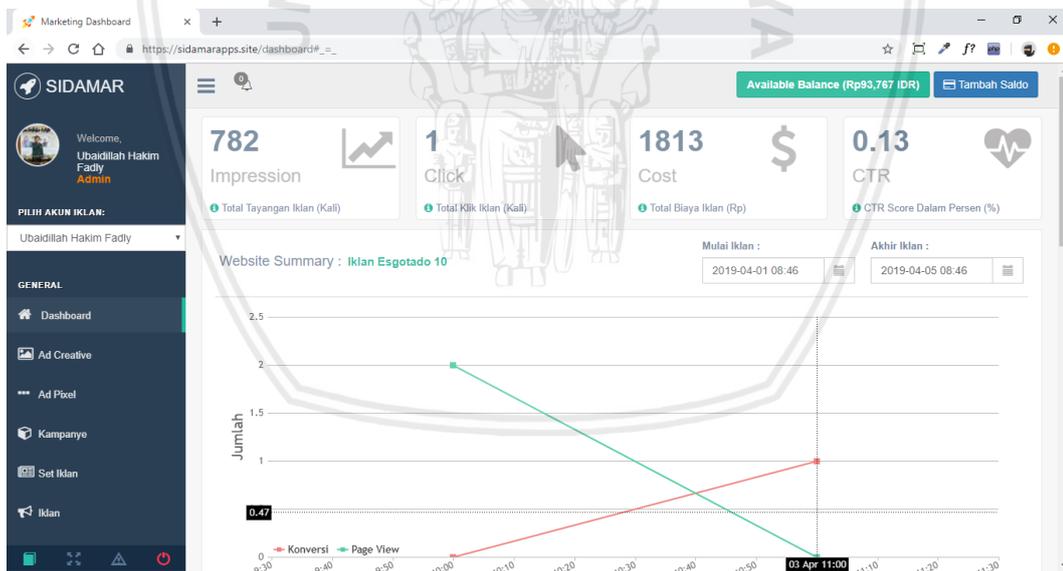


Gambar 6.8 Pengujian Compatibility pada Peramban Opera 4

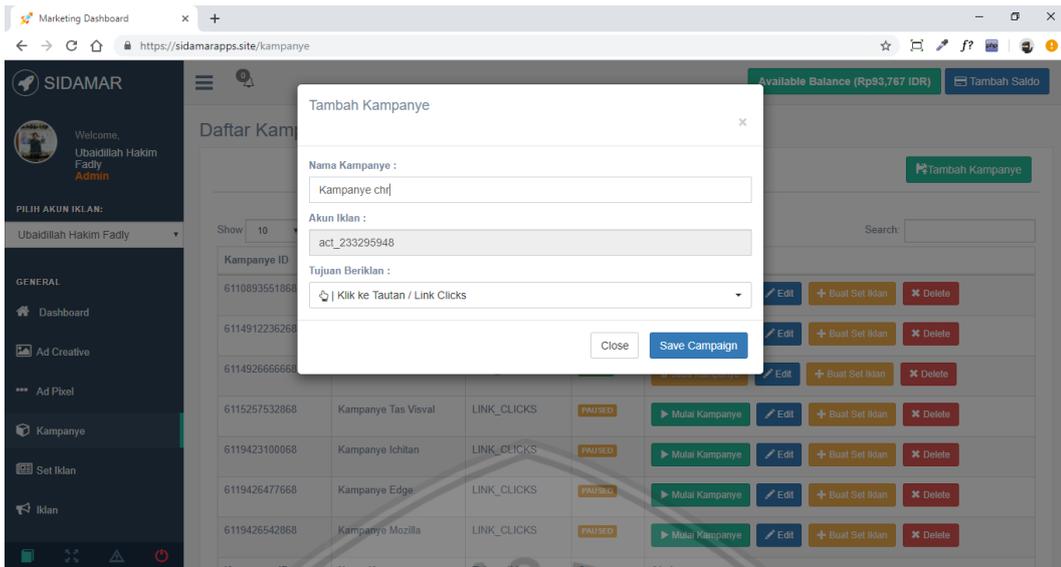




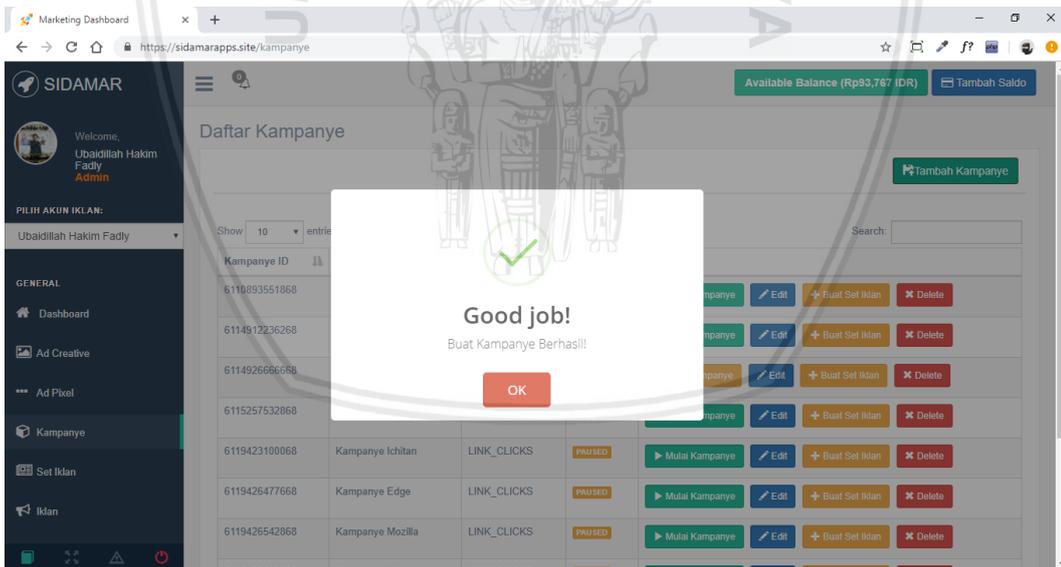
Gambar 6.9 Pengujian Compatibility pada Peramban Chrome 1



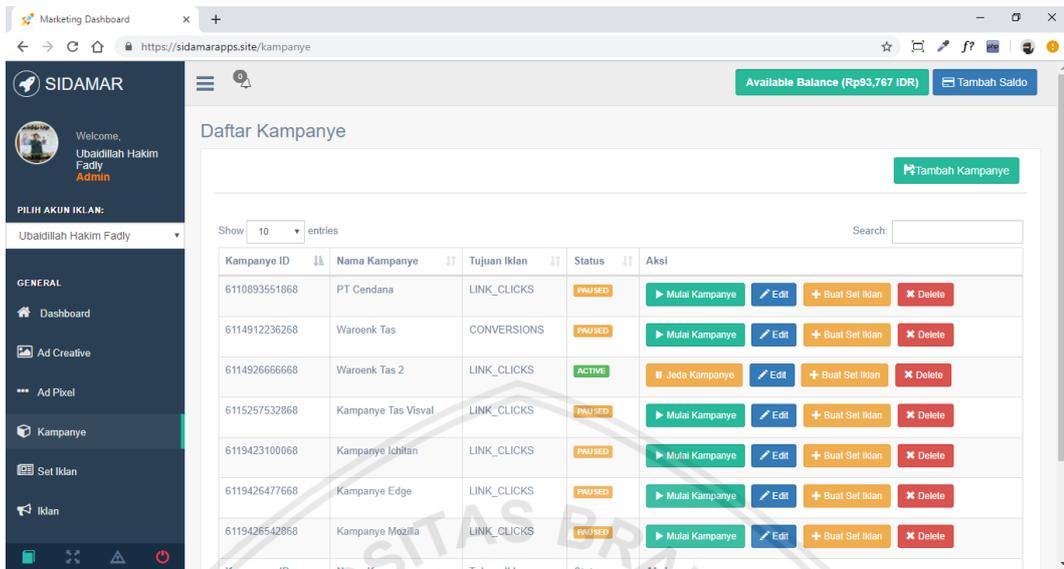
Gambar 6.10 Pengujian Compatibility pada Peramban Chrome 2



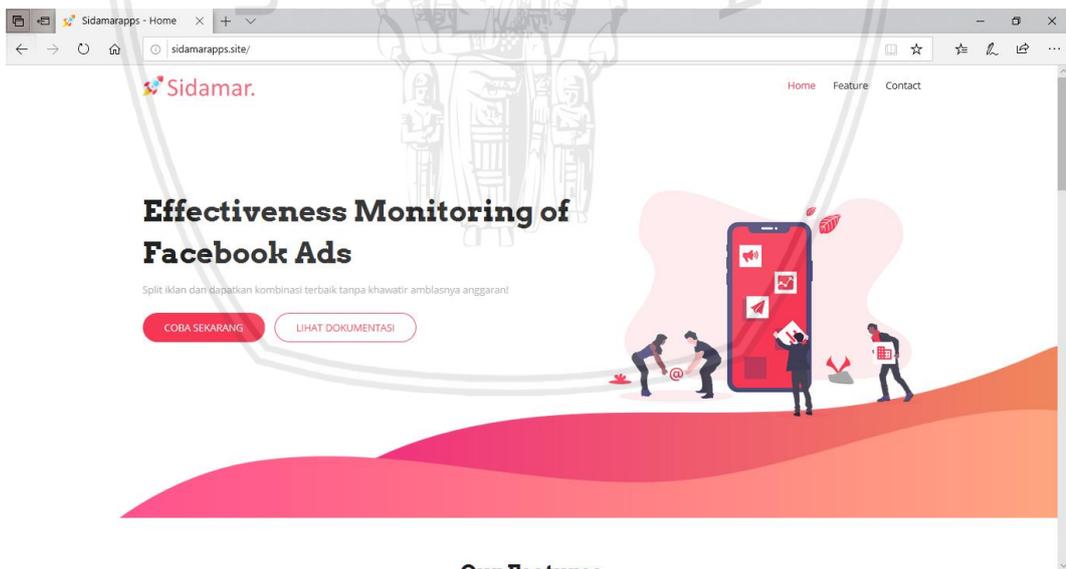
Gambar 6.11 Pengujian Compatibility pada Peramban Chrome 3



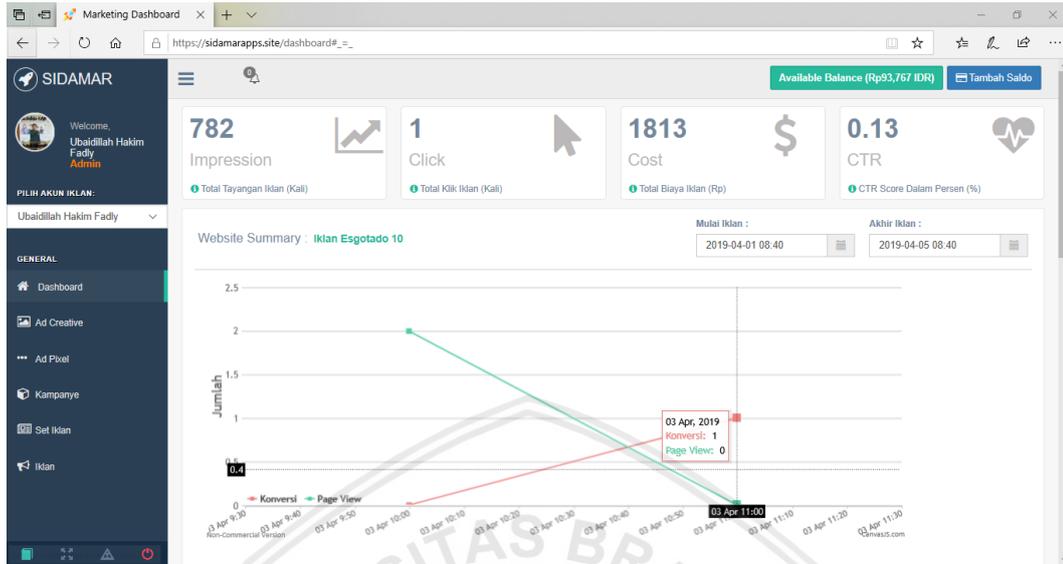
Gambar 6.12 Pengujian Compatibility pada Peramban Chrome 4



Gambar 6.13 Pengujian Compatibility pada Peramban Chrome 5



Gambar 6.14 Pengujian Compatibility pada Peramban Edge 1

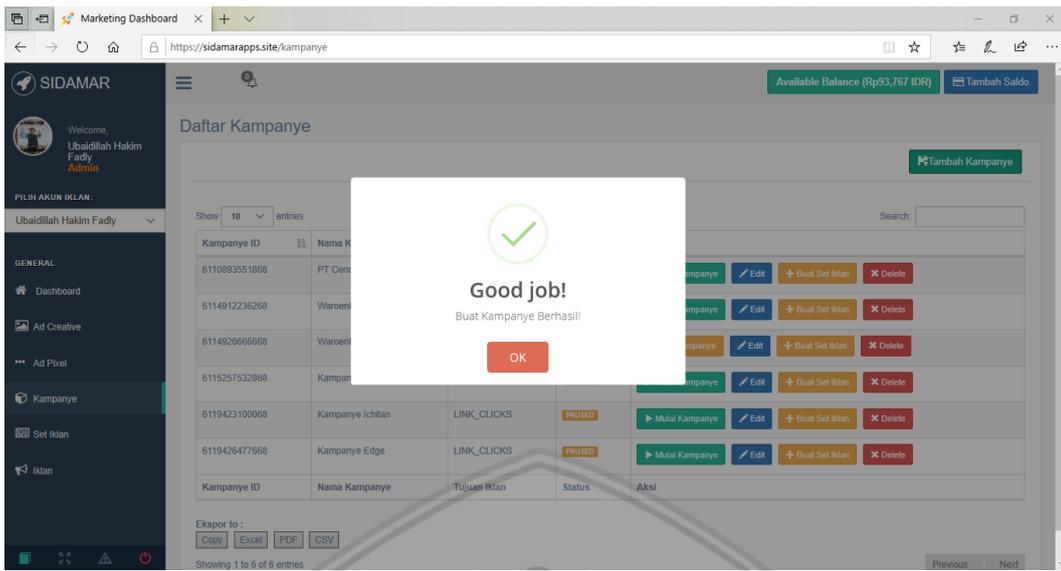


Gambar 6.15 Pengujian Compatibility pada Peramban Edge 2

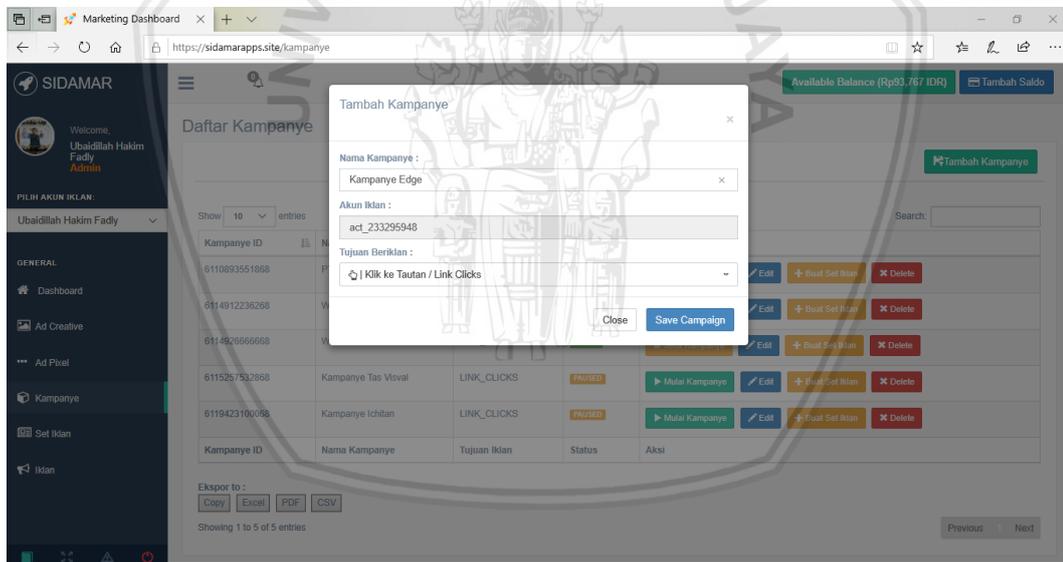
The screenshot shows a 'Daftar Kampanye' (Campaign List) with the following data:

Kampanye ID	Nama Kampanye	Tujuan Iklan	Status	Aksi
6110893551868	PT Cendana	LINK_CLICKS	PAUSED	Mulai Kampanye, Edit, Buat Set Iklan, Delete
6114912236268	Waroenk Tas	CONVERSIONS	PAUSED	Mulai Kampanye, Edit, Buat Set Iklan, Delete
6114926666668	Waroenk Tas 2	LINK_CLICKS	ACTIVE	Jeda Kampanye, Edit, Buat Set Iklan, Delete
6115257532868	Kampanye Tas Mival	LINK_CLICKS	PAUSED	Mulai Kampanye, Edit, Buat Set Iklan, Delete
6119423100068	Kampanye Ichtan	LINK_CLICKS	PAUSED	Mulai Kampanye, Edit, Buat Set Iklan, Delete

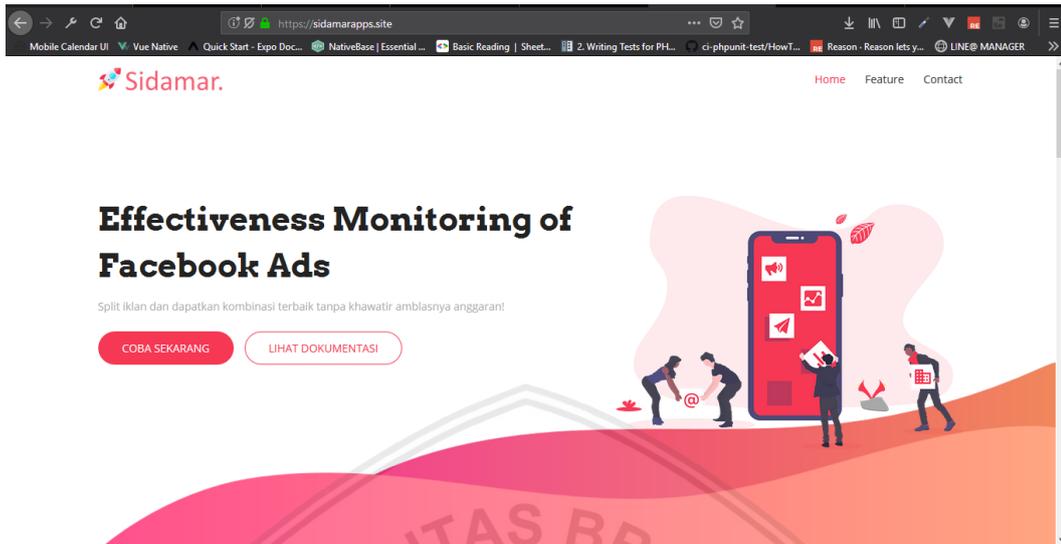
Gambar 6.16 Pengujian Compatibility pada Peramban Edge 3



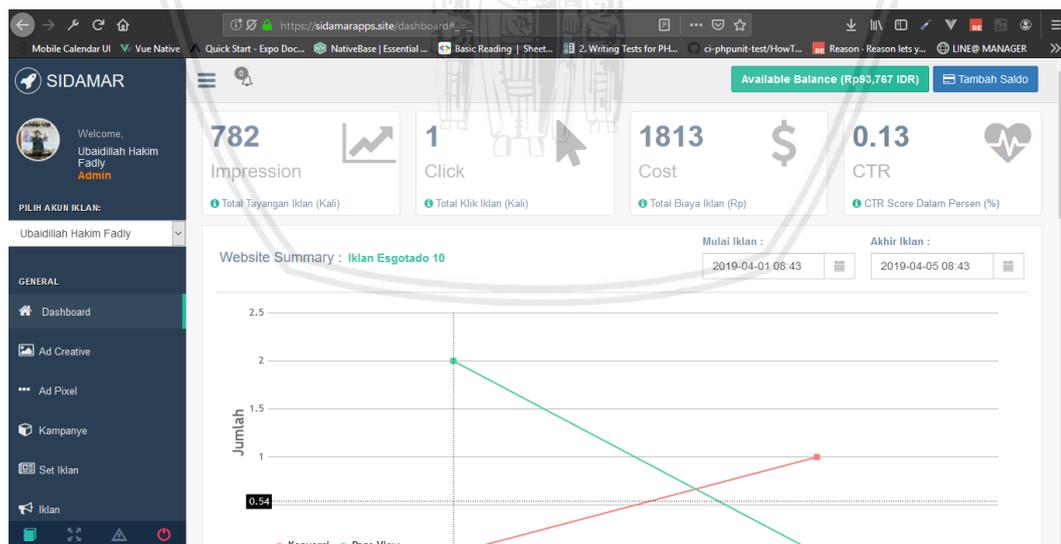
Gambar 6.17 Pengujian Compatibility pada Peramban Edge 4



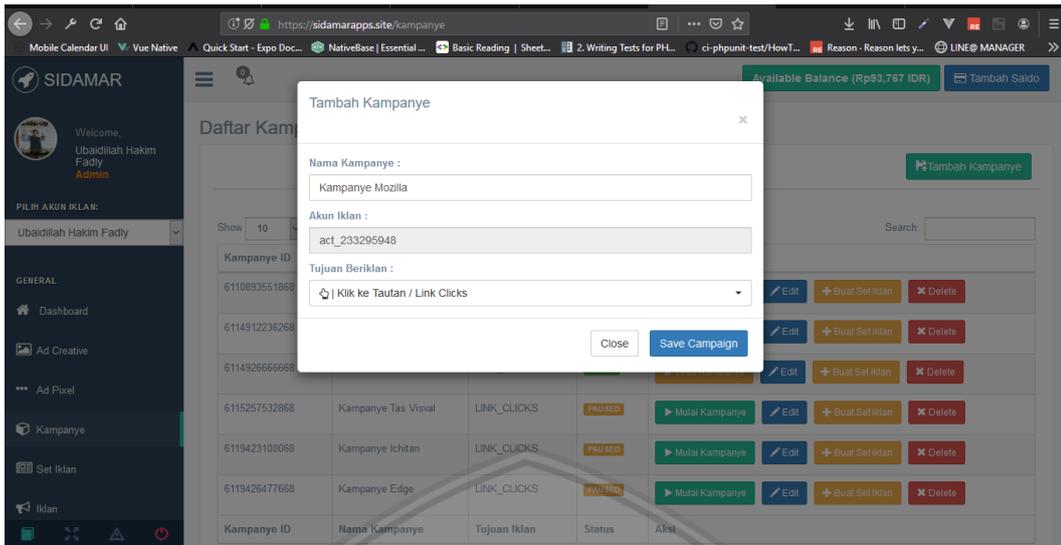
Gambar 6.18 Pengujian Compatibility pada Peramban Edge 5



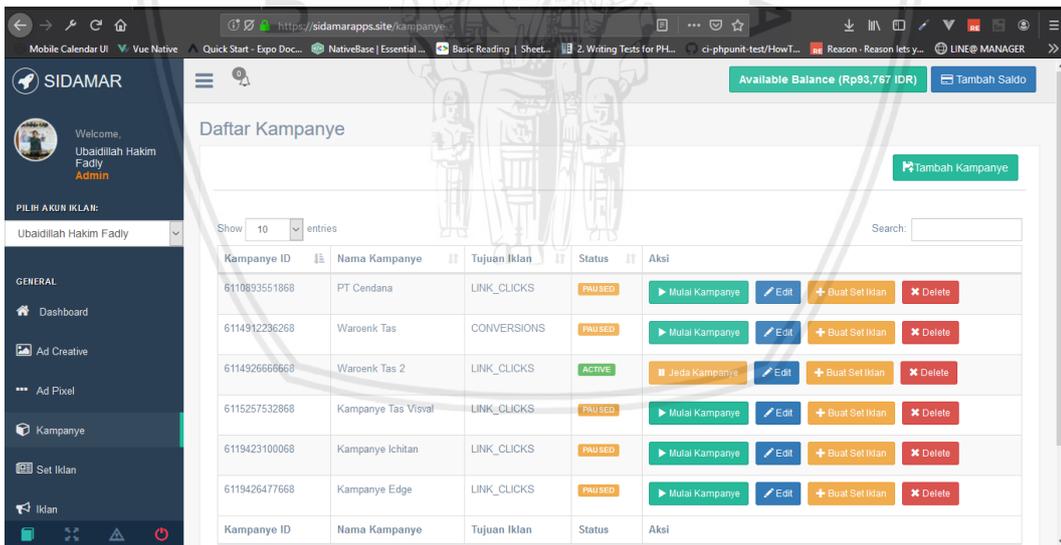
Gambar 6.19 Pengujian Compatibility pada Peramban Mozilla 1



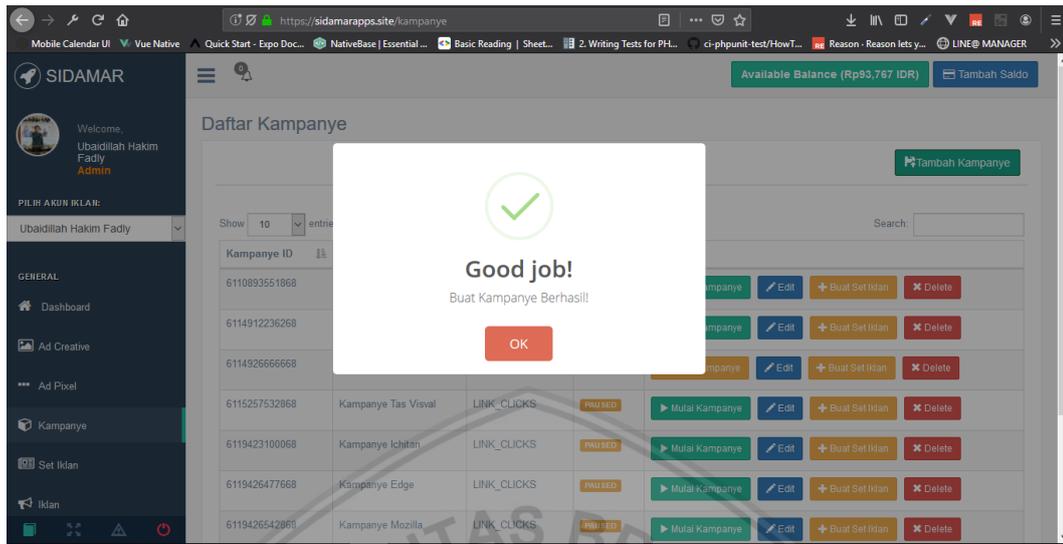
Gambar 6.20 Pengujian Compatibility pada Peramban Mozilla 2



Gambar 6.21 Pengujian Compatibility Pada peramban Mozilla 3



Gambar 6.22 Pengujian Compatibility Pada Peramban Mozilla 4



Gambar 6.23 Pengujian Compatibility Pada Peramban Mozilla 5



BAB 7 PENUTUP

Bab 7 akan membahas kesiimpulan yang didapatkan selama pembangunan aplikasi *dashboard* untuk analisa efektivitas *marketing online* berbasis *website* kemudian akan diberikan saran kepada penelitian kedepannya.

7.1 Kesimpulan

Setelah dilakukan pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* Berbasis *Website* yang telah dilalui. Maka diraih sebagian kesimpulan berikut.

1. Perolehan dari analisis kebutuhan diperoleh 3 aktor yaitu pengunjung, pengguna dan *marketer* dan didapatkan sekitar 52 kebutuhan fungsional yang harus ada pada aplikasi dan juga terdapat 1 kebutuhan non-fungsional yang diperoleh dari hasil studi literatur serta wawancara.
2. Berdasarkan hasil perancangan sistem, diperoleh hasil berupa perancangan arsitektur yang terdiri dari perancangan *sequence diagram* dan perancangan *class diagram*, kemudian perancangan basis data yang terdiri dari perancangan *Entiti Relationship Diagram* (ERD) dan *Physical Data Model* (PDM), kemudian perancangan komponen yang berisi algoritme, dan terakhir perancangan tampilan atau antarmuka.
3. Berdasarkan tahapan implementasi, dihasilkan aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* yang memiliki fitur-fitur utama seperti Membuat satu set iklan (kampanye, set iklan dan iklan), membuat ad creative, membuat ad pixel dan melihat dashboard yang dilengkapi dengan penghentian serta penjadwalan iklan otomatis yang dibangun dengan bahasa pemrograman PHP dan diorganisasikan oleh kerangka Kerja CI.
4. Berdasarkan hasil testing yang dilaksanakan dengan tiga cara untuk fungsionalitas yakni pengujian unit, pengujian integrasi dan pengujian validasi serta 1 pengujian untuk non-fungsional yaitu *compatibility testing*. Didapatkan bahwasannya ketika dilakukan pengujian unit untuk tiga contoh sampel yaitu: `tambahSetIklan()`, `getIklanGrafik`, dan `tambahKampanye()` didapatkan V(G) atau *cyclomatic complexity* sebesar 3 yang menunjukkan tingkat kompleksitas cukup baik. Pada pengujian integrasi, dilaksanakan pada contoh sampel fungsi `ctrCheck` dengan membuat beberapa stub pada kelas lain untuk mencoba integrasi yang dilakukan. Pengujian integrasi ini menggunakan pendekatan *top-down* yang menghasilkan 100% valid atau *passed*, pengujian validasi memiliki karakteristik seperti pengujian kotak hitam atau *blackbox* yang diujikan pada 52 kebutuhan fungsional yang menghasilkan 100% valid, dan terakhir untuk pengujian non-fungsional dilakukan menggunakan *compability testing* untuk 4 peramban yang berlainan yaitu: Opera, Mini, *Microsoft Edge*, *Chrome*, dan

Mozilla Firefox yang menghasilkan 100% valid yang berarti fungsionalitas sistem dapat *running* secara baik untuk keseluruhan peramban diujikan.

7.2 Saran

Adapun untuk saran dari Pembangunan Aplikasi *Dashboard* Untuk Analisa Efektivitas *Marketing Online* diantaranya adalah sebagai berikut.

1. Menambahkan Penargetan Iklan se-spesifik mungkin.
2. Menambahkan sistem pendukung keputusan dalam mengisi setiap langkah pembuatan iklan yang *knowledge base* nya diambil dari pakar marketing online sehingga semakin mempercepat dan mempertajam analisa efektivitas iklan untuk pengambilan keputusan.
3. Mengintegrasikan dengan media marketing online yang lain seperti Google Ads, Youtube, Instagram, Line dsb.



DAFTAR REFERENSI

- Novell (2004) *Secure Enterprise Dashboard: a Key to Business Agility*. White Paper
- Roger S. Pressman (2012) *Rekayasa Perangkat Lunak Pendekatan Praktisi (edisi 7)*. Yogyakarta: Andi Offset.
- Alexa, 2018. *topsites*. Alexa.
- Allen, J., 2018. *Return on Ad Spend, an Unsung Metric*. [online] Available at: <<https://www.ppchero.com/return-on-ad-spend-an-unsung-metric/>> [Accessed 2 Mar. 2019].
- Amy Gallo, 2017. *A Refresher on Marketing ROI*. [online] Harvard Business Review. Available at: <<https://hbr.org/2017/07/a-refresher-on-marketing-roi>> [Accessed 24 Jun. 2019].
- Andrus, A., 2018. *What is ROAS? The Complete Guide to Using Return on Ad Spend*. [online] Available at: <<https://www.disruptiveadvertising.com/marketing/roas-return-on-ad-spend/>> [Accessed 24 Jun. 2019].
- Arief, M.R., 2011. *Pemrograman Web dinamis menggunakan PHP & MySQL*. Yogyakarta: ANDI dan STMIK Amikom.
- Beighley, L., 2010. *jQuery for Dummies*. Indianapolis, Indiana: Wiley Publishing, Inc.
- Daqiqil, I., 2011. *Framework Codeigniter*. Pekanbaru: koder.web.id.
- Deacon, J., 2005. Model-view-controller (mvc) architecture. *Computer Systems Development*.
- Dharwiyanti, S. and Wahono, R.S., 2003. Pengantar Unified Modeling Language (UML). *IlmuKomputer.com*.
- Dianawati, A., 2007. *6 Rahasia Sukses Menjadi jutawan Internet*. Jakarta: Media Kita.
- Few, S., 2006. *Information Dashboard Design. The effective visual communication of data Sebastopol*.
- J.E.N.I, 2007. *Pemrograman Berorientasi Objek Dengan Java*.
- Jeffery, M., 2010. *Data-Driven Marketing*. Canada: John Willey & Sons Inc.
- Khalid, M., 2006. *PHP / CURL Book with Examples Version 1.8*. 1.8 ed. Pakistan: Imran & LAMPTechnologies.
- Kotler, K., 2014. Manajemen Pemasaran. *Manajemen Pemasaran*.
- Kotler Phillip, A.G., 2008. *Prinsip-Prinsip Pemasaran*. Erlangga.
- Mohapatra, P.K.J., 2010. *Software Engineering*. New Delhi: New Age International Publishers Pvt Ltd.

Powell, G., 2002. *ROMI, Return on Marketing Investment, Demand More from your Marketing and Sales Investments*. Atlanta: RPI Press.

Pressman, R.S., 2009. *Software Engineering A Practitioner's Approach 7th Ed - Roger S. Pressman. Software Engineering A Practitioner's Approach 7th Ed - Roger S. Pressman*.

R. Kelly Rainer, E.T., 2009. *Introduction to Information Systems: Supporting and Transforming Business*. 2nd ed. Office. United State Of America: Don Fowley.

Rafi Mohammed, Robert J. Fisher, Bernard Jaworski, G.P., 2003. *Internet Marketing*. 2nd ed. New York: McGraw-Hill Education.

Rosa and Salahudin, 2011. *Modul Pembelajaran Rekayasa Perangkat Lunak : Terstruktur dan Beorientasi Objek*. Bandung: Modula.

Sakur, S.B., 2010. *PHP 5 Pemrograman Berorientasi Objek, Konsep dan Implementasi*. Yogyakarta: Andi.

Sommerville, I., 2010. *Software Engineering. Software Engineering*.

Swastha, B.D.& H.H., 2012. *Manajemen Pemasaran: Analisa Perilaku Konsumen*. 1st ed. Yogyakarta.

We Are Social, 2018. *Digital in 2018 in Southeast Asia Part 2 - South-East*. [online] LinkedIn Slideshare. Available at: <<https://www.slideshare.net/wearesocial/digital-in-2018-in-southeast-asia-part-2-southeast-86866464>> [Accessed 23 Sep. 2018].

Yeni Kustiyahningsih, D.R.A., 2011. *Pemograman Basis Data Berbasis Web Menggunakan PHP & MySQL*. Yogyakarta: Graha Ilmu.

LAMPIRAN A SURAT PERNYATAAN

SURAT KETERANGAN

Yang bertanda tangan di bawah ini :

Nama : S. Nowo Aprilianto, S.Psi

Jabatan : Owner Digital Marketing Malang (www.malangnet.com)

Dengan ini menerangkan bahwa mahasiswa di bawah ini :

Nama : Ubaidillah Hakim Fadly

NIM : 155150201111290

Fakultas : Ilmu Komputer

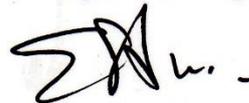
Jurusan : Teknik Informatika

Judul Skripsi : Pembangunan Aplikasi Dashboard Untuk Analisa Efektivitas Marketing Online Berbasis Website

Telah melakukan penelitian di Digital Marketing Malang pada tanggal 16 Januari 2019. Demikian surat keterangan ini diberikan agar dapat dipergunakan sebagaimana mestinya.

Malang, 10 Maret 2019

Owner Digital Marketing Malang,



S. Nowo Aprilianto, S.Psi

LAMPIRAN B HASIL WAWANCARA

Berikut ini merupakan lampiran berisi hasil wawancara yang telah dilakukan kepada. Tujuan dari wawancara ini adalah untuk menemukan permasalahan yang digunakan sebagai latar belakang pembangunan aplikasi.

Hasil Wawancara :

1. Apa yang digunakan para digital marketing untuk optimalisasi iklan?

Jawaban :

Enaknya mas jika beriklan kan ada segmenting yang kita tentukan, jadi semisal saya jualan produk jilbab dan mengkampanyekan iklan di facebook dengan saya kasih budget Rp. 200.000 nah dari budget ini bisa tentukan larinya kemana, itu bisa mas. Jadi ya di tentukan yang pertama muslim kemudian dispesifikkan lagi muslim yang ada ketertarikan dengan jilbab. Nah itu disebut targeting. Nah seringnya orang boncos kan disini mas, ga ada analisis *audience* nya.

2. Apakah ada software khusus yang digunakan untuk analisis wawasan iklan?

Jawaban :

Engga ada mas, kita pakai default dari facebooknya kan disana semua datanya sudah ada. Jadi semisal ada customer dengan budget Rp. 500.000 untuk bulan ini, nah itu biasanya saya sarankan untuk split, jangan konsen pada satu titik. Kenapa? Karena ketika focus pada satu titik dan ketika titik ini *fail* atau gagal udah gagal semuanya mas.

3. Apakah selama beriklan di *social media marketing* pernah mengalami kesulitan atau habisnya anggaran iklan dengan tidak maksimal? Saya dengar dari teman-teman yang pernah beriklan katanya ribet dan anggarannya gampang ludes.

Jawaban :

Ribet kalau saya sih engga, ribet itu kan gatau caranya mas. Nah yang disebut boncos itu ya ngeludesin anggaran iklan itu istilahnya kalo di teman-teman *digital marketing* itu boncos Namanya, keluar duit banyak tapi gaada yang beli gitu kan ya. Ya itu tadi mas harus ada analisis dan strategi, harus ada uang yang dibakar untuk *splitting* iklan. Kemudian contoh paling *basic* ya harus paham desain visual yang menarik jadi ketika orang pertama lihat langsung masuk ingatan nah itu peran utama. Kemudian *copy writing*, redaksionalnya, cara menyampaikannya, cara menggiring orang untuk membaca lebih lanjut kemudian klik dan kemudian masuk *landing page* nah itu bagaimana. Tanpa itu mas, ya iklannya ya iklan biasa meskipun berbayar gaada bedanya mas sama hanya *share* di grup misalnya.

4. Berarti banyak sekali ya pak, cara atau langkah-langkah agar iklan bisa berjalan efektif?

Jawaban :

Iya mas, banyak sekali kalo ini di bikin *workshop complete* ya A-Z itu seminggu mas, kebanyakan orang males gitu mas belajar, maunya bagaimana 2 jam saja sudah ahli. Ya gabisa mas langsung kelihatan gitu karena banyak sekali yang harus dipelajari perlakuan beda produk, beda audience, beda segmen, ya secara psikologis nanti ngerubah semua dari awal, dari konten visual dll.

5. Jadi bagaimana pak cara untuk mengatasi masalah boncos itu tadi pak?

Jawaban :

Ya itu tadi mas, boncos itu kan karena tidak tahu masalah konten, copy writing, redaksional dll. Kemudian yang paling penting terkait masalah kebijakan anggaran iklan, semisal disaku ada Rp. 600.000 untuk iklan 3 bulan jangan dihabiskan sebulan. Bagi tiap bulan sebesar Rp.200.000 itu jauh lebih bijak itu ketika tidak ada budget misalnya ya ambil dr keuntungan, semisal untuk 2 juta itu Rp. 200.000 *keep* buat iklan jangan dimasukkan kantong, itu untuk iklan berikutnya. Kemudian masalah *splitting* iklan mas, iklan itu tidak bisa hanya 1 saja mas, perlu banyak sekali percobaan dan di *running* secara bersama-sama untuk satu iklan yang sama untuk menemukan iklan yang paling menguntungkan istilahnya racikan yang pas.

Malang, 10 Maret 2019

Owner Digital Marketing Malang,



S. Nowo Aprilianto, S.Psi