

**RANCANG BANGUN APLIKASI REKOMENDASI TENTOR
BIMBINGAN BELAJAR MENGGUNAKAN METODE *PROFILE*
MATCHING BERBASIS ANDROID**

**SKRIPSI
LABORATORIUM PEMROGRAMAN APLIKASI PERANGKAT
BERGERAK**

Diajukan untuk Memenuhi Persyaratan Memperoleh Gelar Sarjana Komputer



Disusun Oleh :

LULUK SETIAWATI HARTONO

NIM. 115060801111056

**KEMENTERIAN RISET DAN TEKNOLOGI
PROGRAM STUDI INFORMATIKA/ ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA**

MALANG

2015

LEMBAR PERSETUJUAN
RANCANG BANGUN APLIKASI REKOMENDASI TENTOR
BIMBINGAN BELAJAR MENGGUNAKAN METODE *PROFILE*
***MATCHING* BERBASIS ANDROID**

SKRIPSI
LABORATORIUM PEMROGRAMAN APLIKASI PERANGKAT
BERGERAK

Diajukan untuk Memenuhi Persyaratan Memperoleh Gelar Sarjana Komputer



Disusun Oleh :

LULUK SETIAWATI HARTONO
115060801111056

Telah diperiksa dan disetujui oleh:

Pembimbing 1

Pembimbing 2

Aryo Pinandito, ST, M.MT

NIP. 19830519 201404 1 001

Edy Santoso, S.Si., M.Kom

NIP. 19740414 200312 1 004

LEMBAR PENGESAHAN
RANCANG BANGUN APLIKASI REKOMENDASI TENTOR
BIMBINGAN BELAJAR MENGGUNAKAN METODE *PROFILE*
***MATCHING* BERBASIS ANDROID**

SKRIPSI
LABORATORIUM PEMROGRAMAN APLIKASI PERANGKAT
BERGERAK

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer

Disusun Oleh :

LULUK SETIAWATI HARTONO
NIM. 115060801111056

Skripsi ini telah diuji dan dinyatakan lulus tanggal 23 Juni 2015 xxxx

Penguji I,

Penguji II,

Dr. Eng Herman Tolle, S.T., M.T.
NIP. 19740823 200012 1 001

Agi Putra Kharisma, S.T., M.T.
NIK. 2013048604301001

Penguji III,

Adharul Muttaqin, S.T., M.T.
NIP. 19760121 200501 1 001

Mengetahui,
Ketua Program Studi Informatika/Illmu Komputer

Drs. Marji, M.T.
NIP. 19670801 199203 1 001

LEMBAR PERNYATAAN

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

Apabila ternyata di dalam 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, 23 Juni 2015

Mahasiswa

Luluk Setiawati Hartono

NIM. 115060801111056



KATA PENGANTAR

Puji Syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena hanya dengan rahmat dan karuniaNya, penulis dapat menyelesaikan skripsi dengan judul “Rancang Bangun Aplikasi Rekomendasi Tentor Bimbingan Belajar Menggunakan Metode Profile Matching Berbasis Android”. Melalui kesempatan ini, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan batuan dan dukungan selama penulisan skripsi ini, diantaranya:

1. Ayah Alm. Hartono dan Ibu Saniyanti dan dua adik penulis serta seluruh keluarga besar atas segala doa, nasehat dan dukungan yang diberikan hingga terselesainya skripsi ini.
2. Bapak Ir. Sutrisno, M.T, Bapak Ir. Heru Nurwarsito, M.Kom, Bapak Himawat Aryadita, S.T., M.Sc dan Bapak Edy Santoso, S.Si., M.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2, dan Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
3. Bapak Drs. Marji, M.T dan Bapak Issa Arwani, S.Kom., M.Sc selaku Ketua dan Sekretaris Program Studi Informatika/Ilmu Komputer Universitas Brawijaya.
4. Bapak Aryo Pinandito, S.T., M.MT selaku dosen pembimbing I yang telah memberikan masukan dan ilmu serta saran dalam menyelesaikan skripsi ini.
5. Bapak Edy Santoso, S.Si., M.Kom selaku dosen pembimbing II yang juga memberikan masukan dan ilmu serta saran dalam menyelesaikan skripsi ini.
6. Bapak Sabriansyah Rizqika Akbar, S.T, M.Eng selaku dosen pembimbing akademik yang telah memberikan pengarahan selama penulis menempuh pendidikan di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
7. Seluruh Dosen Teknik Informatika Universitas Brawijaya atas kesediaannya membagi ilmunya kepada penulis.
8. Bimbingan Belajar Sonic Class Malang yang telah mengizinkan penulis untuk dapat melakukan penelitian di tempat dan memberi data serta mendukung terselesainya skripsi ini.

- repository.ub.ac.id
9. Teman-teman penulis yaitu Haris, Vina, Rasita, Tya, Arin, Marwa, Vita, Fenty, Rachma, Rina, TIF-E 2011, BEM TIIK II, dan PMK DANIEL yang selalu memberi semangat dan membantu dalam pengerjaan skripsi ini.
 10. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat secara langsung atau tidak langsung agar terselesaikannya skripsi ini.
 11. Penulis menyadari bahwa skripsi ini masih banyak kekurangan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diharapkan untuk menyempurnakan skripsi ini. Semoga skripsi ini membawa manfaat bagi pihak lain yang menggunakannya.

Malang, 23 Juni 2015

Penulis



ABSTRAK

Rancang Bangun Aplikasi Rekomendasi Tentor Bimbingan Belajar Menggunakan Metode *Profile Matching* Berbasis Android. Oleh: Luluk Setiawati Hartono, Aryo Pinandito dan Edy Santoso. Skripsi Program Studi Informatika/Illmu Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya.

Pendidikan sekarang tidaklah mudah, banyak siswa sekolah yang tidak mampu menyerap pelajaran yang diajarkan oleh gurunya di sekolah. Tidak sedikit orang tua yang mengikutkan anaknya untuk mengikuti bimbingan belajar privat untuk mendapatkan proses belajar yang lebih baik dan mudah dipahami. Namun, ketika administrator bimbingan belajar akan mencari tentor yang tepat untuk seorang siswa les, administrator harus menyortir semua tentor yang ada untuk menentukan tentor yang tepat secara manual. Administrator juga harus menyebarkan informasi siswa les tersebut melalui pesan singkat maupun telepon kepada semua tentor yang ada. Hal ini menyebabkan kurang efektifnya biaya maupun waktu.

Sistem yang dibangun dalam penelitian ini merupakan sistem rekomendasi tentor bimbingan belajar dengan menggunakan metode *Profile Matching*. Konsep umum dari metode *Profile Matching* adalah pencocokan profil calon tentor dengan siswa les sehingga akan diperoleh selisih atau gap. Gap yang semakin kecil akan membuat bobot nilai semakin besar dan kesempatan untuk direkomendasikan juga semakin besar. Hasil akhir dari proses *Profile Matching* ini berupa ranking tentor yang akan digunakan untuk pengiriman informasi penawaran siswa les dengan *push notification*.

Informasi siswa les akan disebarkan melalui *push notification* dengan menggunakan Google Cloud Messaging berdasarkan hasil rekomendasi tadi. Hasil pengujian fungsional sistem 100% valid dan hasil pengujian akurasi untuk metode *Profile Matching* adalah 80%.

Kata Kunci: *Profile Matching*, Tentor, Google Cloud Messaging

ABSTRACT

System Design of Tutor Recommendation Application Using *Profile Matching* Method Based on Android. Author: Luluk Setiawati Hartono, Aryo Pinandito, and Edy Santoso. Departement of Information/Computer Science, Informatics Technology and Computer Science Program, Brawijaya University's Thesis.

Education is not easy, many students are not able to absorb the lessons that taught by a teacher at school. Not a few parents include their children to attend private tutoring to get a better learning process and easily understood. However, when administrators tutoring looking for the right tutor for a student, administrators have to sort through all the existing tutor to determine the exact tentor manually. Administrators also should spread the information about the student via short message or telephone to all existing tentor. This is not effective in terms of cost and time.

The system built in this research is a recommendation tutor system using Profile Matching method. The general concept of the Profile Matching method is matching profile between tentor and student, so we can get the difference or gap. The smaller gap will make the value weight become greater and make the opportunity to be recommended candidate also getting bigger. The last result from the process of Profile Matching is candidate rank that is used to give the recommendation for administrator to be used to delivery students offers with push notification.

Student's information will send via push notification using Google Cloud Messaging. Push notification will be sent based on the tutor rank. Functional testing result is 100% valid and accuracy of test results for Profile Matching method is 80%.

Keywords: *Profile Matching, Tutor, Google Cloud Messaging*

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN	iv
KATA PENGANTAR.....	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xii
DAFTAR PERSAMAAN.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	4
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Kajian Pustaka	6
2.2 Bimbingan Belajar	9
2.3 <i>Profile Matching</i>	9
2.4 <i>Native App</i>	14
2.5 Java	15
2.6 JSON.....	15
2.7 <i>Web service</i>	17
2.8 <i>Google Cloud Messaging (GCM)</i>	18
2.9 Pengujian Perangkat Lunak	20



BAB III METODOLOGI PENELITIAN	22
3.1 Studi Literatur	23
3.3 Perancangan dan Desain Sistem	23
3.4 Implementasi Sistem	24
3.5 Pengujian Sistem	24
3.6 Laporan Akhir	24
BAB IV ANALISIS DAN PERANCANGAN	25
4.1 Analisis Kebutuhan Sistem	26
4.2 Perancangan Perangkat Lunak	32
BAB V IMPLEMENTASI.....	70
5.1 Spesifikasi Sistem	70
5.2 Implementasi Basis Data	71
5.3 Implementasi <i>Class</i>	72
5.4 Implementasi Kode Program	73
5.5 Implementasi Antarmuka	79
BAB VI PENGUJIAN DAN ANALISIS	81
6.1 Pengujian <i>Black Box</i>	81
6.2 Pengujian Akurasi	85
BAB VII PENUTUP.....	90
7.1 Kesimpulan	90
7.2 Saran	90
DAFTAR PUSTAKA.....	91



DAFTAR GAMBAR

Gambar 2. 1 Cara Kerja JSON	16
Gambar 2. 2 Arsitektur Web service	17
Gambar 2. 3 Cara Kerja GCM	19
Gambar 3. 1 Alur Metodologi Penelitian	22
Gambar 4. 1 Pohon Perancangan Sistem	25
Gambar 4. 2 Diagram Use Case	28
Gambar 4. 3 Diagram Activity Melihat Ranking Tentor	31
Gambar 4. 4 Diagram Activity Menjawab Penawaran Admin	32
Gambar 4. 5 Perancangan Arsitektur Sisi Server	33
Gambar 4. 6 Perancangan Arsitektur Sisi Client	34
Gambar 4. 7 Perancangan Fitur Push Notification	34
Gambar 4. 8 Diagram Sequence Melihat Ranking Tentor	35
Gambar 4. 9 Diagram Sequence Menjawab Penawaran Admin	36
Gambar 4. 10 Diagram Class Melihat Ranking Tentor	37
Gambar 4. 11 Diagram Class Menjawab Penawaran Admin	37
Gambar 4. 12 Struktur Data JSON data tentor	50
Gambar 4. 13 Contoh JSON data tentor	51
Gambar 4. 14 Struktur Data JSON jadwal siswa les	51
Gambar 4. 15 Contoh JSON jadwal siswa les	52
Gambar 4. 16 Diagram Alir <i>Profile Matching</i>	52
Gambar 4. 17 Perancangan Antarmuka Melihat Rekomendasi Tentor	65
Gambar 4. 18 Perancangan Antarmuka Menjawab Penawaran Admin	66
Gambar 5. 1 Implementasi Basis Data Sistem Rekomendasi Tentor	71
Gambar 5. 2 Implementasi Code Menghitung Kriteria Jenjang	74
Gambar 5. 3 Implementasi Code Menghitung Kriteria Waktu	75
Gambar 5. 4 Implementasi Code Penentuan Ranking	76
Gambar 5. 5 Implementasi Code Respon Tidak	77
Gambar 5. 6 Implementasi Code Mengirim Notifikasi	78
Gambar 5. 7 Implementasi Antarmuka Melihat Rekomendasi Tentor	79
Gambar 5. 8 Implementasi Antarmuka Menjawab Penawaran Admin	80

DAFTAR TABEL

Tabel 2. 1 Perbandingan Data	7
Tabel 2. 2 Contoh Tabel Bobot Nilai Gap	10
Tabel 4. 1 Tabel Identifikasi Aktor	26
Tabel 4. 2 Tabel Kebutuhan Fungsional	27
Tabel 4. 3 Deskripsi Use Case Melihat ranking tentor yang direkomendasikan ..	28
Tabel 4. 4 Deskripsi Use Case Menjawab Penawaran Admin.....	29
Tabel 4. 5 Pemetaan Aktor terhadap kebutuhan fungsional	31
Tabel 4. 6 Tabel Admin	38
Tabel 4. 7 Tabel Hari	39
Tabel 4. 8 Tabel Jam	39
Tabel 4. 9 Tabel bp_jekel.....	40
Tabel 4. 10 Tabel bp_jenang.....	40
Tabel 4. 11 Tabel bp_matpel.....	41
Tabel 4. 12 Tabel bp_waktu.....	41
Tabel 4. 13 Tabel bn_jekel.....	42
Tabel 4. 14 Tabel bn_jenang.....	43
Tabel 4. 15 Tabel bn_matpel.....	44
Tabel 4. 16 Tabel bn_waktu.....	45
Tabel 4. 17 Tabel tentor	45
Tabel 4. 18 Tabel siswales	47
Tabel 4. 19 Tabel jadwal_siswales.....	48
Tabel 4. 20 Tabel jadwal_tentor.....	48
Tabel 4. 21 Tabel jenjang_tentor	49
Tabel 4. 22 Tabel matpel_tentor	50
Tabel 4. 23 Tabel Rancangan Aturan Profil Penilaian Waktu	53
Tabel 4. 24 Tabel Rancangan Aturan Gap Kriteria Waktu	53
Tabel 4. 25 Tabel Rancangan Aturan Profil Penilaian Mata Pelajaran.....	54
Tabel 4. 26 Tabel Rancangan Aturan Gap Kriteria Mata Pelajaran	55
Tabel 4. 27 Tabel Rancangan Aturan Profil Jenis Kelamin.....	56
Tabel 4. 28 Tabel Rancangan Aturan Gap Kriteria Jenis Kelamin.....	57



Tabel 4. 29 Tabel Rancangan Aturan Profil Penilaian Jenjang	57
Tabel 4. 30 Tabel Rancangan Aturan Gap Kriteria Jenjang	57
Tabel 4. 31 Tabel Hasil Pemetaan Gap Kompetensi Waktu	58
Tabel 4. 32 Tabel Hasil Pemetaan Gap Kompetensi Mata Pelajaran	59
Tabel 4. 33 Tabel Hasil Pemetaan Gap Kompetensi Jenis Kelamin	59
Tabel 4. 34 Tabel Hasil Pemetaan Gap Jenjang	60
Tabel 4. 35 Tabel Hasil Bobot Nilai Gap Waktu	60
Tabel 4. 36 Tabel Hasil Bobot Nilai Gap Mata Pelajaran	61
Tabel 4. 37 Tabel Hasil Bobot Nilai Gap Jenis Kelamin	61
Tabel 4. 38 Tabel Hasil Bobot Nilai Jenjang	61
Tabel 4. 39 Tabel Pengelompokkan Bobot Nilai Gap Waktu	62
Tabel 4. 40 Tabel Nilai Total Kriteria Waktu	63
Tabel 4. 41 Tabel Nilai Total Kriteria Mata Pelajaran	63
Tabel 4. 42 Tabel Nilai Total Kriteria Jenis Kelamin	63
Tabel 4. 43 Tabel Nilai Total Kriteria Jenjang	64
Tabel 4. 44 Tabel Hasil Akhir	64
Tabel 4. 45 Tabel Hasil Perankingan	65
Tabel 4. 46 Tabel Keterangan Antarmuka Melihat Rekomendasi Tentor	66
Tabel 4. 47 Tabel Keterangan Antarmuka Menjawab Penawaran Admin	67
Tabel 4. 48 Desain Uji Melihat Rekomendasi Tentor	67
Tabel 4. 49 Desain Uji Menjawab Penawaran Admin	68
Tabel 4. 50 Tabel Desain Pengujian Akurasi	68
Tabel 5. 1 Implementasi <i>Class</i> Melihat Rekomendasi Tentor	72
Tabel 5. 2 Implementasi <i>Class</i> Menjawab Penawaran Admin	73
Table 6. 1 Tabel Kasus Uji Melihat Rekomendasi Tentor	81
Table 6. 2 Tabel Kasus Uji Menjawab Penawaran Admin	83
Table 6. 3 Tabel Kesimpulan Pengujian Fungsional	85
Table 6. 4 Tabel Data Siswa Les Uji Pertama	86
Table 6. 5 Tabel Pengujian Akurasi Data Uji Pertama	86
Table 6. 6 Tabel Data Siswa Les Uji Kedua	87
Table 6. 7 Tabel Pengujian Akurasi Data Uji Kedua	87
Table 6. 8 Tabel Data Siswa Les Uji Ketiga	88

Table 6. 9 Tabel Pengujian Akurasi Data Uji Ketiga..... 88



DAFTAR PERSAMAAN

Persamaan (2-1)	Penentuan gap kompetensi referensi.....	10
Persamaan (2-2)	Penentuan gap kompetensi penelitian.....	10
Persamaan (2-3)	Perhitungan <i>core factor</i> referensi.....	11
Persamaan (2-4)	Perhitungan <i>core factor</i> penelitian.....	11
Persamaan (2-5)	Perhitungan <i>secondary factor</i> referensi	12
Persamaan (2-6)	Perhitungan <i>secondary factor</i> penelitian.....	12
Persamaan (2-7)	Perhitungan nilai total referensi	13
Persamaan (2-8)	Perhitungan nilai total penelitian	13
Persamaan (2-9)	Perhitungan penentuan ranking referensi.....	13
Persamaan (2-10)	Perhitungan penentuan ranking penelitian.....	14
Persamaan (4-1)	Persamaan Total Aspek Waktu.....	61
Persamaan (4-2)	Persamaan Total Aspek Mata Pelajaran.....	61
Persamaan (4-3)	Persamaan Total Aspek Jenis Kelamin.....	61
Persamaan (4-4)	Persamaan Total Aspek Jenjang	61
Persamaan (4-5)	Persamaan Hasil Akhir.....	63
Persamaan (4-6)	Persamaan Pengujian Akurasi Metode <i>Profile Matching</i>	68



BAB I PENDAHULUAN

1.1 Latar Belakang

Memasuki era globalisasi yang sedang berkembang saat ini, belajar merupakan kebutuhan utama masyarakat Indonesia. Belajar adalah kegiatan yang fundamental dalam pendidikan yang mana seorang individu dapat menjadikannya dasar untuk mencapai keberhasilan dengan interaksi dan pengalaman yang didapatnya [SUH-13]. Belajar di sekolah merupakan belajar yang dilakukan secara formal dan sekolah memberikan pengajar-pengajar yang ahli di bidangnya. Namun, masih ada siswa didik yang masih tidak dapat memahami apa yang disampaikan oleh pengajar. Mengatasi hal tersebut, tidak sedikit orang tua yang memilih untuk memasukan anaknya ke bimbingan belajar dengan harapan anaknya mendapatkan pemahaman materi yang lebih jelas.

Bimbingan belajar adalah suatu proses pemberian bantuan dari guru kepada siswa les dengan cara mengembangkan suasana belajar yang kondusif dan menumbuhkan kemampuan agar siswa dapat mengatasi kesulitan belajar yang dihadapinya sehingga mencapai hasil belajar yang optimal. Dalam lembaga pendidikan, guru berupaya menstimulasi siswa agar potensinya berkembang seoptimal mungkin [SUH-13]. Ketika seorang anak didaftarkan ke dalam sebuah bimbingan belajar maka administrator bimbingan belajar tersebut akan mencari tutor yang sesuai. Dalam mencari tutor yang sesuai tersebut, administrator bimbingan belajar sangat kesulitan dalam memilih tutor yang sesuai dengan siswa les. Selain itu administrator harus menghubungi masing-masing tutor melalui pesan singkat maupun telepon, hal ini belum efektif untuk menawarkan siswa les ke tutor yang bersedia mengajar. Banyak faktor yang dapat dipertimbangkan untuk merekomendasikan tutor yang sesuai dengan data siswa les, seperti jenis kelamin tutor, kesesuaian jadwal siswa dengan jadwal kosong tutor, pelajaran dan jenjang pendidikan yang dikuasai tutor.

Metode *Profile Matching* adalah sebuah mekanisme pengambilan keputusan dengan mengasumsikan bahwa terdapat *variable predicator* yang ideal yang harus dimiliki dan bukan merupakan tingkat minimal yang harus dipenuhi

atau dilewati [SET-13]. Berdasarkan perbandingan antara data siswa les dengan alternatif data tutor akan diketahui perbedaan atau selisih nilai dari masing-masing faktor yang menjadi pertimbangan pemilihan tutor. Perbedaan atau selisih inilah yang disebut gap. Semakin kecil gap yang dihasilkan maka bobot nilainya semakin besar, itu berarti alternatif memiliki peluang lebih besar untuk dipilih [DWI-13]. Dalam sistem ini, *Profile Matching* merupakan metode dimana ada suatu data siswa yang dijadikan acuan untuk menentukan tutor yang sesuai. Dari data yang didapat dari hasil pengolahan metode tersebut, akan dilakukan pengurutan gap dari yang terkecil.

Proses komunikasi administrator bimbingan belajar saat menawarkan siswa les adalah menggunakan pesan singkat maupun telepon. Hal ini dirasa kurang efektif saat seorang administrator harus menghubungi masing-masing tutor dengan resiko bahwa tutor yang dihubungi belum tentu dapat menerima siswa les yang ditawarkan. Proses ini akan menghabiskan banyak waktu dan biaya untuk proses penawaran siswa les. Android menyediakan fasilitas *Google Cloud Messaging* (GCM) dimana fitur ini tidak dimiliki oleh sistem operasi lain. GCM adalah sebuah layanan dari Google kepada para pengembang untuk mengimplementasikan *push notification*. Layanan ini menggantikan *Cloud to Device Messaging* (C2DM) terdahulunya. GCM akan mempermudah aplikasi ini untuk melakukan proses penawaran siswa dari administrator dan proses persetujuan dari tutor kepada administrator bimbingan belajar [YUL-13].

Aplikasi ini diharapkan dapat membantu administrator bimbingan belajar memberikan rekomendasi tutor yang sesuai dengan siswa les dengan menggunakan metode *Profile Matching*. Selain itu, aplikasi ini juga dapat membantu dalam pemberian informasi kepada tutor yang direkomendasikan tersebut dengan *push notification* yang memanfaatkan teknologi GCM.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan di atas maka permasalahan dapat dirumuskan sebagai berikut:

1. Bagaimana penerapan metode *Profile Matching* dalam sistem yang dapat merekomendasikan tutor yang sesuai untuk siswa les berdasarkan jenis

kelamin tentor dengan siswa les, kesesuaian jadwal siswa dengan jadwal kosong tentor, pendidikan yang dikuasai tentor dan mata pelajaran yang dikuasai tentor?

2. Bagaimana rancangan dan implementasi sistem yang dapat memberikan informasi penawaran siswa les kepada tentor yang direkomendasikan dengan menggunakan teknik *push notification*?
3. Seberapa besar tingkat akurasi yang dihasilkan dari pengujian metode *Profile Matching* dalam merekomendasikan tentor yang sesuai untuk siswa les berdasarkan jenis kelamin, jadwal kosong, pendidikan dan mata pelajaran yang dikuasai tentor di *Sonic Class* Bimbingan Belajar Malang?

1.3 Batasan Masalah

Agar penelitian ini dapat terfokus, perlu dilakukan pembatasan masalah, antara lain:

1. Aplikasi ini dibuat pada platform Android untuk client dan menggunakan PHP untuk server.
2. Persoalan yang diselesaikan adalah membangun mekanisme yang efektif pada bimbingan belajar dengan menggunakan *Profile Matching* dalam perekomendasi tentor dan menggunakan *push notification* dalam penawaran siswa les kepada tentor yang direkomendasikan.
3. Data tentor dan siswa les yang digunakan berasal dari *Sonic Class* Bimbingan Belajar Malang.

1.4 Tujuan

Tujuan yang ingin dicapai dari penelitian ini adalah:

1. Menerapkan metode *Profile Matching* ke dalam sebuah sistem yang dapat merekomendasikan tentor yang sesuai untuk siswa les.
2. Membuat suatu sistem dan aplikasi yang dapat mengirimkan pesan penawaran siswa les kepada tentor dengan *push notification* dengan menggunakan teknologi GCM.

1.5 Manfaat

Hasil dari penelitian ini diharapkan memberikan manfaat bagi administrator bimbingan belajar, antara lain:

1. Mempermudah administrator dalam mendapatkan perekomendasi tutor melalui metode *Profile Matching*.
2. Mempermudah administrator dalam menawarkan siswa les serta mendapat konfirmasi persetujuan dari tutor yang bersedia.

1.6 Sistematika Penulisan

Sistematika penulisan memberikan gambaran dan uraian dari penyusunan skripsi secara garis besar yang meliputi beberapa bab, antara lain:

BAB I PENDAHULUAN

Bab ini menguraikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat dari penelitian, dan sistematika penulisan mengenai perlunya penelitian tentang “Rancang Bangun Aplikasi Rekomendasi Tutor Bimbingan Belajar Menggunakan Metode *Profile Matching* Berbasis Android”.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori pendukung dan bahan penelitian mengenai teknologi yang diimplementasikan pada penelitian ini. Teori yang diambil mengenai metode *Profile Matching* dan teknologi *push notification*.

BAB III METODOLOGI PENELITIAN

Bab ini menguraikan tentang metode dan langkah kerja yang dilakukan dalam penulisan skripsi yang terdiri dari studi literatur, metode analisis kebutuhan, perancangan sistem, implementasi, metode dan pengambilan kesimpulan serta metode lain yang relevan dengan penelitian ini.

BAB IV PERANCANGAN

Perancangan pada bab ini terdiri atas 2 bagian, yaitu analisis kebutuhan dan perancangan perangkat lunak baik dalam segi metode *Profile Matching* maupun sistem perangkat lunak tersebut. Analisis kebutuhan dipresentasikan dengan menggunakan diagram *Use Case* dan diagram *Activity*, sedangkan perancangan perangkat lunak disesuaikan dengan analisis kebutuhan sistem dan akan dirancang arsitektur sistem, diagram *Sequence*, diagram *Class* dan perancangan antarmuka.

BAB V IMPLEMENTASI

Bab ini akan membahas tentang implementasi metode dan perangkat lunak. Metode yang diimplementasikan pada sistem ini adalah metode *Profile Matching* dan implementasi perangkat lunak berupa implementasi teknologi *push notification*.

BAB VI PENGUJIAN

Bab ini membahas mengenai teknik dan metode pengujian yang dilakukan pada penelitian ini. Pengujian yang dilakukan pada penelitian ini dengan menggunakan *black box* dan pengujian akurasi.

BAB VII PENUTUP

Pada bab ini membahas tentang kesimpulan dan saran dari pelaksanaan penelitian “Rancang Bangun Aplikasi Rekomendasi Tentor Bimbingan Belajar Menggunakan Metode *Profile Matching* Berbasis Android” ini.

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Pustaka

Penelitian yang menjadi referensi pertama adalah penelitian yang berjudul “Sistem Pendukung Keputusan Seleksi Penerimaan Asisten Praktikum Menggunakan Metode *Profile Matching* (Studi Kasus Prodi Teknik Informatika Universitas Brawijaya)”. Dalam metode ini dilakukan perhitungan dari masing-masing kriteria yang akan ditentukan bobotnya. Setelah itu, nilai akhir dari perhitungan metode *Profile Matching* ini akan menentukan urutan dari calon asisten praktikum yang paling direkomendasikan. Parameter yang digunakan untuk menentukan hasil adalah tes individu yang meliputi tes tulis, *microteaching*, wawancara dan penilaian kepribadian. Hasil dari penelitian tersebut adalah urutan calon asisten praktikum dari yang paling direkomendasikan [NIN-13].

Berikutnya Agus Santoso dan Andreas Handoyo membuat paper dari hasil penelitiannya yang berjudul “Pembuatan Aplikasi *Mobile Broadcast* Informasi Perkuliahan Berbasis Android”. Dalam penelitian ini, digunakan teknologi *push notification* yang diimplementasikan dengan *Google Cloud Messaging* (GCM). Aplikasi dibuat dalam dua bentuk yaitu berupa aplikasi android dan website. Website digunakan sebagai master saja dimana di dalamnya hanya terdapat fitur *add*, *edit* dan *delete*, sedangkan dalam aplikasi android yang merupakan pengaplikasian dari GCM yang dapat menerima informasi secara *realtime* dan *update* ketika memiliki koneksi internet [ASA-13].

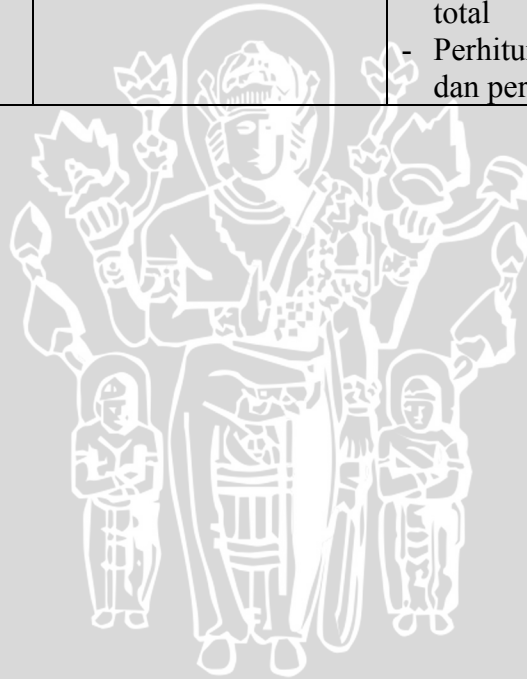
Aplikasi android yang akan dibangun dalam penelitian ini merupakan aplikasi yang menerapkan metode *Profile Matching* dan fitur *push notification* yang memanfaatkan teknologi GCM. Metode ini diharapkan dapat membantu administrator bimbingan belajar dalam mengambil keputusan untuk memilih tutor yang sesuai dengan anak les. Selain itu, diharapkan aplikasi ini dapat mempermudah administrator dalam memberikan informasi berupa penawaran siswa les kepada tutor dengan menggunakan *push notification*. Parameter yang digunakan dalam merekomendasikan tutor adalah jenis kelamin, waktu kosong, jenjang yang dikuasai dan mata pelajaran yang dikuasai.

Tabel 2. 1 Perbandingan Data

No	Judul	Objek	Metode	Output
		Input dan Parameter	Proses	Hasil Penelitian
1	Sistem Pendukung Keputusan Seleksi Penerimaan Asisten Praktikum Menggunakan Metode <i>Profile Matching</i> (Studi Kasus Prodi Teknik Informatika Universitas Brawijaya) [NIN-13]	Kriteria tes: <ul style="list-style-type: none"> - Tes tulis - <i>Microteaching</i> - Wawancara - Penilaian Kepribadian 	Metode: <i>Profile Matching</i> Proses: <ul style="list-style-type: none"> - Perhitungan pemetaan Gap kompetensi - Perhitungan dan pengelompokkan <i>core factor</i> dan <i>secondary factor</i> - Penghitungan nilai total - Perhitungan nilai akhir dan perangkingan 	Rincian nilai tiap kriteria dan nilai akhir tiap calon asisten yang diurutkan dari nilai akhir tertinggi.
2	Pembuatan Aplikasi <i>Mobile Broadcast</i> Informasi Perkuliahan Berbasis Android [ASA-13].	Informasi perkuliahan dari sisi server	Teknologi <i>Google Cloud Messaging</i> (GCM)	Pesan berubah <i>push notification</i> di sisi <i>client</i> pengguna android.
3	Rancang Bangun Aplikasi Rekomendasi Tutor Bimbingan Belajar Menggunakan Metode <i>Profile Matching</i> Berbasis Android [usulan].	Kriteria: <ul style="list-style-type: none"> - Jenis kelamin - Waktu kosong - Jenjang yang dikuasai - Mata pelajaran yang 	Metode: <ol style="list-style-type: none"> 1. Teknologi <i>Google Cloud Messaging</i> (GCM) 2. <i>Profile Matching</i> 	Urutan rekomendasi tutor yang sesuai dengan anak les dan mengirimkan <i>push notification</i> sesuai

		<p>dikuasai</p>	<p>Proses:</p> <ul style="list-style-type: none"> - Perhitungan pemetaan Gap kompetensi - Perhitungan dan pengelompokkan <i>core factor</i> dan <i>secondary factor</i> - Penghitungan nilai total - Perhitungan nilai akhir dan perangkingan 	<p>rekomendasi tadi.</p>
--	--	-----------------	---	--------------------------

Sumber : [NIN-13],[ASA-13]



2.2 Bimbingan Belajar

Menurut Undang-undang sistem pendidikan Nasional tahun 1989, pendidikan dilaksanakan dalam bentuk bimbingan, pengajaran, dan latihan. Bimbingan atau membimbing memiliki dua makna yaitu bimbingan secara umum yang mempunyai arti sama dengan mendidik atau menanamkan nilai-nilai, membina moral, mengarahkan siswa supaya menjadi orang baik. Sedangkan makna bimbingan yang secara khusus yaitu sebagai suatu upaya atau program membantu mengoptimalkan perkembangan siswa [IND-11]. Tidak setiap siswa memiliki kemampuan untuk mengatasi persoalan yang terkait dengan belajar. Seringkali kemampuan mengatasi persoalan itu harus difasilitasi oleh guru pembimbing untuk dapat direalisasikan. Guru pembimbing memiliki kesempatan yang luas untuk bersama dengan siswanya mengembangkan berbagai kemampuan potensial yang diharapkan menunjang kegiatan belajarnya [SUH-13].

Dengan demikian, bimbingan belajar adalah suatu proses pemberian bantuan dari guru pembimbing kepada siswa dengan cara mengembangkan suasana belajar yang kondusif dan menumbuhkan kemampuan agar siswa terhindar dari dan atau dapat mengatasi kesulitan belajar yang mungkin dihadapinya sehingga mencapai hasil belajar yang optimal [SUH-13].

2.3 Profile Matching

Menurut penelitian yang dilakukan oleh Ilman Fahma Dwijaya dengan judul “Sistem Pendukung Keputusan Kenaikan Jabatan pada PT. Sysmex Menggunakan Metode *Profile Matching*”, diketahui bahwa *Profile Matching* secara garis besar merupakan proses membandingkan antara kompetensi individu kedalam kompetensi jabatan sehingga dapat diketahui perbedaan kompetensinya atau biasa disebut Gap, semakin kecil Gap yang dihasilkan maka bobot nilainya semakin besar yang berarti memiliki peluang lebih besar untuk karyawan menempati posisi tersebut [DWI-13].

Dalam penelitian ini, objek yang diteliti adalah tentor bimbingan belajar yang dibandingkan dengan data siswa les. Jadi, *Profile Matching* dalam penelitian ini merupakan proses membandingkan antara data tentor bimbingan belajar dengan data siswa les sehingga dapat diketahui perbedaan kemampuannya yang sering

disebut Gap. Semakin kecil Gap yang dihasilkan maka bobot nilainya semakin besar yang berarti memiliki peluang lebih besar untuk tentor dapat direkomendasikan dalam mengajar siswa les. Metode *Profile Matching* terdiri dari beberapa langkah berikut [KUS-07]:

2.3.1 Penentuan Gap Kompetensi

Dalam penelitian Kusrini (2006), Gap adalah perbedaan antara profil jabatan dengan profil karyawan, yang dapat dirumuskan sebagai berikut ini [KUS-07].

$$Gap = Profil Karyawan - Profil Jabatan \dots\dots\dots(2-1)$$

Berdasarkan keterangan diatas, maka dalam penelitian ini, Gap dapat didefinisikan sebagai perbedaan *value* profil tentor yang ada dengan *value* profil siswa les. Pernyataan ini dapat dirumuskan sebagai berikut ini.

$$Gap = Profil Tmentor - Profil Siswa Les \dots\dots\dots(2-2)$$

2.3.2 Pembobotan

Setelah didapatkan tiap Gap masing-masing tentor maka tiap profil tentor diberi bobot dengan patokan tabel bobot nilai Gap. Tabel 2.2 di bawah ini merupakan contoh tabel bobot nilai Gap [KUS-07].

Tabel 2. 2 Contoh Tabel Bobot Nilai Gap

No	Selisih	Bobot Nilai	Keterangan
1	0	5	Tidak ada selisih (kompetensi sesuai dengan yang dibutuhkan)
2	1	4.5	Kompetensi individu kelebihan 1 tingkat
3	-1	4	Kompetensi individu kekurangan 1 tingkat
4	2	3.5	Kompetensi individu kelebihan 2 tingkat
5	-2	3	Kompetensi individu kekurangan 2 tingkat
6	3	2.5	Kompetensi individu kelebihan 3 tingkat
7	-3	2	Kompetensi individu kekurangan 3 tingkat
8	4	1.5	Kompetensi individu kelebihan 4 tingkat
9	-4	1	Kompetensi individu kekurangan 4 tingkat

Sumber: [KUS-07]



Dalam penelitian ini, setelah diperoleh Gap pada masing-masing tentor, setiap profil tentor diberi bobot nilai sesuai ketentuan tabel bobot Gap tiap-tiap kriteria. Ketentuan tabel bobot Gap tiap kriteria berbeda-beda, tergantung kondisi bobot profil kriteria tersebut. Detail tabel bobot Gap tiap kriteria akan dibahas lebih lanjut pada Bab IV.

2.3.3 Perhitungan dan Pengelompokkan *Core Factor* dan *Secondary Factor*

Setelah menentukan bobot nilai Gap untuk semua aspek dengan cara yang sama, setiap aspek dibagi lagi menjadi dua kelompok yaitu kelompok *core factor* dan *secondary factor*. Contoh rumus perhitungan *core factor* dapat ditunjukkan pada persamaan (2-3) berikut [KUS-07]:

$$NCF = \frac{\sum NC(i,s,p)}{\sum IC} \dots\dots\dots (2-3)$$

Keterangan:

NCF : Nilai rata-rata *core factor*

NC(i,s,p) : Jumlah total nilai *core factor* (Intelektual, Sikap kerja, Perilaku)

IC : Jumlah item *core factor*

Kriteria yang digunakan pada persamaan (2-3) adalah aspek intelektual, sikap kerja dan perilaku. Sedangkan pada penelitian ini ada empat kriteria yang digunakan, yaitu jenis kelamin, waktu kosong, jenjang yang dikuasai, mata pelajaran yang dikuasai. Faktor waktu kosong tentor dan mata pelajaran yang dikuasai merupakan faktor utama dalam penelitian ini sehingga dikategorikan sebagai *core factor*. Rumus perhitungan *core factor* dalam penelitian ini dapat ditunjukkan pada persamaan (2-4) berikut ini.

$$NCF = \frac{\sum NC(w,m,k,j)}{\sum IC} \dots\dots\dots (2-4)$$

Keterangan :

NCF : Nilai rata-rata *core factor*

NC(w,m,k,j) : Jumlah total nilai *core factor* (waktu kosong, mata pelajaran yang dikuasai, jenis kelamin, dan jenjang yang dikuasai)

IC : Jumlah item *core factor*

Perhitungan *secondary factor* pada penelitian Kusrini, ditunjukkan pada persamaan (2-5) berikut ini [KUS-07]:

$$NSF = \frac{\sum NS(i,s,p)}{\sum IS} \dots\dots\dots(2-5)$$

Keterangan :

NSF : Nilai rata-rata *secondary factor*

NS(i,s,p) : Jumlah total nilai *secondary factor* (Intelektual, Sikap Kerja, Perilaku)

IS : Jumlah item *secondary factor*

Pada penelitian ini, jenis kelamin dan pendidikan yang dikuasai tentor merupakan faktor selain faktor utama yang mendukung terpilihnya tentor. Rumus perhitungan *secondary factor* dalam penelitian ini dapat ditunjukkan dalam persamaan (2-6) berikut ini.

$$NSF = \frac{\sum NS(w,m,k,j)}{\sum IS} \dots\dots\dots(2-6)$$

Keterangan :

NSF : Nilai rata-rata *secondary factor*

NS(w,m,k,j) : Jumlah total nilai *secondary factor* (waktu kosong, mata pelajaran yang dikuasai, jenis kelamin, dan jenjang yang dikuasai)

IS : Jumlah item *secondary factor*

2.3.4 Perhitungan Nilai Total

Dari hasil perhitungan dari tiap aspeknya, kemudian dihitung nilai total berdasarkan presentase dari *core factor* dan *secondary factor* yang diperkirakan berpengaruh terhadap kinerja tiap-tiap profil. Perhitungan total menurut penelitian Kusrini, dapat dilihat dalam persamaan berikut ini.



$$(x)\%NCF(i, s, p) + (x)\%NSF(i, s, p) = N(i, s, p) \dots\dots\dots(2-7)$$

Keterangan :

- (i,s,p) : Intelektual, Sikap kerja, Perilaku
- NCF(i,s,p) : Nilai rata-rata *core factor*
- NSF(i,s,p) : Nilai rata-rata *secondary factor*
- N(i,s,p) : Nilai Total dari aspek
- (x)% : Nilai persen yang diinputkan

Sesuai persamaan (2-7), maka rumus perhitungan nilai total dalam penelitian ini adalah sebagai berikut :

$$(x)\%NCF(w, m, k, j) + (x)\%NSF(w, m, k, j) = N(w, m, k, j) \dots\dots\dots(2-8)$$

Keterangan :

- (w,m,k,j) : Waktu kosong, Mata pelajaran yang dikuasai, Jenis Kelamin, dan jenjang yang dikuasai
- NCF(w,m,k,j) : Nilai rata-rata *core factor*
- NSF(w,m,k,j) : Nilai rata-rata *secondary factor*
- N(w,m,k,j) : Nilai Total dari aspek
- (x)% : Nilai persen yang ditentukan

2.3.5 Perhitungan Penentuan Ranking

Hasil akhir dari proses *Profile Matching* pada penelitian Kusrini adalah ranking dari kandidat yang diajukan untuk mengisi suatu jabatan tertentu. Penentuan ranking mengacu pada hasil perhitungan tertentu. Untuk perhitungan penentuan ranking itu ditunjukkan pada persamaan (2-9) berikut ini [KUS-07].

$$Ranking = (x)\%Ni + (x)\%Ns + (x)\%Np \dots\dots\dots(2-9)$$

Keterangan :

- Ni : Nilai Kapasitas Intelektual
- Ns : Nilai Sikap Kerja
- Np : Nilai Perilaku
- (x)% : Nilai persen yang diinputkan



Berdasarkan persamaan (2-9), maka rumus perhitungan ranking pada penelitian ini dapat ditunjukkan pada persamaan (2-10) berikut ini.

$$\text{Ranking} = (k)\%NKT + (w)\%NWT + (j)\%NJT + (m)\%NMT \dots(2-10)$$

Keterangan :

- NKT : Nilai Jenis Kelamin
NWT : Nilai Waktu Kosong
NJT : Nilai jenjang yang dikuasai
NMT : Nilai mata pelajaran yang dikuasai
(k,w,j,m)% : Nilai persen yang ditentukan

2.4 Native App

Native App adalah aplikasi smartphone yang dikodekan dalam bahasa pemrograman tertentu, seperti Objective C untuk iOS dan Java untuk sistem operasi Android. *Native App* memberikan kinerja yang cepat dan mempunyai tingkat kehandalan yang tinggi. Selain itu, pengguna dapat menggunakan beberapa aplikasi tanpa koneksi internet. Namun, ini adalah jenis aplikasi yang mahal dalam pengembangannya karena terikat pada satu jenis sistem operasi [COR-13]. Tools untuk membuat aplikasi *native* di Android adalah sebagai berikut [MAR-13]:

1. ADT (Android Development Tools) for eclipse
2. Android SDK Tools
3. Android Emulator

Pada Android, tool-tools diatas disediakan secara gratis oleh Android.inc. (yang sudah diakuisisi oleh google inc. Kelebihan dari *Native App* adalah dapat mengakses semua fitur yang dimiliki oleh sistem operasi yang bersangkutan seperti GPS, alarm dan camera. Perlu diketahui bahwa GPS, alarm, dan camera ini tidak dapat digunakan pada *Web App* (kecuali dengan bantuan framework tertentu seperti Sencha Touch dan *Phonegap*) [MAR-13].

2.5 Java

Java merupakan bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Dikembangkan oleh Sun Microsystems dan diterbitkan tahun 1995 [NUR-10]. Java adalah bahasa pemrograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut kelas. Kelas terdiri atas metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para pemrogram Java banyak mengambil keuntungan dari kumpulan kelas di pustaka kelas Java, yang disebut dengan *Java Application Programming Interface* (API). Kelas-kelas ini diorganisasikan menjadi sekelompok yang disebut paket (*package*) [DID-09].

Java API telah menyediakan fungsionalitas yang memadai untuk menciptakan applet dan aplikasi canggih. Jadi ada dua hal yang harus dipelajari dalam Java, yaitu mempelajari bahasa Java dan bagaimana mempergunakan kelas pada Java API. Kelas merupakan satu-satunya cara menyatakan bagian eksekusi program, tidak ada cara lain. Pada Java program javac untuk mengkompilasi file kode sumber Java menjadi kelas-kelas bytecode. File kode sumber mempunyai ekstensi *.java. Kompilator javac menghasilkan file bytecode kelas dengan ekstensi *.Class. Interpreter merupakan modul utama sistem Java yang digunakan aplikasi Java dan menjalankan program bytecode Java. Tipe variabel Java mempunyai ukuran sama di semua platform sehingga variabel bertipe integer berukuran sama tidak peduli dimana program java dikompilasi. Begitu telah tercipta file .Class dengan menggunakan kompilator Java di platform manapun, maka file .Class tersebut dapat dijalankan di platform manapun. Jadi “dimanapun dibuat, dimanapun dapat dijalankan”. Slogan ini biasa diringkas sebagai *Write Once, Run Anywhere* (WORA) [DID-09].

2.6 JSON

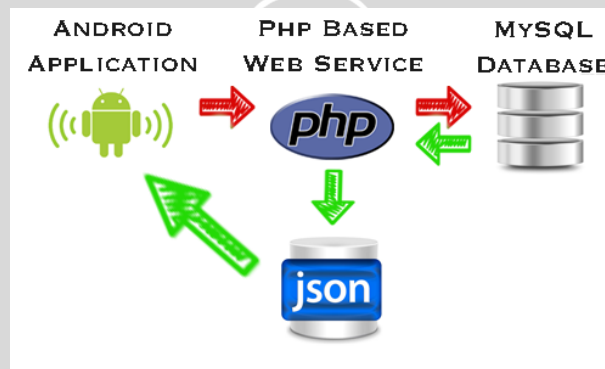
JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa

pemrograman apapun karena menggunakan gaya bahasa yang umum. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data [JSO-13].

JSON terbuat dari dua struktur sebagai berikut [JSO-13]:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
2. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*Sequence*).

Cara kerja JSON dapat dilihat pada gambar 2.1 berikut ini.



Gambar 2. 1 Cara Kerja JSON
Sumber: [TRA-13]

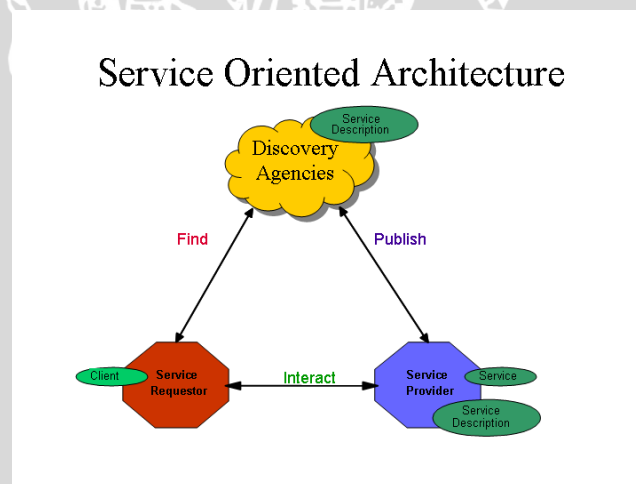
Format data JSON mempunyai aturan sebagai berikut [SRO-13]:

1. Object adalah satu set nama/nilai yang tidak terurut. Penulisan object dimulai dengan tanda { (*left brace*) dan diakhiri dengan tanda } (*right brace*). Setiap nama diikuti oleh tanda : (*colon*) dan pasangan nama/nilai dipisahkan dengan tanda , (*comma*).
2. Array adalah sekumpulan nilai yang teratur. Penulisan sebuah array dimulai dengan tanda [(*left bracket*) dan diakhiri dengan tanda] (*right bracket*). Nilai dipisahkan menggunakan tanda , (*comma*).
3. Nilai dapat berupa string dalam tanda kutip, atau number (angka), TRUE atau FALSE atau NULL, sebuah object atau sebuah array.

4. String adalah rangkaian atau urutan karakter unicode yang berada dalam tanda kutip, bisa juga hanya berisi karakter kosong, menggunakan tanda \ (*backslash*) untuk escape.
5. Number sangat mirip dengan definisi pada bahasa C atau Java, hanya saja tipe bilangan oktal dan heksadesimal tidak digunakan.

Peran JSON bukan hanya memanggil dan menampilkan data saja, JSON juga bisa untuk memasukan data ke dalam database dan menampilkannya secara dinamis dalam suatu program. Objek yang ada pada JSON secara keseluruhan merupakan array, dimana dari setiap objek tersebut JSON merubah setiap tipe data lalu menampilkannya dalam jenis string agar mudah terbaca oleh program. Objek pada JSON memiliki *identifier* (identitas), dimana *identifier* ini nantinya dipanggil oleh program agar menampilkan data dalam jenis string yang bisa dibaca dan dieksekusi oleh program apabila kondisi data memiliki perubahan atau tereksekusi secara *realtime* [ZAP-13].

2.7 Web service



Gambar 2. 2 Arsitektur Web service
Sumber: [SAP-13]

Web service adalah aplikasi sekumpulan data, perangkat lunak (*software*) atau bagian dari perangkat lunak yang dapat diakses secara remote oleh berbagai piranti dengan sebuah perantara tertentu. Secara umum, *web service* dapat diidentifikasi dengan menggunakan URL seperti hanya web pada umumnya. Namun yang membedakan *web service* dengan web pada umumnya adalah

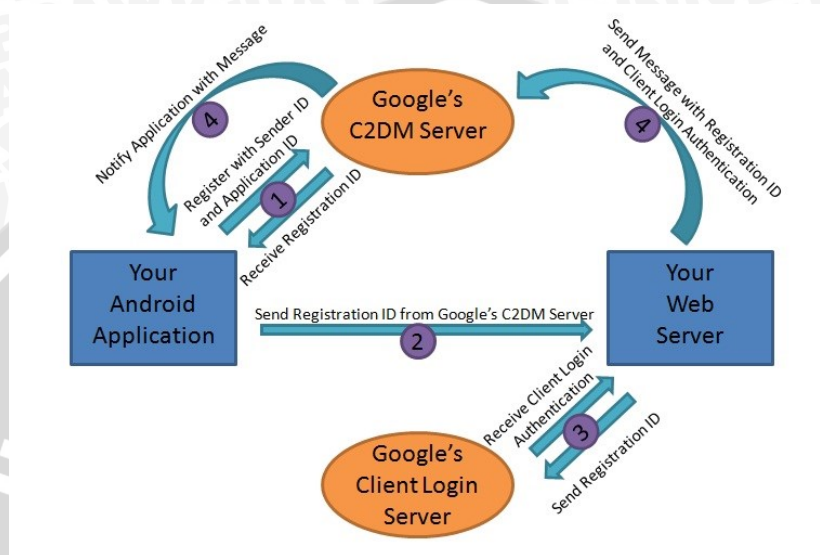
interaksi yang diberikan oleh *web service*. Berbeda dengan URL web pada umumnya, URL *web service* hanya mengandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi. *Web service* dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah database ditanamkan, dibuat dalam bahasa apa sebuah aplikasi yang mengkonsumsi data, dan di platform apa sebuah data itu dikonsumsi. *Web service* mampu menunjang interoperabilitas. Sehingga *web service* mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada [SAP-13].

Web pada umumnya digunakan untuk melakukan respon dan *request* yang dilakukan antara *client* dan server. Sebagai contoh, seorang pengguna layanan web tertentu mengetikkan alamat url web untuk membentuk sebuah *request*. *Request* akan sampai pada server, diolah dan kemudian disajikan dalam bentuk sebuah respon. Sedangkan pada *web service* hubungan antara *client* dan server tidak terjadi secara langsung. Hubungan antara *client* dan server dihubungkan oleh file *web service* dalam format tertentu. Sehingga akses terhadap database akan ditangani tidak secara langsung oleh server, melainkan melalui perantara yang disebut sebagai *web service*. Peran dari *web service* ini akan mempermudah distribusi sekaligus integrasi database yang tersebar di beberapa server sekaligus [SAP-13].

2.8 Google Cloud Messaging (GCM)

Google Cloud Messaging (GCM) untuk Android adalah layanan yang dapat membantu *developer* untuk mengirim data dari server ke aplikasi Android buatan mereka pada perangkat Android setiap kali data baru tersedia. Mengintegrasikan GCM pada aplikasi android kita dapat meningkatkan pengalaman pengguna dan menyimpan banyak tenaga baterai perangkat Android [HER-14]. Ini bisa menjadi pesan ringan memberitahu aplikasi Android bahwa ada data baru yang akan diambil dari server (misalnya, film diunggah oleh seorang teman), atau bisa juga pesan yang berisi sampai dengan 4 KB data payload (sehingga aplikasi seperti instant messaging dapat mengkonsumsi pesan langsung). Layanan GCM menangani semua aspek antrian pesan dan pengiriman ke aplikasi target Android

berjalan pada perangkat target. GCM memungkinkan aplikasi android untuk mngirimkan pesan kepada server untuk mem**roadcast** sebuah notifikasi kepada seluruh *client* yang ada. Hanya membutuhkan sebuah *account* Gmail maka akan langsung dapat menggunakan fasilitas GCM ini [ASA-13]. Cara kerja GCM akan digambarkan pada gambar 2.3 berikut ini.



Gambar 2. 3 Cara Kerja GCM
Sumber: [KAS-12]

Pertama, aplikasi akan perlu mengirim “query” (permintaan) untuk Server C2DM Google. "Query" ini memerlukan informasi tertentu seperti ID Pengirim (biasanya alamat e-mail) dan ID aplikasi ini (nama paket aplikasi). Dan Server C2DM Google ini akan mengirim kembali intent dengan baik pesan gagal atau ID Registrasi. Kemudian aplikasi perlu untuk mengirim ID pendaftaran ke server web, sehingga web server anda dapat menyimpan informasi ini untuk digunakan nanti. Dari titik pandang aplikasi, selain menerima pesan, tidak ada lagi yang perlu dilakukan [KAS-12].

Langkah-langkah lain dari proses ini adalah pada sisi server Web. Web server harus menggunakan ID pengirim dan password untuk mengambil otentikasi ‘Masuk’ *client* dari server ‘Masuk’ Google klien. Dengan ini Web server sekarang memiliki semua informasi yang diperlukan untuk mengirim pesan ke aplikasi. Web server kemudian akan menggunakan ID pendaftaran dan otentikasi klien login dengan pesan ke server yang dituju C2DM Google. Setelah ini Google akan menangani mengirimkan pemberitahuan ke aplikasi dan Anda hanya harus

memastikan aplikasi Anda melakukan ‘tindakan’ yang diperlukan ketika menerima pesan [KAS-12].

2.9 Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah suatu proses yang digunakan untuk mengidentifikasi ketepatan, kelengkapan dan mutu dari perangkat lunak dalam ilmu komputer yang dikembangkan [SUS-13]. Teknik pengujian yang digunakan pada penelitian ini adalah sebagai berikut :

2.9.1 *Black-Box* Testing

Black-box testing berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *black-box* bukan merupakan alternatif dari *white-box* tetapi merupakan pendekatan komplementer yang kemungkinan mampu mengungkap kelas kesalahan daripada metode *white-box*.

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut [AYU-09] :

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau database eksternal
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

Pada pengujian *black-box* cenderung diaplikasikan selama tahap akhir pengujian. Pengujian *black-box* memperhatikan struktur control, sehingga pengujian ini akan fokus pada domain informasi.

2.9.2 Pengujian Akurasi

Proses pengujian akurasi dilakukan untuk mengetahui performa dari metode pendukung keputusan pemilihan tutor yang sesuai untuk siswa les. Prosedur pengujiannya adalah dengan membandingkan antara hasil tutor yang

direkomendasikan oleh sistem menggunakan metode *Profile Matching* dengan rekomendasi dari pakar. Setelah diketahui jumlah rekomendasi yang cocok dan tidak cocok, maka dapat diketahui presentase untuk akurasi rekomendasi tentor bimbingan belajar ini dengan rumus sebagai berikut.

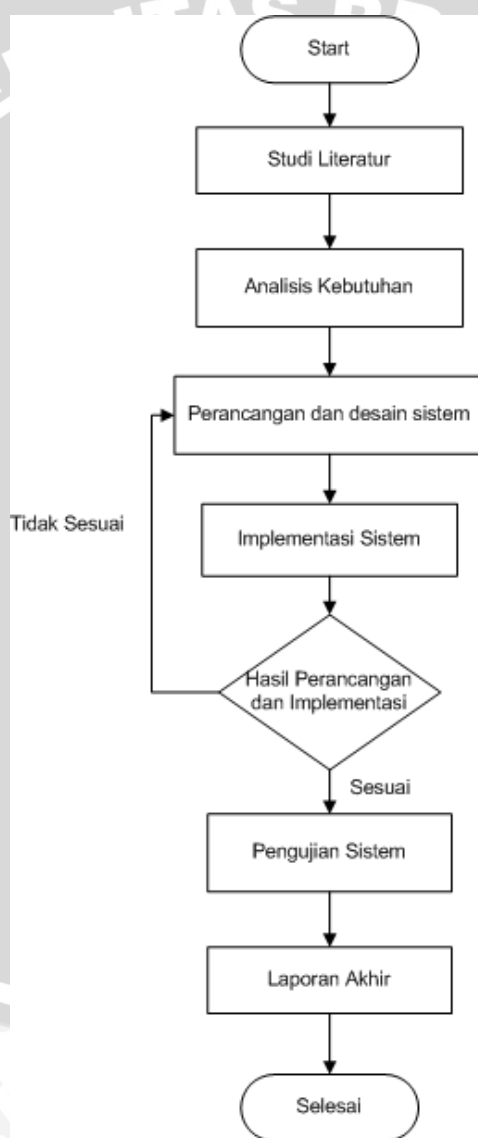
$$\text{Akurasi} = \frac{\text{Jumlah test case valid}}{\text{Jumlah test case diujikan}} \times 100\%$$



BAB III

METODOLOGI PENELITIAN

Bab ini akan menjelaskan langkah-langkah dan metode yang akan ditempuh dalam penyusunan skripsi, yaitu studi literatur yang digunakan sebagai landasan teori untuk penelitian ini, analisis kebutuhan, perancangan dan mendesain sistem, implementasi sistem, pengujian dan laporan akhir. Pada gambar 3.1 berikut akan dijelaskan metodologi penelitian dari sistem rekomendasi tutor bimbingan belajar dengan metode *Profile Matching* berbasis android ini.



Gambar 3. 1 Alur Metodologi Penelitian

3.1 Studi Literatur

Studi literatur dalam penelitian ini adalah untuk menjelaskan teori dari komponen yang akan digunakan dalam penelitian ini. Studi literatur termasuk ke dalam data sekunder dimana data ini telah ada sebelumnya seperti dari literatur atau buku. Studi literatur dalam penelitian ini adalah sebagai berikut:

1. Kajian pustaka dari penelitian yang sudah ada sebelumnya
2. Bimbingan Belajar
3. *Profile Matching*
4. *Native App*
5. Java
6. JSON
7. *Web service*
8. *Google Cloud Messaging (GCM)*
9. Pengujian Perangkat Lunak

3.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk mendapatkan semua kebutuhan yang diperlukan dari sistem yang akan dibangun. Kebutuhan yang dimaksud adalah kriteria rekomendasi, bobot kriteria, kebutuhan user dari segi administrator maupun tentor. Analisis kebutuhan sistem didapatkan melalui wawancara dengan narasumber Administrator *Sonic Class* Bimbingan Belajar Malang. Setelah mengetahui daftar kebutuhan tersebut, kebutuhan akan dijabarkan menggunakan diagram *Use Case*. *Use case* merupakan representasi relasi antara user dengan sistem. Selain itu, dalam analisis kebutuhan juga ada identifikasi aktor, dan daftar kebutuhan sistem serta kebutuhan perangkat keras dan perangkat lunak yang diperlukan sistem. Data yang digunakan berasal dari *Sonic Class* Bimbingan Belajar Malang.

3.3 Perancangan dan Desain Sistem

Perancangan sistem dimulai setelah tahap analisis kebutuhan telah terpenuhi. Perancangan dimulai dari hasil kebutuhan sistem berupa diagram *use case*, kemudian diidentifikasi dan dimodelkan dengan diagram *Sequence*. Setelah itu,

dilakukan identifikasi sistem rekomendasi tentor dengan mengidentifikasi *class* dan relasinya yang akan dimodelkan ke dalam diagram *class*. Perancangan basis data perlu dilakukan untuk mengetahui representasi data yang akan diolah oleh sistem ini. Kemudian yang terakhir adalah perancangan antarmuka sistem.

3.4 Implementasi Sistem

Implementasi sistem dilakukan dengan mengacu pada perancangan dan desain sistem. Implementasi sistem akan diterapkan dengan menggunakan bahasa pemrograman Java dan PHP. Pembuatan sistem akan dilakukan dengan menggunakan Android Studio dan Netbeans. Agar mempermudah pembuatan sistem maka akan dijalankan di emulator sebelum diterapkan secara langsung ke dalam perangkat bergerak.

3.5 Pengujian Sistem

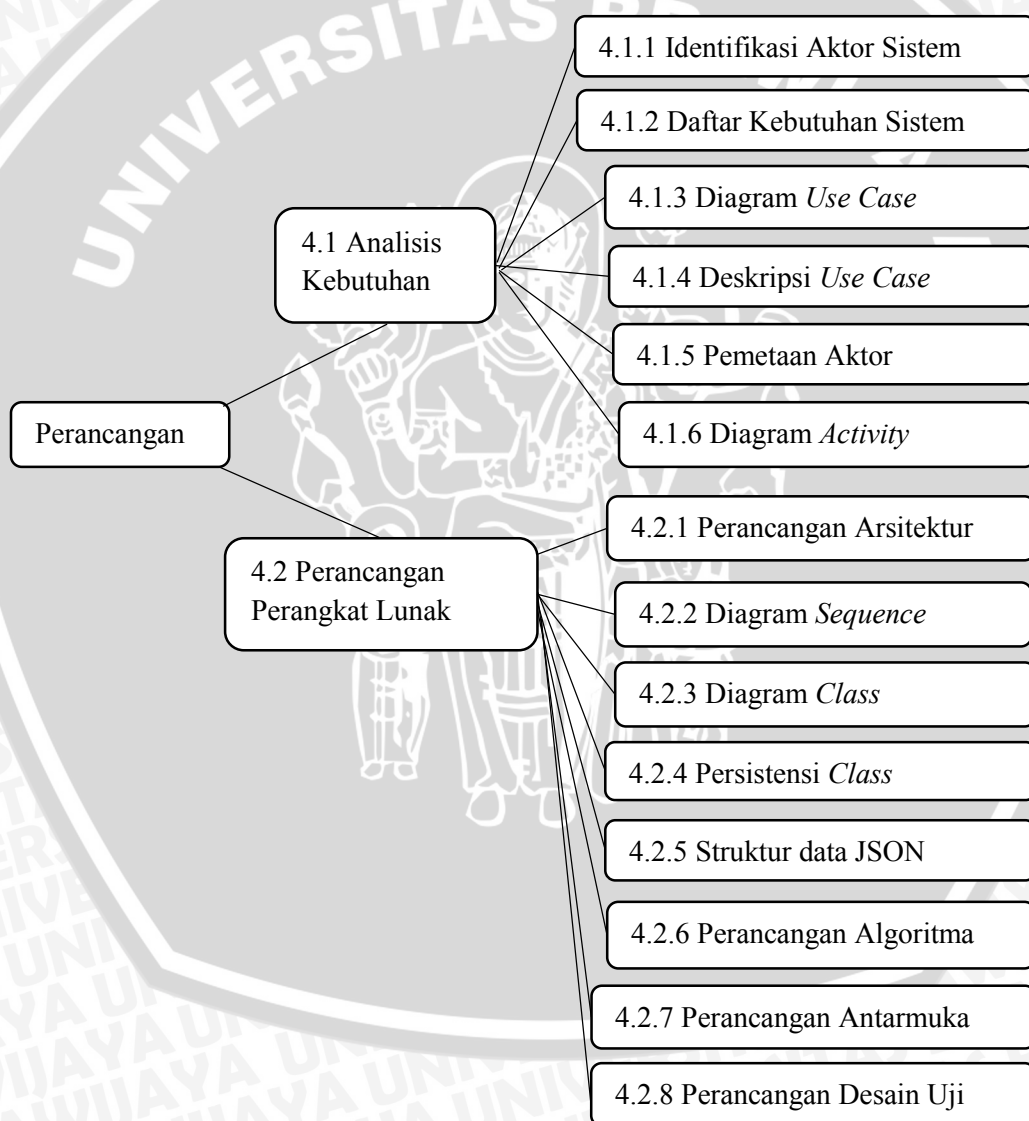
Pengujian sistem dilakukan agar dapat menunjukkan bahwa perangkat lunak mampu bekerja sesuai dengan spesifikasi dari kebutuhan penggunanya. Terdapat dua macam pengujian dalam penelitian ini yaitu *black-box* dan pengujian akurasi. Pengujian *black-box* ini berarti menguji fungsionalitas dari sistem. Pengujian dilakukan berdasarkan implementasi yang telah dibuat untuk memvalidasi kebenaran dari hasil output sistem. Pengujian akurasi dilakukan dengan membandingkan hasil output sistem dengan hasil perhitungan manual dan hasil output sistem dengan keputusan dari pakar.

3.6 Laporan Akhir

Setelah tahap-tahap di atas selesai, maka tahap terakhir adalah menyelesaikan pembuatan laporan akhir. Pada bagian akhir laporan terdapat bagian kesimpulan dan saran dimana kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Sedangkan saran dimaksudkan untuk memperbaiki kesalahan-kesalahan dan menyempurnakan penelitian serta untuk menjadi pertimbangan dalam pengembangan sistem selanjutnya.

BAB IV ANALISIS DAN PERANCANGAN

Pada bab ini penulis akan membahas tentang perancangan sistem rekomendasi tutor bimbingan belajar. Perancangan sistem terdiri dari analisis kebutuhan perangkat lunak dan perancangan perangkat lunak. Berikut ilustrasi pohon perancangan sistem rekomendasi tutor bimbingan belajar dengan menggunakan metode *Profile Matching* berbasis Android.



Gambar 4. 1 Pohon Perancangan Sistem

Gambar 4.1 di atas menjelaskan mengenai struktur perancangan pada sistem rekomendasi tentor. Pada perancangan akan dibahas mengenai analisis kebutuhan dan perancangan perangkat lunak. Dalam analisis kebutuhan, terdapat pembahasan mengenai identifikasi aktor sistem, daftar kebutuhan sistem, diagram *use case*, deskripsi *use case*, pemetaan aktor dan diagram *activity*. Sedangkan dalam perancangan perangkat lunak akan dibahas perancangan arsitektur, diagram *sequence*, diagram *class*, persistensi *class*, struktur JSON, perancangan algoritma, perancangan antarmuka, dan perancangan desain uji.

4.1 Analisis Kebutuhan Sistem

Analisis kebutuhan bertujuan untuk menentukan semua kebutuhan yang diperlukan untuk membangun sistem perangkat lunak. Berikut ini dijelaskan tentang Analisis Kebutuhan Sistem yang terdiri dari Identifikasi Aktor, Daftar Kebutuhan Sistem, Diagram *Use Case*, Deskripsi *Use Case*, Pemetaan Aktor dan Diagram *Activity*.

4.1.1 Identifikasi Aktor

Pada tahap ini bertujuan melakukan identifikasi terhadap user yang berhubungan dengan sistem yang dibuat. Penjelasan dari masing-masing user dapat dilihat pada Tabel 4.1 berikut ini.

Tabel 4. 1 Tabel Identifikasi Aktor

Aktor	Deskripsi
User Administrator Bimbingan Belajar	User admin dapat memanipulasi data siswa dan jadwal siswa. User admin dapat melihat hasil dari perhitungan kriteria untuk menentukan tentor yang sesuai dengan siswa les menggunakan metode <i>Profile Matching</i> . User admin juga berperan dalam mengirimkan <i>push notification</i> berdasarkan rekomendasi yang ditentukan menggunakan metode <i>Profile Matching</i> tadi.
User Tentor	User tentor dapat mendaftar akun baru tentor, menambahkan jadwal kosong, mata pelajaran yang dikuasai dan jenjang yang dikuasai serta menerima penawaran siswa les berupa <i>push notification</i> dari administrator dan dapat merespon penawaran tersebut.

4.1.2 Daftar Kebutuhan Sistem

Daftar kebutuhan sistem merupakan kebutuhan fungsional sistem dimana terdiri dari 4 kebutuhan fungsional. Kebutuhan fungsional ini diperoleh melalui proses tanya jawab mengenai analisis kebutuhan dari aktor administrator bimbingan belajar *Sonic Class*. Kebutuhan sistem dimodelkan ke dalam bentuk *use case*, dan setiap kebutuhan fungsional harus dilakukan pengujian terlebih dahulu sebelum digunakan oleh aktor yang bersangkutan.

Tabel 4. 2 Tabel Kebutuhan Fungsional

Identifikasi	Kebutuhan	Use Case
FU-01	Sistem mampu mengelola data tentor	Mengelola data tentor
FU-02	Sistem mampu mengelola data siswa les	Mengelola data siswa les
FU-03	Sistem mampu menampilkan ranking tentor yang telah direkomendasikan menggunakan metode <i>Profile Matching</i> dan dapat mengirimkan <i>push notification</i> .	Melihat ranking tentor yang direkomendasikan
FU-04	Sistem mampu merespon penawaran dari admin berupa jawaban bersedia atau tidak untuk mengajar siswa les yang ditawarkan.	Menjawab penawaran admin

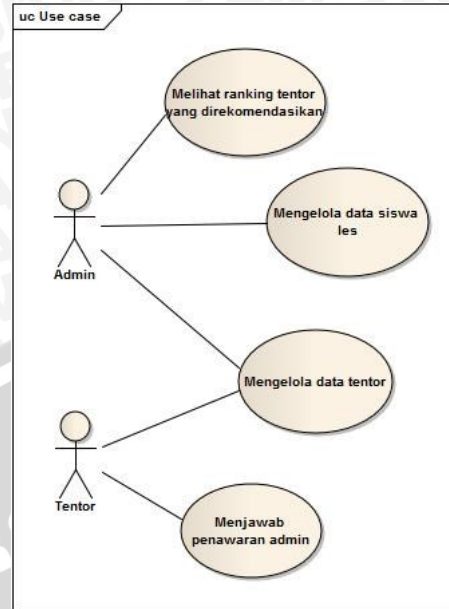
Kebutuhan Non Fungsional

1. Accuracy

Tingkat keakuratan sistem dalam merekomendasikan tentor bimbingan belajar menggunakan metode Profile Matching.

4.1.3 Diagram Use Case

Pemodelan diagram *use case* sistem diperoleh dari kebutuhan fungsional pada Tabel 4.2 diatas. Diagram *use case* dibuat sesuai dengan kebutuhan fungsional dan ini agar semua kebutuhan dari aktor dan interaksi antara satu atau lebih aktor dengan sistem yang dibuat dapat terpenuhi. Gambar 4.2 merupakan *use case* sistem yang terdiri dari 2 aktor, dimana masing-masing pengguna mempunyai peran dalam sistem.



Gambar 4. 2 Diagram Use Case

4.1.4 Deskripsi Use Case

Bagian deskripsi *Use Case* berisi nama *Use Case*, aktor yang menjalankan *Use Case*, tujuan dari *Use Case*, deskripsi mengenai *Use Case*, kondisi awal yang harus dipenuhi dan kondisi akhir yang diharapkan setelah berjalannya fungsional *Use Case*. Lalu aksi yang dilakukan aktor terhadap aplikasi yang dijalankan dan reaksi sistem atas aksi aktor tersebut ditampilkan. Berikut merupakan deskripsi dari masing-masing *Use Case* yang ada dalam sistem rekomendasi tentor bimbingan belajar ini.

Tabel 4. 3 Deskripsi Use Case Melihat ranking tentor yang direkomendasikan

Identifikasi					
Nama	Melihat ranking tentor yang direkomendasikan				
Deskripsi	<i>Use Case</i> ini menjelaskan aktor dapat melihat ranking tentor yang telah direkomendasikan menggunakan metode <i>Profile Matching</i> .				
Aktor	Admin				
Kondisi Awal	Sistem telah terpasang dan menampilkan halaman user yang sudah login menggunakan akun admin dan telah menginputkan data siswa les				
Kondisi Akhir	Sistem menampilkan ranking tentor yang sesuai dengan siswa les dan dapat mengirimkan <i>push notification</i>				
Normal Flow	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; text-align: center;">Admin</td> <td style="width: 50%; text-align: center;">Sistem</td> </tr> <tr> <td colspan="2">1. Memilih siswa les yang ingin dilihat jadwalnya</td> </tr> </table>	Admin	Sistem	1. Memilih siswa les yang ingin dilihat jadwalnya	
Admin	Sistem				
1. Memilih siswa les yang ingin dilihat jadwalnya					



		2. Menampilkan jadwal siswa yang dipilih
	3. Memilih jadwal yang ingin dilihat rekomendasi tentornya	
		4. a. Menampilkan detail siswa les beserta jadwal dan rekomendasi tentornya sesuai
	5. Menekan tombol kirim <i>push notification</i>	
		6a. Ada koneksi internet, <i>notification</i> terkirim ke rekomendasi tentor pertama
Alternative Flow		4b. Menampilkan detail siswa les beserta jadwal dan rekomendasi tentornya tidak sesuai
		6b. Tidak ada koneksi internet, <i>notification</i> tidak terkirim ke tentor

Pada Tabel 4.3 menjelaskan deskripsi *use case* melihat ranking tentor yang direkomendasikan. Deskripsi *use case* ini bertujuan untuk menampilkan daftar tentor sesuai dengan urutan perhitungan kriteria menggunakan metode *Profile Matching*. *Use case* ini hanya dapat dilakukan oleh admin. Kondisi yang harus dipenuhi sebelum *use case* ini dijalankan adalah sistem sudah terpasang di komputer administrator bimbingan belajar serta harus login menggunakan akun admin, serta sudah ada data siswa les yang akan ditentukan tentornya. Sedangkan kondisi akhir yang harus dipenuhi adalah sistem akan menampilkan daftar tentor sesuai dengan urutan yang paling direkomendasikan, dan dapat mengirimkan *push notification* sesuai dengan urutan tersebut.

Tabel 4. 4 Deskripsi Use Case Menjawab Penawaran Admin

Identifikasi	
Nama	Menjawab penawaran admin
Deskripsi	<i>Use Case</i> ini menjelaskan aktor dapat merespon penawaran siswa les yang telah diberikan admin
Aktor	Tentor
Kondisi Awal	Sistem telah terpasang di perangkat bergerak dan menampilkan halaman user yang sudah login menggunakan akun tentor
Kondisi	Respon dari tentor tersimpan di database

Akhir	Tentor	Sistem
Normal Flow	1a. Ada koneksi internet, tentor menerima <i>push notification</i> dari admin	
	2. Menjawab ya atau tidak atas penawaran admin	
		3a. Ada koneksi internet. Jika jawaban ya, maka data akan tersimpan di database. Jika jawaban tidak, maka <i>notification</i> akan dilanjutkan ke rekomendasi tentor berikutnya.
Alternative Flow	1b. Tidak ada koneksi internet. <i>Notification</i> tidak diterima oleh tentor	
		3b. Tidak ada koneksi internet. Maka data tidak tersimpan di database dan <i>notification</i> tidak dilanjutkan ke rekomendasi tentor berikutnya.

Deskripsi *use case* pada Tabel 4.4 menjawab penawaran admin menjelaskan sistem dapat melakukan respon terhadap penawaran yang dilakukan admin. *Use case* ini dapat dilakukan oleh tentor saja. Kondisi awal yang harus dipenuhi adalah aplikasi sudah harus terinstall di smartphone android. Sedangkan kondisi akhir yang diharapkan adalah jika respon yang diberikan tentor “tidak”, maka respon tidak akan tersimpan di database. Hal ini akan menyebabkan *notification* akan diteruskan ke tentor rekomendasi kedua sampai yang terakhir. Respon akan tersimpan di database jika respon yang diterima adalah “ya”, respon yang disimpan berupa nama tentor yang akan berada di jadwal siswa les yang bersangkutan.

4.1.5 Pemetaan Aktor

Pada pemetaan aktor ini akan dibahas mengenai aktor dan fitur apa saja yang dapat dilakukan oleh aktor tersebut.

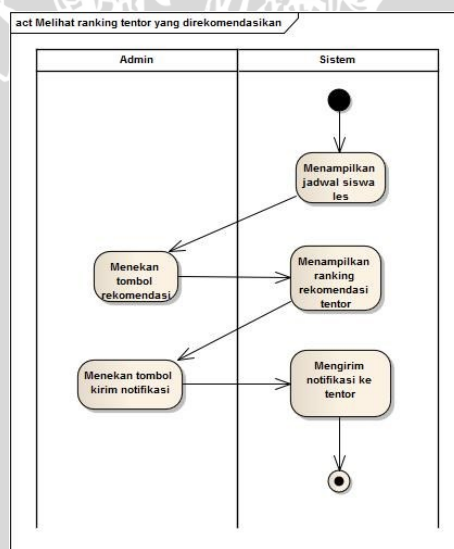
Tabel 4. 5 Pemetaan Aktor terhadap kebutuhan fungsional

ID Aktor	Aktor	FU-01	FU-02	FU-03	FU-04
A1	Administrator	√	√	√	
A2	Tentor	√			√

Pada Tabel 4.5 diatas menjelaskan tentang pemetaan daftar kebutuhan fungsional setiap aktor. Admin pada sistem merupakan administrator bimbingan belajar, dimana admin dapat melakukan fitur FU-01, FU-02, dan FU-03. Sedangkan aktor tentor dapat melakukan fitur FU-01, FU-02, FU-03 dan FU-04.

4.1.6 Diagram Activity

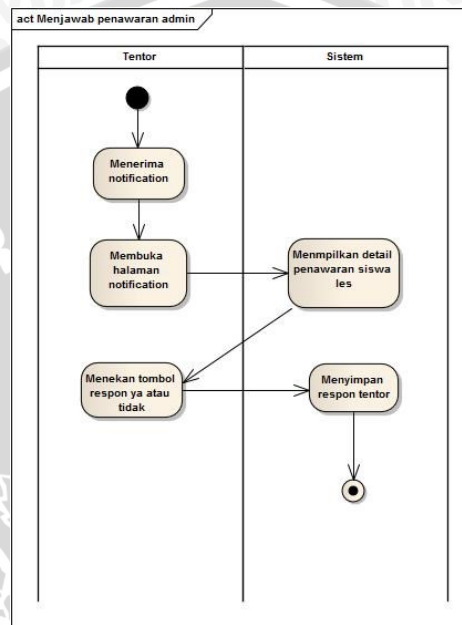
Diagram *Activity* memodelkan aktivitas antara user dan sistem yang berjalan sesuai dengan deskripsi *Use Case* yang telah dibuat sebelumnya. Adapun diagram *Activity* yang akan dibuat adalah *Activity* melihat ranking yang direkomendasikan dan menjawab penawaran dari admin. Berikut diagram *Activity* dari sistem rekomendasi tentor bimbingan belajar.



Gambar 4. 3 Diagram Activity Melihat Ranking Tentor

Berdasarkan Deskripsi Use Case pada Tabel 4.3 dapat dibuat sebuah Diagram *Activity* melihat ranking tentor. Pada Gambar 4.3 menjelaskan bagaimana ranking tentor dapat dilihat sampai mengirimkan *notification* ke tentor yang bersangkutan. Setelah login sebagai admin, admin membuka daftar siswa les

lalu menekan tombol “jadwal”, sehingga akan muncul jadwal siswa les yang bersangkutan. Dalam jadwal siswa les ada tombol rekomendasi yang bila ditekan akan muncul ranking tentor yang direkomendasikan dan tombol “kirim notification” untuk mengirimkan *notification* penawaran berupa data siswa les kepada tentor.



Gambar 4. 4 Diagram Activity Menjawab Penawaran Admin

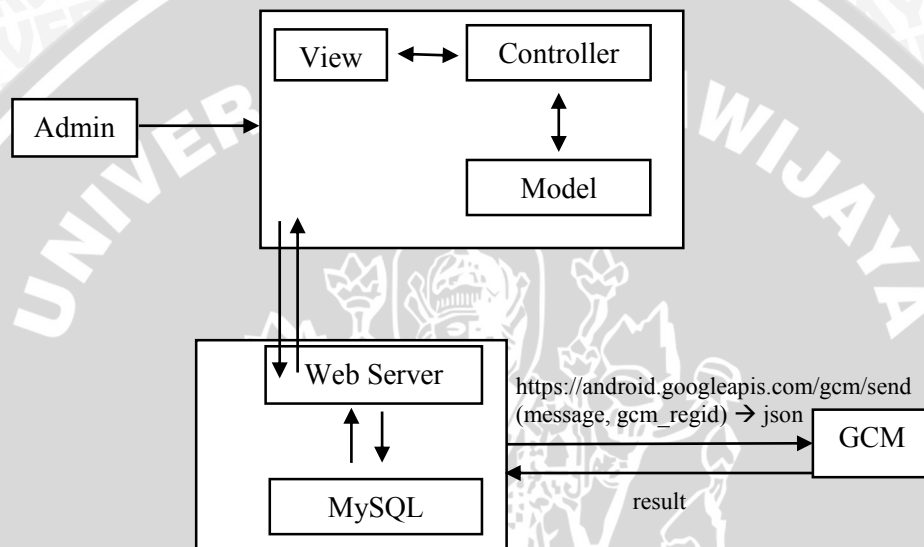
Berdasarkan Deskripsi Use Case yang telah dijabarkan pada Tabel 4.4 dapat dibuat Diagram *Activity* menjawab penawaran admin. Gambar 4.4 diatas menjelaskan proses tentor membuka dan merespon penawaran admin. Setelah *notification* dari admin masuk ke smartphone android tentor, tentor dapat membuka *notification* tersebut. *Notification* tersebut akan menampilkan detail siswa les beserta tombol “ya” dan “tidak” untuk merespon penawaran admin tersebut.

4.2 Perancangan Perangkat Lunak

Dalam perancangan perangkat lunak ini akan dibahas mengenai perancangan arsitektur sistem, diagram *Sequence*, diagram *Class*, persistensi *Class*, struktur data JSON, dan perancangan antarmuka.

4.2.1 Perancangan Arsitektur

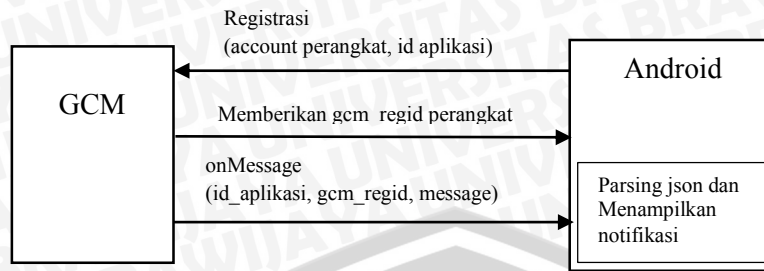
MVC merupakan metode yang digunakan untuk perancangan arsitektur pada sistem rekomendasi tutor bimbingan belajar ini. View merupakan antarmuka yang digunakan untuk interaksi antara user dan sistem, sedangkan controller merupakan jembatan yang menghubungkan view dan model, kemudian model berisikan data yang dibutuhkan untuk mengakses sistem. Gambar 4.5 dibawah ini merupakan perancangan arsitektur sistem dari segi server.



Gambar 4. 5 Perancangan Arsitektur Sisi Server

Setiap data yang terupdate diambil oleh web service, akan dikirim ke server GCM berupa message yang akan ditampilkan ke perangkat tutor dan berisi gcm_regid tutor yang dituju melalui url <https://android.googleapis.com/gcm/send> selaku GCM server untuk memberikan persetujuan pengiriman pesan serta meminta GCM server untuk mengirimkan pesan notifikasi kepada perangkat sesuai dengan registrasi id yang dimiliki masing-masing perangkat.

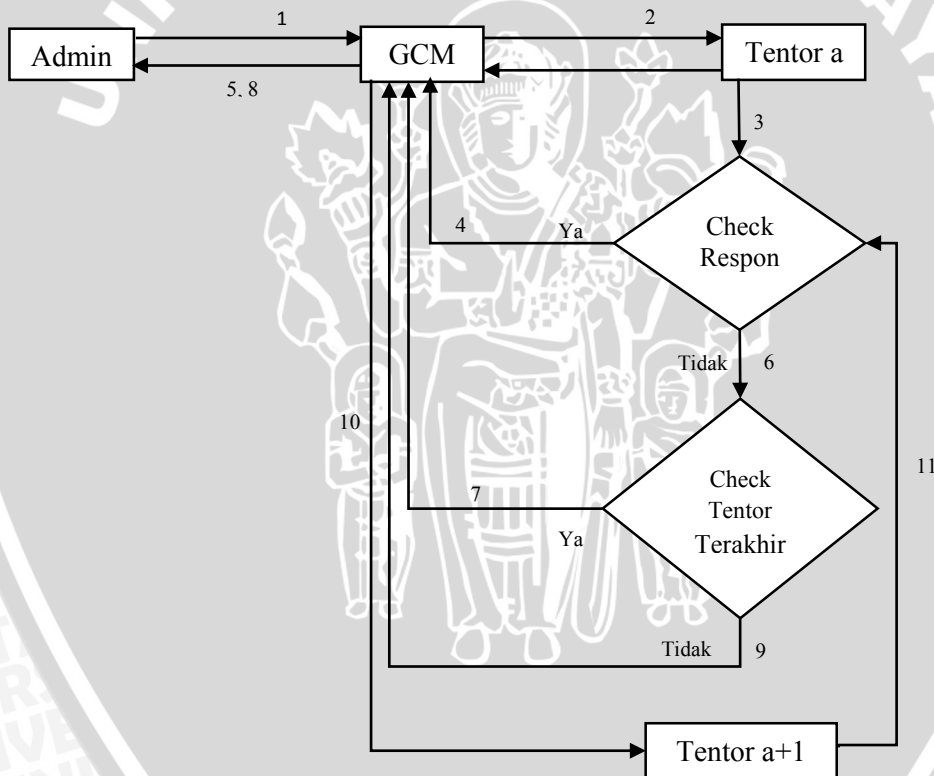
Sedangkan dari segi client, ketika server GCM mendapatkan *request* dari server sistem untuk mengirimkan message ke tutor dengan gcm_regid tertentu, maka GCM server akan meneruskan message tersebut ke perangkat dengan gcm_regid tersebut. Setelah itu tutor akan menerima notifikasi berupa penawaran siswa les dimana tutor dapat meresponnya dengan jawaban “ya” atau “tidak”. Detail alur pengiriman notifikasi sistem dapat dilihat pada Gambar 4.7.



Gambar 4. 6 Perancangan Arsitektur Sisi Client

4.2.2 Perancangan Fitur *Push Notification*

Pada bagian ini dibahas mengenai perancangan dari fitur push notification yang diterapkan dalam sistem rekomendasi tentor bimbingan belajar ini.



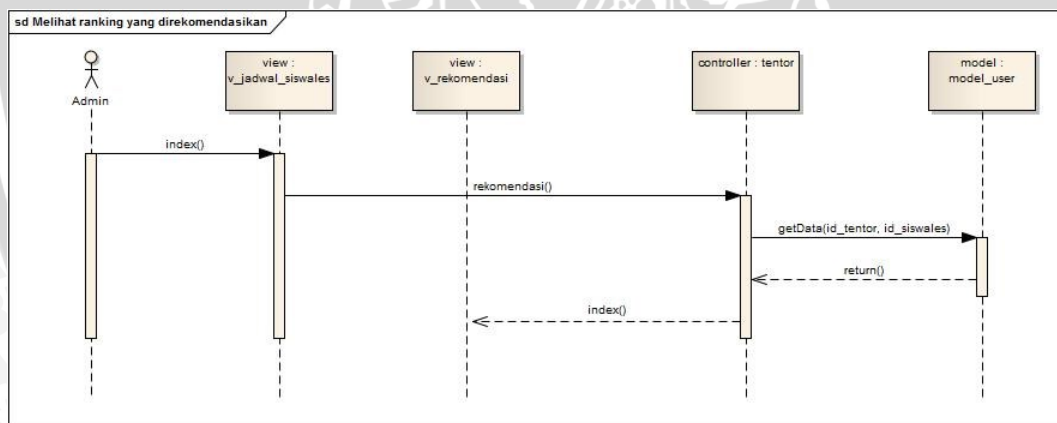
Gambar 4. 7 Perancangan Fitur Push Notification

Berdasarkan Gambar 4.7 alur dari pengiriman *push notification* pada sistem rekomendasi tentor bimbingan belajar ini dimulai dari admin yang mengirimkan notifikasi melalui web. Lalu notifikasi tersebut dikirimkan ke GCM, dan GCM akan mengirimkan pesan tersebut ke tentor rekomendasi pertama. Setelah tentor

pertama menerima notifikasi tersebut, tentor dapat menjawab penawaran tersebut dengan jawaban “ya” atau “tidak”. Ketika respon di cek dan tentor pertama menjawab “ya”, maka notifikasi tersebut akan dikirimkan ke perangkat admin melalui GCM juga. Namun, saat tentor menjawab “tidak”, maka akan di cek apakah tentor tersebut merupakan tentor terakhir dari rekomendasi, jika “ya” maka tentor akan mengirimkan notifikasi ke perangkat admin bahwa tidak ada tentor yang menerima siswa les dengan detail tersebut. Sedangkan jika tentor tadi bukan merupakan tentor terakhir, maka notifikasi dikirimkan ke GCM dan GCM mengirimkan notifikasi tersebut dikirimkan ke tentor rekomendasi berikutnya.

4.2.3 Diagram Sequence

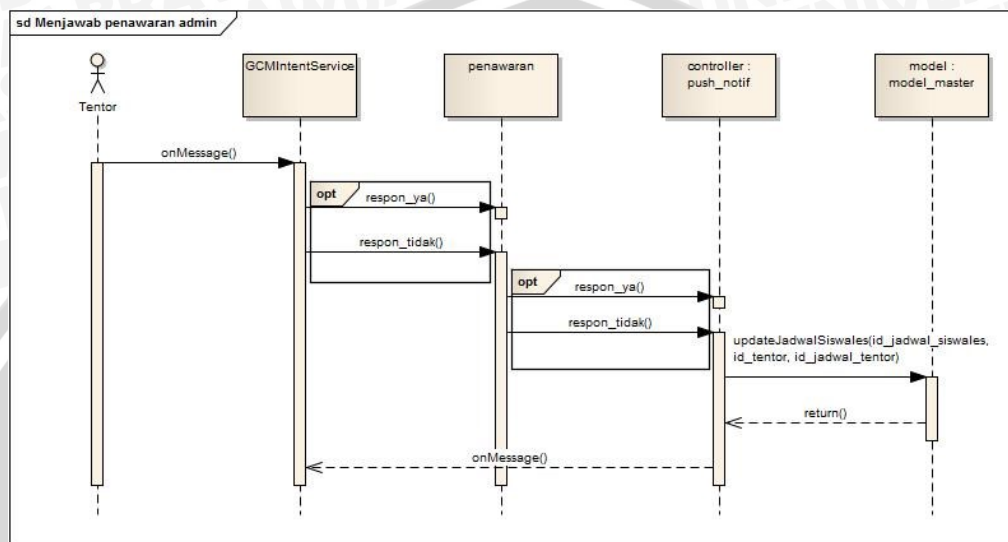
Diagram *Sequence* digunakan untuk menggambarkan interaksi antar user dan sistem yang disusun dalam urutan waktu. Diagram *Sequence* berupa deskripsi atau rangkaian langkah-langkah yang dilakukan dari sebuah kejadian untuk menghasilkan keluaran tertentu. Berikut merupakan diagram *Sequence* dari sistem rekomendasi tentor bimbingan belajar.



Gambar 4. 8 Diagram Sequence Melihat Ranking Tendor

Berdasarkan Diagram Activity pada Gambar 4.3 maka dapat dirancang Diagram Sequence untuk melihat ranking tentor yang direkomendasikan. Pada Gambar 4.8 menjelaskan interaksi menampilkan ranking tentor yang direkomendasikan sistem melalui metode *Profile Matching*, dimulai dengan membuka sistem kemudian admin akan melihat jadwal siswa les melalui view. Setelah tampilan selesai dimuat, admin akan menekan tombol rekomendasi untuk siswa les yang ingin dicarikan tentor untuk mendapatkan daftar tentor yang

direkomendasikan beserta rankingnya. Controller akan berinteraksi dengan model untuk mengambil data tentor dan data siswa les yang akan dilakukan perhitungan menggunakan metode *Profile Matching*. Lalu hasil perhitungan tadi akan ditampilkan oleh sistem melalui view. Diagram *sequence* melihat ranking tentor yang direkomendasikan dapat dilihat pada Gambar 4.9.

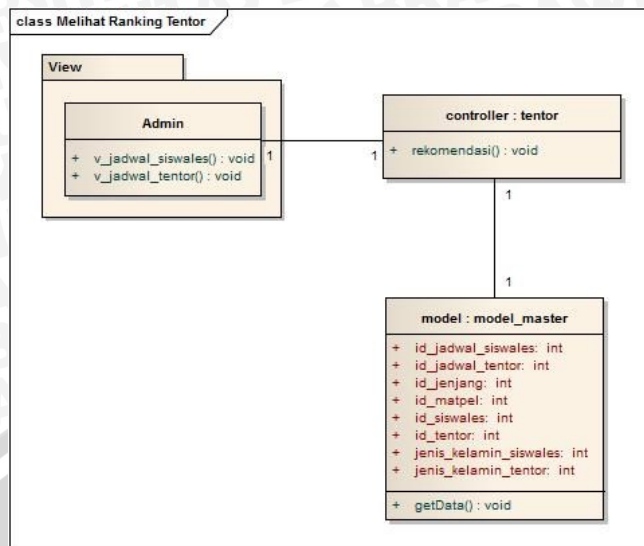


Gambar 4. 9 Diagram Sequence Menjawab Penawaran Admin

Berdasarkan Diagram *Activity* pada Gambar 4.4 maka dapat dibuat suatu Diagram Sequence untuk menjawab penawaran admin. Gambar 4.9 menjelaskan mengenai alur dari proses respon tentor terhadap penawaran admin. Pertama-tama, ketika tentor membuka *notification*, tentor akan melihat detail siswa les melalui view. Setelah itu saat tentor menekan tombol untuk merespon, controller akan melihat apakah jawaban yang diberikan “ya” atau “tidak”. Jika “ya”, maka controller akan bekerja sama dengan model untuk menyimpan respon tersebut berupa nama tentor di jadwal siswa les atau nama siswa les di jadwal tentor. Lalu setelah itu controller bekerja sama dengan view untuk menampilkan jadwal tentor. Jika respon dari tentor “tidak”, maka respon akan diteruskan ke tentor rekomendasi berikutnya.

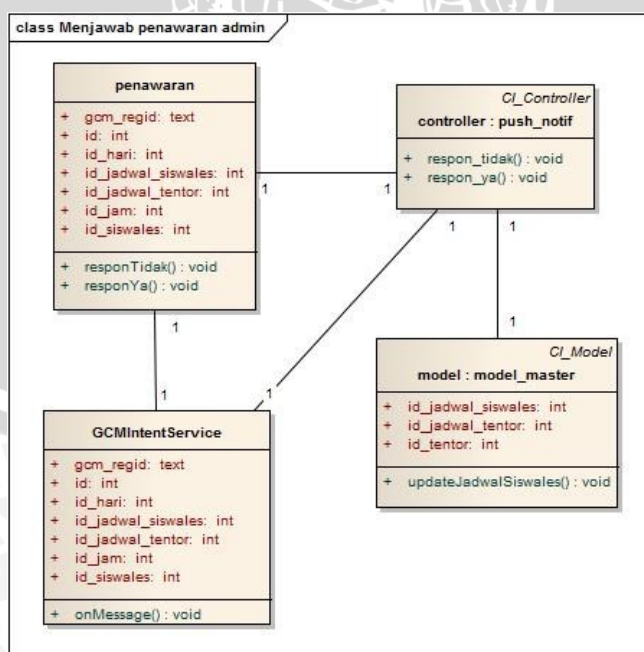
4.2.4 Diagram Class

Diagram *Class* merupakan sebuah gambaran tentang sistem dan relasi-relasi yang terdapat di dalamnya. *Class* diagram dari sistem rekomendasi tentor bimbingan belajar dapat dipresentasikan sebagai berikut :



Gambar 4. 10 Diagram Class Melihat Ranking Tentor

Pada Gambar 4.10 diatas menjelaskan diagram *class* dari melihat rekomendasi tentor. Tampilan rekomendasi tentor ini terdapat beberapa view diantaranya jadwal_siswales untuk menampilkan jadwal siswa les yang diminta, dan view rekomendasi untuk melihat hasil rekomendasi yang sudah dihitung menggunakan metode *Profile Matching*. Pada model terdapat method *getData* yang dipergunakan untuk mengambil data-data kriteria dari siswa les maupun tentor. Pada controller terdapat method *rekomendasi* untuk mendapatkan rekomendasi tentor dengan menggunakan metode *Profile Matching*.



Gambar 4. 11 Diagram Class Menjawab Penawaran Admin

Pada Gambar 4.11 diatas menjelaskan diagram *class* dari menjawab penawaran admin. Setelah ranking tentor sudah ditampilkan, maka ada tombol *notification* dimana admin dapat mengirimkan *notification* mengenai siswa les yang bersangkutan. Ketika tentor menerima *notification* mengenai penawaran siswa les tersebut, tentor dihadapkan dengan detail siswa les yang ada melalui *v_siswales* dan juga terdapat dua tombol untuk merespon. Method *responYa* dipanggil jika jawaban yang diberikan tentor adalah “ya”, sedangkan method *responTidak* akan dipanggil jika respon yang diberikan tentor adalah “Tidak”.

4.2.5 Persistensi *Class*

Perancangan basis data pada aplikasi digunakan untuk menyimpan data dan jadwal dari tentor dan siswa les. Berdasarkan class diagram melihat rekomendasi tentor pada Gambar 4.10 dapat diketahui atribut-atribut yang diperlukan dalam menghitung maupun menampilkan rekomendasi tentor. Sehingga, Gambar 4.8 ini dapat dipersestensikan menjadi beberapa tabel di dalam database, diantaranya tabel untuk menyimpan bobot profil dan data siswa les maupun tentor. Hal ini dapat dilihat pada Tabel 4.7, Tabel 4.8, Tabel 4.9, Tabel 4.10, Tabel 4.11, Tabel 4.12, Tabel 4.13, Tabel 4.14, Tabel 4.15, Tabel 4.16, Tabel 4.17, Tabel 4.18, Tabel 4.19, Tabel 4.20, Tabel 4.21, dan Tabel 4.22. Sedangkan, class diagram menjawab penawaran admin pada Gambar 4.11 dapat dipersestensikan menjadi dua tabel yang menyimpan respon berupa penyimpanan id siswa les dan tentor yang dapat dilihat pada Tabel 4.19 dan Tabel 4.20.

Berikut ini merupakan detail tabel dari database sistem rekomendasi tentor bimbingan belajar.

Tabel 4. 6 Tabel Admin

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_admin</u>	Tinyint(4)	Untuk menyimpan id_admin	1, 2
2	Username	Varchar(10)	Menyimpan username admin untuk login	Admin, administrator
3	Password	Varchar(100)	Untuk menyimpan password admin berupa enkripsi md5	21232f297a57a5a743894a0e4a801fc3

Tabel 4.6 merupakan tabel yang menyimpan akun admin. Di dalam tabel Admin ini terdapat kolom `id_admin` yang bertipe `tinyint` dan `username` serta `password` yang tipenya `varchar`. `Password` untuk akun admin ini disimpan dalam bentuk enkripsi `md5`.

Tabel 4. 7 Tabel Hari

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_hari</u>	Int(11)	Untuk menyimpan id hari sebagai primary key untuk hari	3, 5
2	Keterangan_hari	Varchar(10)	Untuk menyimpan penjelasan dari id_hari	Rabu, Jumat

Tabel 4.7 merupakan Tabel Hari, dimana di dalamnya terdapat kolom `id_hari` yang bertipe integer dan kolom `keterangan_hari` yang bertipe `varchar` dengan panjang karakter 10. Tabel Hari ini digunakan untuk menyimpan nama-nama hari beserta id nya. `Id_hari` merupakan primary key dari tabel ini, dimana `id_hari` ini nantinya dapat dipanggil di tabel lain dalam bentuk foreign key.

Tabel 4. 8 Tabel Jam

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_jam</u>	Int(11)	Untuk menyimpan id jam sebagai primary key untuk jam	2, 4
2	Keterangan_jam	Varchar(20)	Untuk menyimpan penjelasan dari id_jam	11.00-12.30, 15.00-16.30

Tabel 4.8 merupakan Tabel Jam, dimana di dalamnya terdapat kolom `id_jam` yang bertipe integer dan kolom `keterangan_jam` yang bertipe `varchar` dengan panjang karakter 20. Tabel Jam ini digunakan untuk menyimpan jam-jam yang sudah ditentukan untuk les beserta id jamnya. `Id_jam` merupakan primary key dari tabel ini, dimana `id_jam` ini nantinya dapat dipanggil di tabel lain dalam bentuk foreign key.

Tabel 4. 9 Tabel bp_jekel

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bp_jekel</u>	Int(11)	Untuk menyimpan id_bp_jekel yang digunakan sebagai primary key untuk jenis kelamin	1, 2
2	Bobot_profil_jekel	Int(11)	Untuk menyimpan bobot dari id_bp_jekel tertentu	1, 2
3	Keterangan_jekel	Varchar(8)	Untuk menyimpan penjelasan dari id_bp_jekel tertentu	Pria, Wanita

Tabel 4.9 merupakan Tabel bp_jekel, dimana di dalamnya terdapat kolom id_bp_jekel, kolom bobot_profil_jekel yang bertipe integer dan kolom keterangan_jekel yang bertipe varchar dengan panjang karakter 8. Tabel bp_jekel ini digunakan untuk menyimpan bobot profil dari jenis kelamin tentor maupun siswa les. Id_bp_jekel merupakan primary key dari tabel ini.

Tabel 4. 10 Tabel bp_jenang

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bp_jenang</u>	Int(11)	Untuk menyimpan id_bp_jenang yang digunakan sebagai primary key untuk jenjang pendidikan	1, 2, 3
2	Bobot_profil_jenang	Int(11)	Untuk menyimpan bobot dari id_bp_jenang tertentu	1, 2, 3
3	Keterangan_jenang	Varchar(5)	Untuk menyimpan penjelasan dari id_bp_jenang tertentu	SD, SMP, SMA

Tabel 4.10 merupakan Tabel bp_jenang, dimana di dalamnya terdapat kolom id_bp_jenang, kolom bobot_profil_jenang yang bertipe integer dan kolom keterangan_jenang yang bertipe varchar dengan panjang karakter 5. Tabel bp_jenang ini digunakan untuk menyimpan bobot profil dari jenjang pendidikan

yang dikuasai tentor maupun jenjang pendidikan siswa les yang terdaftar. Id_bp_jenjang merupakan primary key dari tabel ini.

Tabel 4. 11 Tabel bp_matpel

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bp_matpel</u>	Int(11)	Untuk menyimpan id_bp_matpel yang digunakan sebagai primary key untuk mata pelajaran	8, 11
2	Bobot_profil_matpel	Int(11)	Untuk menyimpan bobot dari id_bp_jenjang tertentu	8, 11
3	Keterangan_matpel	Varchar(20)	Untuk menyimpan penjelasan dari id_bp_jenjang tertentu	Fisika, Sosiologi

Tabel 4.11 merupakan Tabel bp_matpel, dimana di dalamnya terdapat kolom id_bp_matpel, kolom bobot_profil_matpel yang bertipe integer dan kolom keterangan_matpel yang bertipe varchar dengan panjang karakter 20. Tabel bp_matpel ini digunakan untuk menyimpan bobot profil dari mata pelajaran yang dikuasai oleh tentor maupun mata pelajaran yang diinginkan siswa les untuk diberikan les. Id_bp_matpel merupakan primary key dari tabel ini.

Tabel 4. 12 Tabel bp_waktu

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bp_waktu</u>	Int(11)	Untuk menyimpan id_bp_waktu yang digunakan sebagai primary key untuk waktu hari dan jam	4, 5
2	Bobot_profil_waktu	Int(11)	Untuk menyimpan bobot dari id_bp_jekel tertentu	4, 5
3	Keterangan_waktu	Varchar(150)	Untuk menyimpan penjelasan dari id_bp_jekel tertentu	Jadwal siswa les jam/hari pertama

Tabel 4.12 merupakan Tabel bp_waktu, dimana di dalamnya terdapat kolom id_bp_waktu, kolom bobot_profil_waktu yang bertipe integer dan kolom keterangan_waktu yang bertipe varchar dengan panjang karakter 150. Tabel bp_waktu ini digunakan untuk menyimpan bobot profil dari jadwal yang diinginkan siswa les untuk diberikan les dan juga jadwal kosong tentor. Waktu ini terdiri dari 2 kriteria, yaitu hari dan jam. Id_bp_waktu merupakan primary key dari tabel ini.

Tabel 4. 13 Tabel bn_jekel

No	Kolom	Type	Keterangan	Contoh
1	Id_bn_jekel	Int(11)	Untuk menyimpan id_bn_jekel yang digunakan sebagai primary key untuk bobot nilai jenis kelamin	1, 2
2	Selisih_jekel	Int(11)	Untuk menyimpan kemungkinan selisih bobot profil jenis kelamin	0, 1, -1
3	Bobot_nilai_jekel	Float	Untuk menyimpan bobot nilai berdasarkan selisih_jekel	1.5, 2
4	Keterangan_bn_jekel	Varchar(100)	Untuk menyimpan penjelasan mengenai selisih_jekel	Ada kelebihan 1 bobot profil jenis kelamin siswa dengan tentor

Tabel 4.13 merupakan Tabel bn_jekel, dimana di dalamnya terdapat kolom id_bn_jekel, selisih_jekel yang bertipe integer, kolom bobot_nilai_jekel yang bertipe float dan kolom keterangan_bn_jekel yang bertipe varchar dengan panjang karakter 100. Tabel bn_jekel ini digunakan untuk menyimpan bobot nilai yang ditentukan dengan selisih gap antara jenis kelamin tentor dengan jenis kelamin siswa les. Id_bn_jekel merupakan primary key dari tabel ini.

Tabel 4. 14 Tabel bn_jenang

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bn_jenang</u>	Int(11)	Untuk menyimpan id_bn_jenang yang digunakan sebagai primary key untuk bobot nilai jenjang	1, 2, 3
2	Selisih_jenang	Int(11)	Untuk menyimpan kemungkinan selisih bobot profil jenjang	0, -2
3	Bobot_nilai_jenang	Float	Untuk menyimpan bobot nilai berdasarkan selisih_jenang	1, 2.5
4	Keterangan_bn_jenang	Varchar(100)	Untuk menyimpan penjelasan mengenai selisih_jenang	Ada kekurangan 2 bobot profil jenjang siswa les dengan jenjang yang dikuasai tentor

Tabel 4.14 merupakan Tabel bn_jenang, dimana di dalamnya terdapat kolom id_bn_jenang, selisih_jenang yang bertipe integer, kolom bobot_nilai_jenang yang bertipe float dan kolom keterangan_bn_jenang yang bertipe varchar dengan panjang karakter 100. Tabel bn_jenang ini digunakan untuk menyimpan bobot nilai yang ditentukan dengan selisih gap antara jenjang tentor dengan jenjang siswa les. Id_bn_jenang merupakan primary key dari tabel ini.

Tabel 4.15 merupakan Tabel bn_matpel, dimana di dalamnya terdapat kolom id_bn_matpel, selisih_matpel yang bertipe integer, kolom bobot_nilai_matpel yang bertipe float dan kolom keterangan_bn_matpel yang bertipe varchar dengan panjang karakter 150. Tabel bn_matpel ini digunakan untuk menyimpan bobot nilai yang ditentukan dengan selisih gap antara mata

pelajaran yang dikuasai tentor dengan mata pelajaran yang diinginkan siswa les. Id_bn_matpel merupakan primary key dari tabel ini.

Tabel 4. 15 Tabel bn_matpel

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bn_matpel</u>	Int(11)	Untuk menyimpan id_bn_matpel yang digunakan sebagai primary key untuk bobot nilai mata pelajaran	1, 3, 5
2	Selisih_matpel	Int(11)	Untuk menyimpan kemungkinan selisih bobot profil mata pelajaran	-5, 8, 10
3	Bobot_nilai_matpel	Float	Untuk menyimpan bobot nilai berdasarkan selisih_matpel	4, 6, 7.5
4	Keterangan_bn_matpel	Varchar(150)	Untuk menyimpan penjelasan mengenai selisih_matpel	Ada kelebihan 8 bobot profil mata pelajaran antara tentor dengan siswa les

Tabel 4.16 merupakan Tabel bn_waktu, dimana di dalamnya terdapat kolom id_bn_waktu, selisih_waktu yang bertipe integer, kolom bobot_nilai_waktu yang bertipe float dan kolom keterangan_bn_waktu yang bertipe varchar dengan panjang karakter 100. Tabel bn_waktu ini digunakan untuk menyimpan bobot nilai yang ditentukan dengan selisih gap antara jadwal tentor yang terdiri dari hari dan jam kosong dengan jadwal les yang diinginkan siswa les. Id_bn_waktu merupakan primary key dari tabel ini.

Tabel 4. 16 Tabel bn_waktu

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_bn_waktu</u>	Int(11)	Untuk menyimpan id_bn_waktu yang digunakan sebagai primary key untuk bobot nilai waktu	1, 3, 4
2	Selisih_waktu	Int(11)	Untuk menyimpan kemungkinan selisih bobot profil waktu	3, -4
3	Bobot_nilai_waktu	Float	Untuk menyimpan bobot nilai berdasarkan selisih_waktu	3.5, 5
4	Keterangan_bn_waktu	Varchar(100)	Untuk menyimpan penjelasan mengenai selisih_waktu	Waktu kosong tentor dan jam siswa les kekurangan 1 jam / 1 hari

Tabel 4.17 merupakan Tabel Tendor, dimana di dalamnya terdapat kolom id_tendor yang bertipe integer, kolom gcm_regid untuk menyimpan gcm registrasi id yang diberikan oleh server GCM ketika registrasi, kolom nama_tendor, jenis_kelamin_tendor, email_tendor, telepon_tendor, password_tendor yang bertipe varchar serta created_at yang bertipe timestamp. Tabel tendor ini digunakan untuk menyimpan data primer dari tendor dalam bimbingan belajar. Id_tendor merupakan primary key dari tabel ini.

Tabel 4. 17 Tabel tendor

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_tendor</u>	Int(11)	Untuk menyimpan id tendor yang merupakan primary key dari data tendor	2
2	Gcm_regid	Text	Untuk	APA91bHZRDqpBqIE

			menyimpan GCM registrasi ID yang diberikan oleh GCM	WYgTGvBwDCS1cJexRfVPomD1pDxkuYFUhfK4- _CJeK9_4OB0YCEHU LEV_- JZykc7mf7WR6OFGy7 91T5U1aZfXp7UXCDs O1HWD1mtaOaONPH nwoF55r7a1Ea- IzmmJJCMVSx_Mi- AR8TkiUkIQ
3	Nama_tentor	Varchar(50)	Untuk menyimpan nama dari tentor	Luluk Setiawati Hartono
4	Jenis_kelamin_tentor	Varchar(10)	Untuk menyimpan jenis kelamin tentor	Wanita
5	Email_tentor	Varchar(25)	Untuk menyimpan email tentor yang digunakan untuk login	luluksetiawatihartono@gmail.com
6	Telepon_tentor	Varchar(20)	Untuk menyimpan telepon tentor	085649885675
7	Password_tentor	Varchar(50)	Untuk menyimpan password tentor yang digunakan untuk login	Luluk
8	Created_at	Timestamp	Untuk menyimpan waktu tentor itu terdaftar pada aplikasi	2015-04-21 13:29:33

Tabel 4.18 merupakan Tabel Siswales, dimana di dalamnya terdapat kolom id_siswales yang bertipe integer, kolom nama_siswales, alamat_siswales, telepon_siswales, dan sekolah_siswales yang bertipe varchar. Lalu terdapat kolom jenjang yang bertipe integer. Tabel siswales ini digunakan untuk menyimpan data

primer dari siswa les yang terdaftar di bimbingan belajar. Id_siswales merupakan primary key dari tabel ini.

Tabel 4. 18 Tabel siswales

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_siswales</u>	Int(11)	Untuk menyimpan id siswa les yang digunakan sebagai primary key data siswa les	2, 5
2	Nama_siswales	Varchar(50)	Untuk menyimpan nama dari siswa les	Karina
3	Jenis_kelamin_siswales	int(11)	Untuk menyimpan jenis kelamin siswa les	Wanita
4	Alamat_siswales	Varchar(100)	Untuk meyimpan alamat dari siswa les	Jalan Kebalen 4
5	Telepon_siswales	Varchar(15)	Untuk menyimpan telepon siswa les	928831
6	Jenjang_siswales	Int(11)	Untuk menyimpan jenjang pendidikan siswa les	1
7	Sekolah_siswales	Varchar(50)	Untuk menyimpan sekolah siswa les	SMAN 1

Tabel 4.19 merupakan Tabel jadwal_siswales, dimana di dalamnya terdapat kolom id_jadwal_siswales, id_siswales, id_hari, id_jam, id_matpel dan id_tentor yang bertipe integer. Tabel jadwal_siswales ini digunakan untuk menyimpan data jadwal les dari siswa les yang terdaftar di bimbingan belajar.

Id_jadwal_siswales merupakan primary key dari tabel ini. Sedangkan kolom selain id_jadwal_siswales merupakan foreign key dari tabel lainnya.

Tabel 4. 19 Tabel jadwal_siswales

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_jadwal_siswales</u>	Int(11)	Untuk menyimpan id dari jadwal siswa les yang digunakan sebagai primary key jadwal siswa les	1, 4
2	<u>Id_siswales</u>	Int(11)	Untuk menyimpan id dari siswa les yang mempunyai jadwal tersebut	1, 5
3	<u>Id_hari</u>	Int(11)	Untuk menyimpan id hari dari jadwal siswa les	2, 3
4	<u>Id_jam</u>	Int(11)	Untuk menyimpan id jam dari jadwal siswa les	3, 4
5	<u>Id_matpel</u>	Int(11)	Untuk menyimpan id matpel dari mata pelajaran yang diinginkan siswa les pada jadwal tersebut	5, 11
6	<u>Id_tentor</u>	Int(11)	Untuk menyimpan id tentor yang mengisi jadwal siswa les tersebut	3, 5

Tabel 4.20 merupakan Tabel jadwal_tentor, dimana di dalamnya terdapat kolom id_jadwal_tentor, id_tentor, id_hari, id_jam dan id_jadwal_siswales yang bertipe integer. Tabel jadwal_tentor ini digunakan untuk menyimpan data jadwal kosong dari tentor yang ada di bimbingan belajar. Id_jadwal_tentor merupakan primary key dari tabel ini. Sedangkan kolom selain id_jadwal_tentor merupakan foreign key dari tabel lainnya.

Tabel 4. 20 Tabel jadwal_tentor

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_jadwal_tentor</u>	Int(11)	Untuk menyimpan id dari jadwal tentor	2, 3

			yang digunakan sebagai primary key pada jadwal tertentu	
2	<u>Id_tentor</u>	Int(11)	Untuk menyimpan id dari tentor yang memiliki jadwal tersebut	2, 4
3	<u>Id_hari</u>	Int(11)	Untuk menyimpan id hari dari jadwal kosong yang dimiliki tentor	5, 6
4	<u>Id_jam</u>	Int(11)	Untuk menyimpan id jam dari jadwal kosong yang dimiliki tentor	4, 5
5	<u>Id_jadwal_siswales</u>	Int(11)	Untuk menyimpan id dari jadwal siswa les yang diajar pada jadwal yang dimiliki tentor tersebut	2, 5

Tabel 4.21 merupakan Tabel jenjang_tentor, dimana di dalamnya terdapat kolom id_jenjang_tentor, id_tentor, id_jenjang yang bertipe integer. Tabel jenjang_tentor ini digunakan untuk menyimpan data jenjang yang dikuasai oleh tentor karena satu tentor dapat menguasai lebih dari satu jenjang pendidikan. Id_jenjang_tentor merupakan primary key dari tabel ini. Sedangkan kolom id_tentor dan id_jenjang merupakan foreign key dari tabel tentor dan bp_jenjang.

Tabel 4. 21 Tabel jenjang_tentor

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_jenjang_tentor</u>	Int(11)	Sebagai primary key yang menandakan seorang tentor menguasai jenjang tertentu	1, 2
2	<u>Id_tentor</u>	Int(11)	Untuk menyimpan id tentor	2, 4

3	<u>Id_jenang</u>	Int(11)	Untuk menyimpan id jenjang yang dikuasai tentor	1, 2
---	------------------	---------	---	------

Tabel 4.22 merupakan Tabel `matpel_tentor`, dimana di dalamnya terdapat kolom `id_matpel_tentor`, `id_tentor`, `id_matpel` yang bertipe integer. Tabel `matpel_tentor` ini digunakan untuk menyimpan data mata pelajaran yang dikuasai oleh tentor karena satu tentor dapat menguasai lebih dari satu mata pelajaran. `Id_matpel_tentor` merupakan primary key dari tabel ini. Sedangkan kolom `id_tentor` dan `id_matpel` merupakan foreign key dari tabel `tentor` dan `bp_matpel`.

Tabel 4. 22 Tabel `matpel_tentor`

No	Kolom	Type	Keterangan	Contoh
1	<u>Id_matpel_tentor</u>	Int(11)	Sebagai primary key yang digunakan untuk menandakan bahwa seorang tentor menguasai mata pelajaran tertentu	1, 6
2	<u>Id_tentor</u>	Int(11)	Untuk menyimpan id tentor	2, 4
3	<u>Id_matpel</u>	Int(11)	Untuk menyimpan id matpel yang dikuasai oleh tentor	7, 10

4.2.6 Struktur Data JSON

```
{
  "tentor": [
    {
      "id_tentor": "integer",
      "gcm_regid": "string",
      "nama_tentor": "string",
      "jenis_kelamin_tentor": "integer",
      "email_tentor": "string",
      "telepon_tentor": "string",
      "password_tentor": "string",
      "created_at": "timestamp"
    }
  ]
}
```

Gambar 4. 12 Struktur Data JSON data tentor

```
{
  "tentor": [
    {
      "id_tentor": "6",
      "gcm_regid": "APA91bFhIT4R3HlwPuKS08FXpOe9cSAfB31U4fThGJve
ZO-iH-fXBnh6jZX8-
g4Hp4s7iapcYFnSDQORIKGA7Fks40a3BNeqyqXRRfJLskyWD4v-MKd-
M15JgwXqi422uviOgNjGmzANdX5WPY_yG-y_OSIZ9SFsIQ",
      "nama_tentor": "luluk",
      "jenis_kelamin_tentor": "1",
      "email_tentor": "luluk",
      "telepon_tentor": "826496",
      "password_tentor": "luluk",
      "created_at": "2015-04-21 13:29:33"
    }
  ]
}
```

Gambar 4. 13 Contoh JSON data tentor

Pada Gambar 4.12 dijelaskan mengenai struktur data JSON tentor yang berisi id untuk kode unik masing-masing tentor, gcm_regid sebagai tanda pengenal smartphone yang digunakan oleh tentor yang nantinya akan digunakan admin untuk dapat mengirim *push notification*. Nama, alamat, dan telepon yang digunakan sebagai tanda pengenal seorang tentor dan juga email serta password yang digunakan untuk login dalam aplikasi. Created_at menyatakan mulai kapan tentor tersebut bergabung menjadi tentor di bimbingan belajar. Gambar 4.13 merupakan contoh JSON dari format JSON daftar tentor.

```
{
  "jadwal_siswales": [
    {
      "id_jadwal_siswales": "integer",
      "nama_siswales": "string",
      "keterangan_hari": "string",
      "keterangan_jam": "string",
      "keterangan_matpel": "string",
      "nama_tentor": "string"
    }
  ]
}
```

Gambar 4. 14 Struktur Data JSON jadwal siswa les

Struktur JSON dari jadwal siswa les dapat dilihat pada Gambar 4.14, dimana di dalamnya terdapat id_jadwal karena masing-masing siswa les dapat mempunyai lebih dari satu jadwal dengan mata pelajaran, hari, jam dan tentor yang berbeda-beda. Keterangan_hari dan keterangan_jam untuk menyatakan hari dan jam siswa les serta mata pelajaran dan tentor yang mengajar pada jadwal

tersebut. Gambar 4.15 merupakan contoh dari format JSON untuk jadwal siswa les.

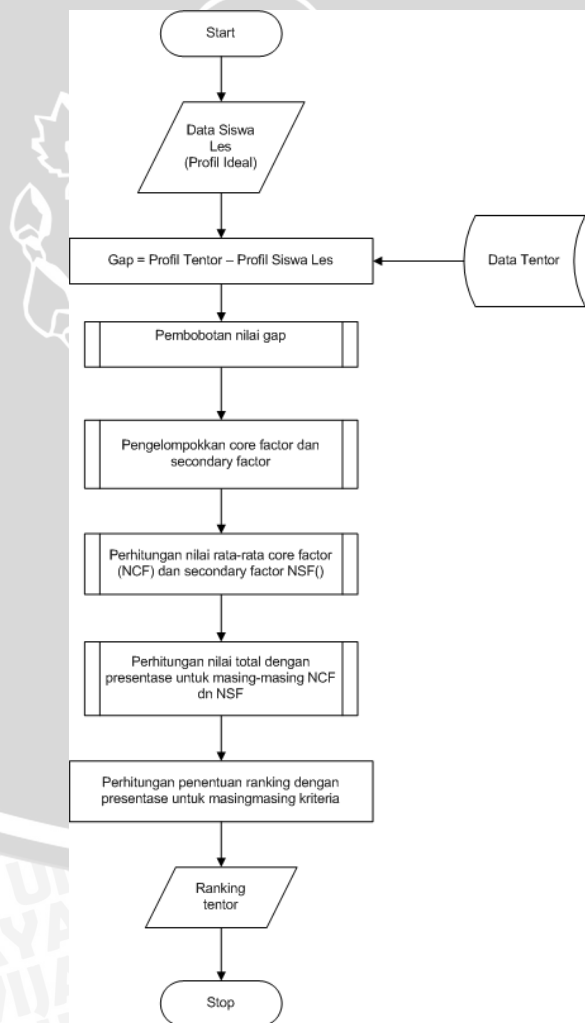
```

{
  "jadwal_siswales": [
    {
      "id_jadwal_siswales": "7",
      "nama_siswales": "Karina",
      "keterangan_hari": "Jumat",
      "keterangan_jam": "09.00-10.30",
      "keterangan_matpel": "Komputer",
      "nama_tentor": "luluk"
    }
  ]
}

```

Gambar 4. 15 Contoh JSON jadwal siswa les

4.2.7 Perancangan Algoritma *Profile Matching*



Gambar 4. 16 Diagram Alir *Profile Matching*



a. Aturan / Rule Penilaian

Sistem rekomendasi tutor bimbingan belajar ini menggunakan 4 kriteria, yaitu waktu, mata pelajaran, jenis kelamin dan jenjang. Untuk mendukung pengolahan data menggunakan algoritma *Profile Matching*, maka terdapat aturan/rule. Aturan tersebut dapat dijelaskan sebagai berikut:

1. Rule Penilaian Waktu

Pada penilaian waktu terdapat 2 subkriteria penilaian, yaitu hari dan jam. Berikut ini adalah tabel untuk bobot profil dan bobot gap untuk kriteria penilaian waktu.

Tabel 4. 23 Tabel Rancangan Aturan Profil Penilaian Waktu

Bobot Profil	Keterangan
1	Jadwal siswa les pada jam pertama atau hari pertama
2	Jadwal siswa les pada jam kedua atau hari kedua
3	Jadwal siswa les pada jam ketiga atau hari ketiga
4	Jadwal siswa les pada jam keempat atau hari keempat
5	Jadwal siswa les pada jam kelima atau hari kelima
6	Jadwal siswa les pada jam keenam atau hari keenam

Tabel 4.23 menunjukkan bahwa penentuan bobot profil untuk penilaian waktu. Profil ideal yang diinginkan adalah sesuai dengan waktu yang diinginkan oleh siswa les yang bersangkutan. Untuk mendapatkan nilai gap, dilakukan dengan menghitung selisih antara bobot profil tutor dengan profil siswa les dan selanjutnya akan didapatkan bobot gap. Aturan bobot gap dapat dilihat pada tabel 4.24.

Tabel 4.24 menunjukkan bahwa pada aturan bobot nilai gap kriteria waktu memiliki level tertinggi = 6 untuk bobotnya, dalam hal ini selisih = 0. Selanjutnya level terendah adalah 1 yang ditunjukkan dengan selisih antara profil tutor dengan profil siswa les = -5. Untuk *core factor* adalah hari dengan bobotnya sebesar 60%, sedangkan *secondary factor* adalah jam dengan bobot sebesar 40%.

Tabel 4. 24 Tabel Rancangan Aturan Gap Kriteria Waktu

Selisih	Bobot Nilai	Keterangan
0	6	Tidak ada selisih antara waktu kosong tutor dan jam siswa les
1	5.5	Waktu kosong tutor dan jam siswa les kelebihan 1

		jam / 1 hari
-1	5	Waktu kosong tentor dan jam siswa les kekurangan 1 jam / 1 hari
2	4.5	Waktu kosong tentor dan jam siswa les kelebihan 2 jam / 2 hari
-2	4	Waktu kosong tentor dan jam siswa les kekurangan 2 jam / 2 hari
3	3.5	Waktu kosong tentor dan jam siswa les kelebihan 3 jam / 3 hari
-3	3	Waktu kosong tentor dan jam siswa les kekurangan 3 jam / 3 hari
4	2.5	Waktu kosong tentor dan jam siswa les kelebihan 4 jam / 4 hari
-4	2	Waktu kosong tentor dan jam siswa les kekurangan 4 jam / 4 hari
5	1.5	Waktu kosong tentor dan jam siswa les kelebihan 5 jam / 5 hari
-5	1	Waktu kosong tentor dan jam siswa les kekurangan 5 jam / 5 hari

Core Factor (60%) : Hari

Secondary Factor (40%) : Jam

2. Rule Penilaian Mata Pelajaran

Pada kriteria mata pelajaran tidak memiliki subkriteria penilaian seperti kriteria waktu. Berikut ini adalah tabel untuk bobot profil dan bobot gap untuk kriteria penilaian mata pelajaran.

Tabel 4. 25 Tabel Rancangan Aturan Profil Penilaian Mata Pelajaran

Bobot Profil	Mata Pelajaran	Keterangan
1	Bahasa Mandarin	Pelajaran yang diminta siswa les adalah bahasa mandarin
2	Bahasa Inggris	Pelajaran yang diminta siswa les adalah bahasa inggris
3	Komputer	Pelajaran yang diminta siswa les adalah komputer
4	Matematika	Pelajaran yang diminta siswa les adalah matematika
5	IPA	Pelajaran yang diminta siswa les adalah IPA
6	Biologi	Pelajaran yang diminta siswa les adalah biologi
7	Fisika	Pelajaran yang diminta siswa les adalah fisika
8	Kimia	Pelajaran yang diminta siswa les adalah kimia
9	IPS	Pelajaran yang diminta siswa les adalah IPS

10	Ekonomi	Pelajaran yang diminta siswa les adalah ekonomi
11	Akuntansi	Pelajaran yang diminta siswa les adalah akuntansi
12	Sosiologi	Pelajaran yang diminta siswa les adalah sosiologi
13	Geografi	Pelajaran yang diminta siswa les adalah geografi

Tabel 4.25 menunjukkan bahwa penentuan bobot profil kriteria mata pelajaran berdasarkan mata pelajaran yang diminta siswa les dan mata pelajaran yang dikuasai tentor. Aturan untuk bobot nilai gap kriteria mata pelajaran ditunjukkan pada Tabel 4.26.

Tabel 4. 26 Tabel Rancangan Aturan Gap Kriteria Mata Pelajaran

Selisih	Bobot Nilai	Keterangan
0	7	Tidak ada selisih mata pelajaran antara tentor dengan siswa les
1	6.75	Ada kelebihan 1 bobot profil mata pelajaran antara tentor dengan siswa les
-1	6.5	Ada kekurangan 1 bobot profil mata pelajaran antara tentor dengan siswa les
2	6.25	Ada kelebihan 2 bobot profil mata pelajaran antara tentor dengan siswa les
-2	6	Ada kekurangan 2 bobot profil mata pelajaran antara tentor dengan siswa les
3	5.75	Ada kelebihan 3 bobot profil mata pelajaran antara tentor dengan siswa les
-3	5.5	Ada kekurangan 3 bobot profil mata pelajaran antara tentor dengan siswa les
4	5.25	Ada kelebihan 4 bobot profil mata pelajaran antara tentor dengan siswa les
-4	5	Ada kekurangan 4 bobot profil mata pelajaran antara tentor dengan siswa les
5	4.75	Ada kelebihan 5 bobot profil mata pelajaran antara tentor dengan siswa les
-5	4.5	Ada kekurangan 5 bobot profil mata pelajaran antara tentor dengan siswa les
6	4.25	Ada kelebihan 6 bobot profil mata pelajaran antara tentor dengan siswa les
-6	4	Ada kekurangan 6 bobot profil mata pelajaran antara tentor dengan siswa les
7	3.75	Ada kelebihan 7 bobot profil mata pelajaran antara tentor dengan siswa les
-7	3.5	Ada kekurangan 7 bobot profil mata pelajaran antara tentor dengan siswa les
8	3.25	Ada kelebihan 8 bobot profil mata pelajaran antara tentor dengan siswa les
-8	3	Ada kekurangan 8 bobot profil mata pelajaran

		antara tentor dengan siswa les
9	2.75	Ada kelebihan 9 bobot profil mata pelajaran antara tentor dengan siswa les
-9	2.5	Ada kekurangan 9 bobot profil mata pelajaran antara tentor dengan siswa les
10	2.25	Ada kelebihan 10 bobot profil mata pelajaran antara tentor dengan siswa les
-10	2	Ada kekurangan 10 bobot profil mata pelajaran antara tentor dengan siswa les
11	1.75	Ada kelebihan 11 bobot profil mata pelajaran antara tentor dengan siswa les
-11	1.5	Ada kekurangan 11 bobot profil mata pelajaran antara tentor dengan siswa les
12	1.25	Ada kelebihan 12 bobot profil mata pelajaran antara tentor dengan siswa les
-12	1	Ada kekurangan 12 bobot profil mata pelajaran antara tentor dengan siswa les

Tabel 4.26 menunjukkan bahwa pada aturan bobot nilai gap kriteria mata pelajaran memiliki level tertinggi = 7 untuk bobotnya, dalam hal ini selisih = 0. Selanjutnya level terendah adalah 1 yang ditunjukkan dengan selisih antara profil tentor dengan profil siswa les = -12.

3. Rule Penilaian Jenis Kelamin

Pada kriteria jenis kelamin tidak memiliki subkriteria penilaian seperti kriteria waktu. Berikut ini adalah tabel untuk bobot profil dan bobot gap untuk kriteria penilaian jenis kelamin.

Tabel 4. 27 Tabel Rancangan Aturan Profil Jenis Kelamin

Bobot Profil	Jenis Kelamin	Keterangan
1	Pria	Jenis kelamin tentor adalah pria
2	Wanita	Jenis kelamin tentor adalah wanita

Tabel 4.27 menunjukkan bahwa penentuan bobot profil kriteria jenis kelamin. Aturan untuk bobot nilai gap kriteria jenis kelamin ditunjukkan pada Tabel 4.28.

Tabel 4. 28 Tabel Rancangan Aturan Gap Kriteria Jenis Kelamin

Selisih	Bobot Nilai	Keterangan
0	2	Tidak ada selisih antara bobot profil jenis kelamin tentor dengan siswa les
1	1.5	Ada kelebihan 1 bobot profil jenis kelamin tentor dengan siswa les
-1	1	Ada kekurangan 1 bobot profil jenis kelamin tentor dengan siswa les

Tabel 4.28 menunjukkan bahwa pada aturan bobot nilai gap kriteria jenis kelamin memiliki level tertinggi = 2 untuk bobotnya, dalam hal ini selisih = 0 . Selanjutnya level terendah adalah 1 yang ditunjukkan dengan selisih antara profil tentor dengan profil siswa les = -1.

4. Rule Penilaian Jenjang

Pada kriteria jenjang tidak memiliki subkriteria penilaian seperti kriteria waktu. Berikut ini adalah tabel untuk bobot profil dan bobot gap untuk kriteria penilaian jenjang.

Tabel 4. 29 Tabel Rancangan Aturan Profil Penilaian Jenjang

Bobot Profil	Jenjang Sekolah	Keterangan
1	SD	Jenjang siswa les adalah SD
2	SMP	Jenjang siswa les adalah SMP
3	SMA	Jenjang siswa les adalah SMA

Tabel 4.29 menunjukkan bahwa penentuan bobot profil kriteria jenjang berdasarkan jenjang siswa les dan jenjang yang dikuasai tentor. Aturan untuk bobot nilai gap kriteria jenjang ditunjukkan pada Tabel 4.30.

Tabel 4. 30 Tabel Rancangan Aturan Gap Kriteria Jenjang

Selisih	Bobot Nilai	Keterangan
0	3	Tidak ada selisih bobot profil jenjang siswa les dengan jenjang yang dikuasai tentor
1	2.5	Ada kelebihan 1 bobot profil jenjang siswa les dengan jenjang yang dikuasai tentor
-1	2	Ada kekurangan 1 bobot profil jenjang siswa les dengan jenjang yang dikuasai tentor
2	1.5	Ada kelebihan 2 bobot profil jenjang siswa les dengan jenjang yang dikuasai tentor
-2	1	Ada kekurangan 2 bobot profil jenjang siswa les dengan jenjang yang dikuasai tentor

Tabel 4.30 menunjukkan bahwa pada aturan bobot nilai gap kriteria jenjang memiliki level tertinggi = 3 untuk bobotnya, dalam hal ini selisih = 0. Selanjutnya level terendah adalah 1 yang ditunjukkan dengan selisih antara profil tentor dengan profil siswa les = -2.

b. Proses Perhitungan Pemetaan Gap Kompetensi

Proses pemetaan gap yang terjadi sebenarnya, mempunyai rumus umum yang berlaku untuk menghitung bobot dari masing-masing kriteria, yaitu:

$$\text{Gap} = \text{Profil Tantor} - \text{Profil Siswa Les}$$

Gap adalah beda antara profil tentor dengan profil siswa les. Berikut ini merupakan hasil proses perhitungan pemetaan gap kompetensi yang diambil dari data sample untuk masing-masing kriteria.

1. Waktu

Contoh hasil proses perhitungan pemetaan gap waktu tertera pada Tabel 4.31. Data pada kolom hari merupakan hari kosong dari masing-masing tentor. Data pada kolom bobot profil merupakan normalisasi nilai berdasarkan pada norma bobot profil waktu. Profil ideal ditunjukkan dengan record berlatar belakang abu-abu, yaitu hari adalah 4 yang artinya adalah hari keempat yakni Kamis dan jam adalah 2 yakni jam 11.00 – 12.30. Selanjutnya dilakukan perhitungan nilai gap antara bobot profil masing-masing tentor dengan profil siswa les, sehingga didapatkan gap atau selisihnya.

Tabel 4. 31 Tabel Hasil Pemetaan Gap Kompetensi Waktu

No	Id tentor	Jam	Nilai hari	Bobot Profil 1	Bobot Profil 2
1	A	09.00-10.30	Selasa	2	1
2	B	15.00-16.30	Selasa	2	4
3	C	17.00-18.30	Kamis	4	5
	Profil Ideal	11.00-12.30	Kamis	4	2
Gap					
1	A			-2	-1
2	B			-2	2
3	C			0	3

2. Mata Pelajaran

Tabel 4. 32 Tabel Hasil Pemetaan Gap Kompetensi Mata Pelajaran

No	Id Tentor	Mata Pelajaran	Bobot Profil
1	A	Fisika	7
2	B	Matematika	4
3	C	Bahasa Inggris	2
	Profil Ideal	Kimia	8
			Gap
1	A		-1
2	B		-4
3	C		-6

Contoh hasil proses perhitungan pemetaan gap mata pelajaran tertera pada Tabel 4.32. Data pada kolom mata pelajaran merupakan mata pelajaran yang dikuasai masing-masing tentor. Data pada kolom bobot profil merupakan normalisasi nilai berdasarkan pada norma bobot profil mata pelajaran. Profil ideal ditunjukkan dengan record berlatar belakang abu-abu, yaitu mata pelajaran adalah 8 yang artinya adalah mata pelajaran kimia. Selanjutnya dilakukan perhitungan nilai gap antara bobot profil masing-masing tentor dengan profil siswa les, sehingga didapatkan gap atau selisihnya.

3. Jenis Kelamin

Tabel 4. 33 Tabel Hasil Pemetaan Gap Kompetensi Jenis Kelamin

No	Id Tentor	Jenis Kelamin	Bobot Profil
1	A	Pria	1
2	B	Wanita	2
3	C	Pria	1
	Profil Ideal	Pria	1
1	A		0
2	B		1
3	C		0

Contoh hasil proses perhitungan pemetaan gap jenis kelamin tertera pada Tabel 4.33. Data pada kolom jenis kelamin merupakan jenis kelamin dari tentor. Data pada kolom bobot profil merupakan normalisasi nilai berdasarkan pada norma bobot profil jenis kelamin. Profil ideal ditunjukkan dengan record berlatar

belakang abu-abu, yaitu jenis kelamin adalah Pria yang artinya adalah jenis kelamin yang diharapkan adalah Pria. Selanjutnya dilakukan perhitungan nilai gap antara bobot profil masing-masing tentor dengan profil siswa les, sehingga didapatkan gap atau selisihnya.

4. Jenjang

Contoh hasil proses perhitungan pemetaan gap jenjang tertera pada Tabel 4.34. Data pada kolom jenjang merupakan jenjang yang dikuasai tentor. Data pada kolom bobot profil merupakan normalisasi nilai berdasarkan pada norma bobot profil jenjang. Profil ideal ditunjukkan dengan record berlatar belakang abu-abu, yaitu jenjang adalah 1 yang artinya adalah jenjang siswa les adalah SD. Selanjutnya dilakukan perhitungan nilai gap antara bobot profil masing-masing tentor dengan profil siswa les, sehingga didapatkan gap atau selisihnya.

Tabel 4. 34 Tabel Hasil Pemetaan Gap Jenjang

No	Id Tendor	Nilai	Bobot Profil
1	A	SMP	2
2	B	SD	1
3	C	SMA	3
	Profil Ideal	SD	1
1	A		1
2	B		0
3	C		2

c. Penentuan Bobot Gap

Penentuan bobot gap diperoleh berdasarkan aturan dari tabel bobot nilai gap masing-masing subkriteria. Berikut adalah contoh perhitungan bobot nilai gap untuk tiap-tiap kriteria.

1. Waktu

Tabel 4.35 adalah contoh cara penentuan bobot nilai gap untuk aspek waktu.

Tabel 4. 35 Tabel Hasil Bobot Nilai Gap Waktu

No	Id Tendor	GAP_1	GAP_2	BGAP_1	BGAP_2
1	A	-2	-1	4	5
2	B	-2	2	4	4.5
3	C	0	3	6	3.5

Keterangan :

GAP_1 : Gap Hari

GAP_2 : Gap Jam

BGAP_1 : Bobot gap hari

BGAP_2 : Bobot gap jam

2. Mata Pelajaran

Tabel 4. 36 Tabel Hasil Bobot Nilai Gap Mata Pelajaran

No	Id Tentor	GAP	Bobot GAP
1	A	-1	6.5
2	B	-4	5
3	C	-6	4

Tabel 4.36 adalah contoh cara penentuan bobot nilai gap untuk aspek mata pelajaran.

3. Jenis Kelamin

Tabel 4.37 adalah contoh cara penentuan bobot nilai gap untuk aspek jenis kelamin.

Tabel 4. 37 Tabel Hasil Bobot Nilai Gap Jenis Kelamin

No	Id Tentor	GAP	Bobot GAP
1	A	0	2
2	B	1	1.5
3	C	0	2

4. Jenjang

Tabel 4.38 adalah contoh cara penentuan bobot nilai gap untuk aspek jenjang.

Tabel 4. 38 Tabel Hasil Bobot Nilai Jenjang

No	Id Tentor	GAP	Bobot GAP
1	A	1	2.5
2	B	0	3
3	C	2	1.5

d. Perhitungan dan Pengelompokan *Core Factor* dan *Secondary Factor*

Setelah menentukan bobot nilai gap untuk semua aspek/kriteria, kemudian tiap aspek dikelompokkan lagi menjadi dua kelompok yaitu *core factor* dan *secondary factor*. Berikut adalah contoh perhitungan *core factor* dan *secondary factor*.

1. Waktu

Contoh hasil perhitungan *core factor* dan *secondary factor* untuk kriteria waktu seperti pada Tabel 4.39.

Tabel 4. 39 Tabel Pengelompokan Bobot Nilai Gap Waktu

No	Id Tentor	BGAP_1	BGAP_2	Core	Secondary
1	A	5	4	5	4
2	B	4.5	4	4.5	4
3	C	3.5	6	3.5	6

Pengelompokan bobot gap pada sistem ini hanya berlaku untuk kriteria waktu karena kriteria yang lain tidak memiliki subkriteria seperti halnya waktu. Sehingga nilai pengelompokan bobot gap sama dengan nilai bobot gap.

e. Perhitungan Nilai Total Tiap Aspek

Dari hasil perhitungan NCF dan NSF tiap aspek/kriteria, kemudian dihitung nilai total berdasarkan presentase dari bobot *core factor* dan *secondary factor* yang diperkirakan berpengaruh terhadap kinerja tiap-tiap profil. Hasil presentase bobot *core factor* dan *secondary factor* diperoleh melalui hasil wawancara dengan Administrator Bimbingan Belajar *Sonic Class*, Malang. Perhitungannya dapat dilihat pada persamaan berikut.

$$\text{Nilai Total Aspek Waktu} = 60\% \text{ NCF} + 40\% \text{ NSF} \dots \dots \dots (4-1)$$

$$\text{Nilai Total Aspek Mata Pelajaran} = \text{Bobot Nilai Gap} \dots \dots \dots (4-2)$$

$$\text{Nilai Total Aspek Jenis Kelamin} = \text{Bobot Nilai Gap} \dots \dots \dots (4-3)$$

$$\text{Nilai Total Aspek Jenjang} = \text{Bobot Nilai Gap} \dots \dots \dots (4-4)$$

1. Waktu

Tabel 4.40 menunjukkan hasil perhitungan nilai total kriteria waktu. Hasil ini didapatkan dari persamaan (4-1).

Tabel 4. 40 Tabel Nilai Total Kriteria Waktu

No	Id Tentor	Core	Secondary	NWT
1	A	5	4	4.6
2	B	4.5	4	4.3
3	C	3.5	6	4.5

Pada Tabel 4.40, id tentor A memperoleh nilai 4.6, id tentor B memperoleh nilai 4.3 dan id tentor C memperoleh nilai 4.5. Nilai total itu diperoleh dari 60% *core factor* dan 40% *secondary factor*.

2. Mata Pelajaran

Tabel 4.41 menunjukkan hasil perhitungan nilai total kriteria mata pelajaran. Hasil ini didapatkan dari persamaan (4-2).

Tabel 4. 41 Tabel Nilai Total Kriteria Mata Pelajaran

No	Id Tentor	GAP	Bobot GAP	NMT
1	A	-1	6.5	6.5
2	B	-4	5	5
3	C	-6	4	4

Nilai total kriteria mata pelajaran didapat sesuai dengan bobot gap yang diperoleh. Hal ini dikarenakan aspek mata pelajaran tidak memiliki subkriteria, sehingga NCF dan NSF sama.

3. Jenis Kelamin

Tabel 4.42 menunjukkan hasil perhitungan nilai total kriteria jenis kelamin. Hasil ini didapatkan dari persamaan (4-3).

Tabel 4. 42 Tabel Nilai Total Kriteria Jenis Kelamin

No	Id Tentor	GAP	Bobot GAP	NRT
1	A	0	2	2
2	B	1	1.5	1.5
3	C	0	2	2

Nilai total kriteria jenis kelamin didapat sesuai dengan bobot gap yang diperoleh. Hal ini dikarenakan jenis kelamin tidak memiliki subkriteria, sehingga NCF dan NSF sama.

4. Jenjang

Tabel 4.43 menunjukkan hasil perhitungan nilai total kriteria jenjang. Hasil ini didapatkan dari persamaan (4-4).

Tabel 4. 43 Tabel Nilai Total Kriteria Jenjang

No	Id Tentor	GAP	Bobot GAP	NJT
1	A	1	2.5	2.5
2	B	0	3	3
3	C	2	1.5	1.5

Nilai total kriteria jenjang didapat sesuai dengan bobot gap yang diperoleh. Hal ini dikarenakan aspek jenjang tidak memiliki subkriteria, sehingga NCF dan NSF sama.

f. Perhitungan Penentuan Ranking

Perhitungan ranking atau nilai akhir adalah tahap akhir dari perhitungan *Profile Matching*. Hasil akhir berupa ranking dari tentor yang dapat dijadikan tentor bagi siswa les yang bersangkutan. Hasil presentase bobot masing-masing kriteria diperoleh melalui hasil wawancara dengan Administrator Bimbingan Belajar *Sonic Class*, Malang. Perhitungan penentuan ranking ini berdasarkan rumus sebagai berikut:

$$\text{Hasil Akhir} = 40\% \text{ NWT} + 30\% \text{ NMT} + 15\% \text{ NRT} + 15\% \text{ NJT} \dots \dots \dots (4-5)$$

Keterangan :

NWT : Nilai Total Aspek Waktu

NMT : Nilai Total Aspek Mata Pelajaran

NRT : Nilai Total Aspek Jenis Kelamin

NJT : Nilai Total Aspek Jenjang

Tabel 4. 44 Tabel Hasil Akhir

No	Id Tentor	NWT	NMT	NKT	NJT	0.4* NWT	0.3 * NMT	0.15 * NKT	0.15 * NJT	Hasil Akhir
1	A	4.6	6.5	2	2.5	1.84	1.95	0.3	0.375	4.465
2	B	4.3	5	1.5	3	1.72	1.5	0.225	0.45	3.895
3	C	4.5	4	2	1.5	1.8	1.2	0.3	0.225	3.525

Tabel 4.44 merupakan hasil akhir dari masing-masing nilai tentor. Hasil dari Tabel 4.44 selanjutnya diurutkan mulai dari peserta yang memiliki hasil akhir paling tinggi. Hasil ranking dapat dilihat pada Tabel 4.45.

Tabel 4. 45 Tabel Hasil Perankingan

No	Id Tendor	Hasil Akhir
1	A	4.465
2	B	3.895
3	C	3.525

Dari Tabel 4.45, tampak bahwa id tentor menempati urutan tertinggi. Keluaran yang dihasilkan sistem setelah melalui berbagai tahap perhitungan adalah berupa perankingan. Hasil dari perankingan tertinggi yang akan direkomendasikan oleh sistem untuk dapat dijadikan sebagai tentor terbaik untuk dapat mengajar siswa les tersebut.

4.2.8 Perancangan Antarmuka

Perancangan antarmuka merupakan kerangka pembuatan sebuah aplikasi. Tujuan dari perancangan antarmuka untuk membuat sebuah tampilan dari aplikasi yang sederhana sehingga memudahkan user dalam mengoperasikan aplikasi.

The screenshot shows a web application interface. At the top right, there is a button labeled "Admin". Below it, there are three tabs: "Bimbingan Belajar", "Siswa Les", and "Tendor". The "Tendor" tab is currently selected. The main content area is divided into two sections. On the left, there is a box labeled "Detail Siswa Les" with a "Kirim notification" button below it. On the right, there is a box labeled "Rekomendasi Tendor" containing a list of five items: 1. A, 2. B, 3. C, 4. D, and 5. E.

Gambar 4. 17 Perancangan Antarmuka Melihat Rekomendasi Tendor

Gambar 4.17 merupakan rancangan antarmuka melihat rekomendasi tentor. Pada tampilan antarmuka ini, admin dapat melihat detail dari siswa les dan juga urutan ranking dari tentor yang sudah dihitung menggunakan metode *Profile Matching*. Pada antarmuka ini juga terdapat tombol kirim *notification* untuk mengirimkan *notification* kepada tentor yang direkomendasikan sesuai urutannya. Tabel 4.46 akan memberikan keterangan mengenai antarmuka melihat rekomendasi tentor.

Tabel 4. 46 Tabel Keterangan Antarmuka Melihat Rekomendasi Tentor

Nama	Keterangan
Home	Halaman awal pada system rekomendasi tentor bimbingan belajar
Siswa Les	Menu utama yang akan merujuk pada data-data siswa les
Tentor	Menu utama yang akan merujuk pada data-data tentor
Logout	Keluar dari system
Detail Siswa Les	Menampilkan detail dari siswa yang dipilih
Rekomendasi Tentor	Menampilkan urutan ranking tentor yang sudah dilakukan perhitungan
Kirim <i>notification</i>	Tombol ini akan mengirimkan <i>notification</i> kepada tentor yang sudah direkomendasikan

Penawaran Siswa Les

Nama
Alamat
Telepon
Jenjang
Sekolah
Hari
Jam
Mata Pelajaran

Gambar 4. 18 Perancangan Antarmuka Menjawab Penawaran Admin

Gambar 4.18 merupakan rancangan antarmuka untuk mengkonfirmasi tawaran siswa les dari admin. Pada tampilan antarmuka ini, tentor dapat melihat detail dari siswa les yang ditawarkan beserta dengan jadwalnya. Selain itu, terdapat juga 2 tombol yang digunakan untuk merespon tawaran admin. Tabel 4.47 akan memberikan keterangan mengenai antarmuka melihat rekomendasi tentor.

Tabel 4. 47 Tabel Keterangan Antarmuka Menjawab Penawaran Admin

Nama	Keterangan
Detail Siswa Les	Menampilkan detail dari siswa yang ditawarkan oleh admin
Tombol “Ya”	Merespon tawaran admin dengan jawaban “ya”, yang berarti respon tersebut akan disimpan ke dalam database.
Tombol “Tidak”	Merespon tawaran admin dengan jawaban “tidak”, yang berarti respon tersebut tidak akan disimpan dalam database.

4.2.9 Perancangan Desain Uji

Pengujian pada desain uji ini dilakukan dengan tujuan untuk dapat mengetahui apakah fungsi-fungsi yang telah dimiliki oleh sistem rekomendasi tentor ini telah dapat berjalan sesuai dengan apa yang diharapkan. Adapun desain uji dari fungsi-fungsi yang diujikan terdapat pada Tabel 4.48 dan Tabel 4.49.

Tabel 4. 48 Desain Uji Melihat Rekomendasi Tentor

Test Case Id	Tujuan	Input	Output yang diharapkan	Status
1	Mendapatkan rekomendasi tentor untuk siswa les	Data siswa les dan data tentor	Urutan ranking tentor yang direkomendasikan dan dapat mengirimkan <i>notification</i> sesuai dengan ranking tersebut	

Tabel 4.48 bertujuan untuk menguji apakah tujuan dari sistem dalam mendapatkan rekomendasi tentor untuk siswa les sudah tepat. Input yang digunakan adalah data dari siswa les dan data dari tentor. Sedangkan keluaran

yang diharapkan adalah urutan ranking dari tentor yang direkomendasikan sehingga dapat dikirimkan *notification* berdasarkan urutan ranking tadi.

Tabel 4. 49 Desain Uji Menjawab Penawaran Admin

Test Case Id	Tujuan	Input	Output yang diharapkan	Status
2	Mendapatkan respon tentor yang bersedia mengajar siswa les yang ditawarkan	Respon berupa “ya” atau “tidak”	Ketika tentor menjawab “ya”, maka respon akan disimpan dalam database berupa nama tentor pada jadwal siswa les, tapi jika tentor menjawab “tidak”, maka respon tidak akan disimpan	

Tabel 4.49 digunakan untuk menguji apakah respon yang diterima dapat ditindak lanjuti sesuai dengan aturannya. Hal ini bertujuan untuk mendapatkan tentor yang bersedia mengejar siswa les yang ditawarkan. Input yang digunakan adalah respon tentor berupa “ya” atau “tidak”. Sedangkan output yang diharapkan adalah jika “ya”, maka respon akan tersimpan berupa nama tentor muncul pada jadwal siswa les, tapi jika “tidak”, maka respon tidak akan disimpan.

Sedangkan dalam pengujian akurasi, perancangan desain uji untuk keakuratan metode *Profile Matching* dapat dilihat pada Tabel 4.50. Tabel ini digunakan untuk menguji seberapa akurat daftar rekomendasi yang dihasilkan oleh sistem.

Tabel 4. 50 Tabel Desain Pengujian Akurasi

No	Sistem	Admin	Status
1	Tentor A	Tentor E	Tidak Valid
2	Tentor B	Tentor B	Valid
3	Tentor C	Tentor C	Valid
4	Tentor D	Tentor A	Tidak Valid
5	Tentor E	Tentor D	Tidak Valid

Pengujian akurasi metode *Profile Matching* dalam menentukan rekomendasi tentor yang sesuai untuk jadwal siswa les dapat dirumuskan sebagai berikut :

$$\frac{\sum \text{test case yang valid}}{\sum \text{test case yang diujikan}} \times 100\% \dots\dots\dots(4-6)$$



BAB V

IMPLEMENTASI

Bab ini membahas mengenai tahapan implementasi sistem rekomendasi tutor bimbingan belajar berdasarkan hasil analisis kebutuhan dan perancangan aplikasi. Pembahasan terdiri dari implementasi basis data yang merupakan implementasi dari perancangan persistensi class, implementasi class yang merupakan implementasi dari perancangan class diagram, implementasi kode program dan kode algoritma *Profile Matching* yang merupakan implementasi dari perancangan algoritma dan implementasi antarmuka yang merupakan implementasi dari perancangan antarmuka.

5.1 Spesifikasi Sistem

Hasil analisis kebutuhan dan perancangan perangkat lunak yang telah diuraikan pada Bab IV menjadi acuan untuk melakukan implementasi menjadi sistem yang dapat berfungsi sesuai dengan kebutuhan. Spesifikasi sistem diimplementasikan pada spesifikasi perangkat keras dan perangkat lunak.

5.1.1 Spesifikasi Perangkat Keras

Perangkat keras yang dibutuhkan untuk mengimplementasikan sistem rekomendasi tutor ini adalah:

- Processor : Intel® Core™ i3-M 380 CPU @2.53 GHz
- Memori : 6 GB
- Hardisk : 500 GB

5.1.2 Spesifikasi Perangkat Lunak

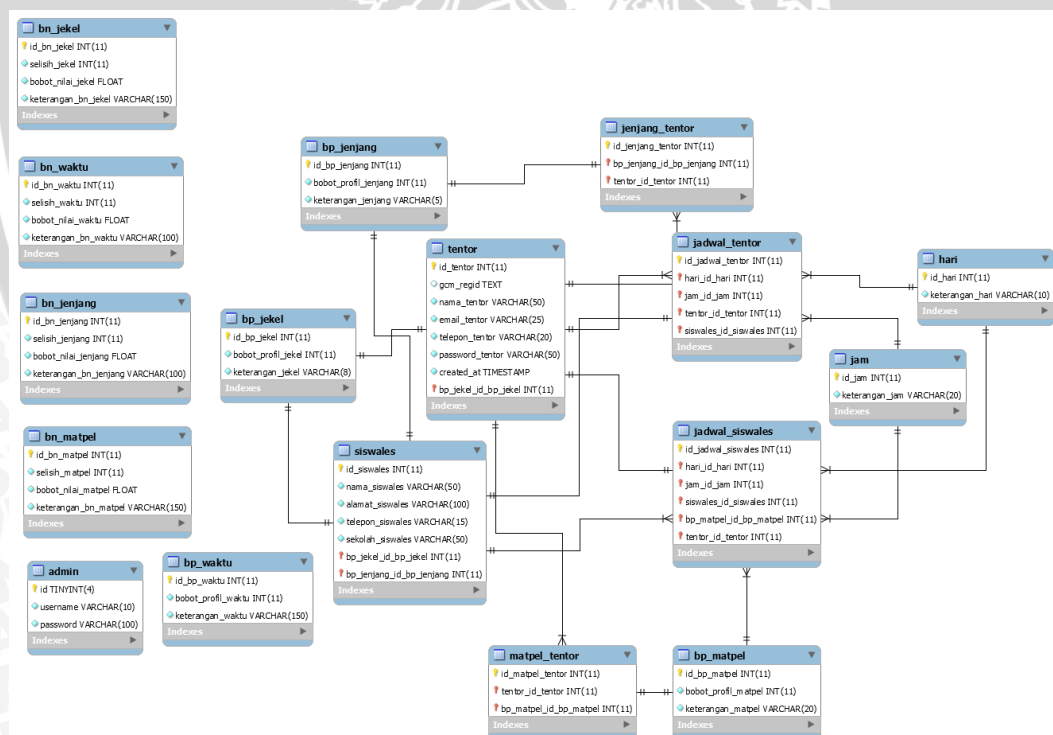
Perangkat lunak yang dibutuhkan untuk mengimplementasikan sistem rekomendasi tutor ini adalah:

- Tools Web Server : XAMPP 1.7.7
- Tools Pemrograman : Netbeans, Android Studio
- Bahasa Pemrograman : PHP, HTML, Java Android
- Browser : Google Chrome

- Sistem Operasi : Windows 8 Pro
- DBMS : MySQL

5.2 Implementasi Basis Data

Berdasarkan perancangan persistensi *class* pada Tabel 4.6, Tabel 4.7, Tabel 4.8, Tabel 4.9, Tabel 4.10, Tabel 4.11, Tabel 4.12, Tabel 4.13, Tabel 4.14, Tabel 4.15, Tabel 4.16, Tabel 4.17, Tabel 4.18, Tabel 4.19, Tabel 4.20, Tabel 4.21, dan Tabel 4.22 yang telah dijabarkan pada Bab IV, maka dapat dibuat sebuah *Entity Relationship Diagram* (ERD) yang mengimplementasikan hubungan antar tabel dalam database. Pada aplikasi tahap implementasi basis data ini menjelaskan penggunaan basis data sebagai tempat untuk menyimpan data tentor, data siswa les, dan bobot profil serta bobot nilai dari tiap kriteria yang menjadi parameter untuk merekomendasikan tentor menggunakan metode *Profile Matching*. Gambar 5.1 merupakan hasil dari implementasi database yang digunakan untuk menyimpan data pada sistem rekomendasi tentor bimbingan belajar ini.



Gambar 5. 1 Implementasi Basis Data Sistem Rekomendasi Tentor

Gambar 5. 1 merupakan implementasi basis data dari sistem rekomendasi tentor bimbingan belajar. Tabel bp_jekel, bp_waktu, bp_jenang, dan bp_matpel

merupakan tabel master untuk menentukan bobot profil dari kriteria. Sedangkan *bn_jekel*, *bn_waktu*, *bn_jenang*, dan *bn_matpel* merupakan tabel master untuk menentukan bobot nilai dari gap yang sudah dihitung dari selisih bobot profil kriteria siswa les dengan tentor. Tabel *siswales* mempunyai *id_siswales* sebagai primary key, dalam tabel ini menyimpan data-data siswa les. Sedangkan di tabel *tentor* menyimpan data-data tentor termasuk GCM registrasi id dari perangkat tentor. Tabel *jadwal_siswales* berisi data jadwal siswa les yang terdiri dari *id_jadwal_siswales* sebagai primary key, dan *id_hari*, *id_jam*, *id_matpel* dan *id_tentor* sebagai *foreign key* yang berpacu dari tabel lain. Tabel *jadwal_tentor*, *matpel_tentor*, dan *jenjang_tentor* berisi jadwal kosong tentor, mata pelajaran dan jenjang yang dikuasai tentor. Satu orang tentor dapat mempunyai waktu kosong dan mata pelajaran serta jenjang yang dikuasai lebih dari satu, maka perlu dibuat tabel baru yang menyimpannya.

5.3 Implementasi Class

Setiap *class* yang telah dirancang pada perancangan diagram *class* dalam Bab IV direalisasikan pada sebuah file. Dalam Gambar 4.8 yang merupakan perancangan diagram *class* dari melihat rekomendasi tentor direalisasikan pada Tabel 5.1 dengan mengimplementasikan *class* tersebut ke dalam beberapa file *.php*. Implementasi *Class* ini menjelaskan mengenai nama *package* yang digunakan, tipe dari objek, nama *class*nya dan juga nama file yang mengimplementasikan *class* diagram pada Gambar 4.8.

Tabel 5. 1 Implementasi *Class* Melihat Rekomendasi Tentor

No	Package	Objek	Nama Class	Nama File Program
1	Server	View dan Controller	Siswales	Siswales.php
2	Server	View	V_rekomendasi	V_rekomendasi.php
3	Server	Controller	Push_notif	Push_notif.php

Hasil dari implementasi *class* diagram melihat rekomendasi tentor dapat dilihat dalam Tabel 5.1 yang menjelaskan mengenai *package*, objek, nama *class*, dan nama file program dari use case melihat rekomendasi tentor. Di dalam *siswales.php* dijabarkan mengenai algoritma dari metode *Profile Matching*, sedangkan di *push_notif.php* terdapat fungsi untuk mengirimkan *push notification*

kepada tentor yang direkomendasikan. `V_rekomendasi.php` merupakan view yang menampilkan detail dari siswa les dan juga daftar tentor yang direkomendasikan.

Class Diagram menjawab penawaran admin yang telah dirancang pada Gambar 4.9 diimplementasikan ke dalam file dengan ekstensi `.java` dan file layout `.xml` sebagai tampilan dari *class* tersebut. Implementasi dilakukan dengan cara membuat tabel yang berisi daftar nama *package*, nama objek, nama *class*, nama file `.java` dan juga nama file `.xml` dari *class* yang sudah dirancang pada Gambar 4.9.

Tabel 5. 2 Implementasi *Class* Menjawab Penawaran Admin

No	Package	Objek	Nama Class	Nama Program	Nama File	Nama File Layout
1	com.skripsi.bimbel	Controller	Penawaran	Penawaran.java	Activity_penawaran.xml	
2	com.skripsi.bimbel	Controller	GCMIntentService	GCMIntentService.java	-	

Hasil dari implementasi *class diagram* menjawab penawaran admin dapat dilihat pada Tabel 5.2 yang menjelaskan mengenai pasangan antara *class* dengan file program dan layout yang digunakan untuk mengimplementasikan use case menjawab penawaran admin. Di dalam `GCMIntentService` akan diterima variabel yang akan ditampilkan di *notification* di perangkat tentor. Jika tentor menerima notifikasi, maka notifikasi tersebut akan ditampilkan melalui `activity_penawaran.xml`. Sedangkan di `penawaran.java` terdapat fungsi yang akan dijalankan jika tentor merespon tawaran admin berupa jawaban “ya” atau “tidak”.

5.4 Implementasi Kode Program

Berdasarkan flowchart metode *Profile Matching* pada Gambar 4.14 dapat dilihat tahapan dari algoritma ini dalam menentukan tentor yang sesuai untuk siswa les. Perancangan metode *Profile Matching* tersebut diimplementasikan ke dalam beberapa kode program. Implementasi kode program untuk diagram *class* melihat rekomendasi tentor yang telah dirancang pada Gambar 4.8 terdapat beberapa *class*, yaitu view `jadwal_siswales`, controller `siswales`, view `v_rekomendasi` dan model `model_master`. Dari *class* tersebut, implementasi metode *Profile Matching*

untuk mendapatkan rekomendasi tentor bimbingan belajar terdapat pada method rekomendasi().

```
//kriteria jenjang
echo "<br><br>Jenjang<br>";
1. $result2 = mysql_query("select id_jenjang from
   jenjang_tentor where id_tentor = '$id_tentor'");
   $bobot_max3 = 0;
2. while ($row6 = mysql_fetch_assoc($result2)) {
   $id_jenjang = $row6['id_jenjang'];
3. $selisih_jenjang = $id_jenjang -
   $jenjang_siswales;
4. $bobot_jenjang = "SELECT bobot_nilai_jenjang from
   bn_jenjang where selisih_jenjang =
   '$selisih_jenjang'";
5. $res_bobot_jenjang = mysql_query($bobot_jenjang);
6. while ($row7 =
   mysql_fetch_assoc($res_bobot_jenjang)) {
   $bobot_jenjang =
   $row7['bobot_nilai_jenjang'];
7. if ($bobot_max3 < $bobot_jenjang) {
   $bobot_max3 = $bobot_jenjang;
   }
   }
```

Gambar 5. 2 Implementasi Code Menghitung Kriteria Jenjang

Gambar 5.2 diatas merupakan hasil implementasi dari perancangan algoritma *Profile Matching* yang menjelaskan mengenai perhitungan dari kriteria jenjang. Perhitungan tersebut sama dengan perhitungan untuk kriteria jenis kelamin dan mata pelajaran. Baris kode 1 dan 2 adalah kode untuk mengambil data jenjang yang dimiliki masingmasing tentor. Baris 3 menjelaskan mengenai perhitungan gap yaitu selisih jenjang yang dikuasai tentor dengan jenjang siswa les. Setelah didapat selisihnya, maka dicari bobot nilai berdasarkan selisih tersebut yang dijelaskan oleh baris 4, 5 dan 6. Satu tentor dapat menguasai beberapa jenjang, maka untuk ke proses berikutnya diambil jenjang yang bobotnya paling besar yang dijelaskan pada baris 7. Sehingga didapatkan nilai total NJT yang akan diproses ke tahap selanjutnya.

Selain menghitung bobot dari kriteria jenjang, mata pelajaran dan jenis kelamin, akan dihitung juga nilai waktu sebagai salah satu kriteria dalam penentuan rekomendasi tentor ini. Perhitungan pada kriteria ini sedikit berbeda dengan perhitungan pada kriteria jenjang, jenis kelamin dan mata pelajaran, karena dalam kriteria ini terdapat *core factor* yaitu hari dan *secondary factor* yaitu

jam. Sehingga dalam penentuan perhitungan total didapat dari penjumlahan 60% NCF dan 40% NSF yang dapat dilihat pada baris 14. Presentase ini didapat melalui proses wawancara dan analisis kebutuhan yang dilakukan pada Bab IV. Bobot nilai terbesar akan dipilih dari banyak jadwal kosong yang mempunyai masing--masing tentor utuk dapat diproses di tahap selanjutnya. Gambar 5.3 merupakan hasil dari implementasi perhitungan kriteria waktu dalam menentukan rekomendasi tentor.

```
//kriteria waktu
1. while ($rows =
mysql_fetch_assoc($id_tentor_siswales)) {
    $id_tentor = $rows['id_tentor'];
    echo "<br>$id_tentor <br>";
2. $result1 = mysql_query("select id_hari, id_jam from
jadwal_tentor where id_tentor = '$id_tentor'");
    $nwt = 0;
3. while ($row2 = mysql_fetch_assoc($result1)) {
    $id_hari = $row2['id_hari'];
    $id_jam = $row2['id_jam'];
4. $selisih_hari = $id_hari - $hari_siswales;
5. $selisih_jam = $id_jam - $jam_siswales;
6. $bobot_hari = "SELECT bobot_nilai_waktu from
bn_waktu where selisih_waktu = '$selisih_hari'";
7. $bobot_jam = "SELECT bobot_nilai_waktu from bn_waktu
where selisih_waktu = '$selisih_jam'";
8. $res_bobot_hari = mysql_query($bobot_hari);
9. $res_bobot_jam = mysql_query($bobot_jam);
10. while ($row3 = mysql_fetch_assoc($res_bobot_hari)) {
11. $bobot_hari = $row3['bobot_nilai_waktu'];
    }
12. while ($row4 = mysql_fetch_assoc($res_bobot_jam)) {
13. $bobot_jam = $row4['bobot_nilai_waktu'];
    }
14. $nilai_waktu_total = (0.6 * $bobot_hari) + (0.4 *
$bobot_jam);
15. if ($nwt < $nilai_waktu_total) {
    $nwt = $nilai_waktu_total;
    }
}
```

Gambar 5. 3 Implementasi Code Menghitung Kriteria Waktu

Setelah didapatkan nilai total dari masing-masing kriteria, maka dapat dihitung nilai total dari metode *Profile Matching* yang nantinya akan diurutkan berdasarkan rekomendasi dengan bobot terbesar. *Array_push* digunakan jika isi dari array tersebut dinamis. Gambar 5.4 merupakan hasil implementasi dari

tahapan selanjutnya dari metode *Profile Matching* pada Gambar 5.2 dan Gambar 5.3. Nilai total diberi bobot presentase untuk masing-masing kriteria. Bobot ini diperoleh melalui wawancara dan analisis kebutuhan yang didapat pada Bab IV. Baris 5 adalah implementasi kode untuk mengurutkan rekomendasi dari nilai total yang sudah didapat tadi. Setelah itu untuk dapat mengirimkan `gcm_regid` secara berurutan sesuai rekomendasi maka dibentuk array yang dapat dilihat pada baris 11 dan 12 ke tombol yang akan mengirimkan notifikasi.

```
//Nilai total dan perankingan
1. $total = (0.3 * $nwt) + (0.3 * $nmt) + (0.15 * $nrt)
   + (0.15 * $njt);
   echo "<strong><br><br>$id_tentor TOTAL NILAI ADALAH
   $total<br><br></strong>";

2. $this->array_push_assoc($rekomendasi, $id_tentor,
   $total);
3. $rekomendasi[$id_tentor] = $total;
   }
   echo "Kumpulan Nilai Total<br>";
4. foreach ($rekomendasi as $id_tentor => $nilai_total)
   {
       echo "<br>$id_tentor =>
   $nilai_total<br>";
   }
   echo "<br>Rekomendasinya adalah<br>";
5. arsort($rekomendasi);
6. $regId = array();
7. $idRek = array();
8. foreach ($rekomendasi as $key => $value) {
   echo "<br>$key => $value<br>";
9. $nama_rekomendasi = mysql_query("select id_tentor,
   nama_tentor, gcm_regid from tentor where
   id_tentor='$key'");
10. while ($row5 = mysql_fetch_assoc($nama_rekomendasi))
   {
       $id_tentor = $row5['nama_tentor'];
       $id_tentorRek = $row5['id_tentor'];
       $gcm_regid = $row5['gcm_regid'];
       echo "$id_tentor yang idnya $id_tentorRek mempunyai
   $gcm_regid ";
11. array_push($regId, $gcm_regid);
12. array_push($idRek, $id_tentorRek);
   }
   }
```

Gambar 5. 4 Implementasi Code Penentuan Ranking

```
function respon_tidak() {  
1.   $un = $_POST["username"];  
     $idRek = $_POST["idRek"];  
     $id_siswa = $_POST["id_siswa"];  
     $message = $_POST["message"];  
     $regId = $_POST["regId"];  
     $id_jadwal_siswa = $_POST["id_jadwal_siswa"];  
2.   $query_id = mysql_query("select id_tentor from  
     tentor where gcm_regid = '$regId'");  
3.   while ($row = mysql_fetch_assoc($query_id)) {  
     $id_rekomendasi = $row["id_tentor"];  
     }  
4.   $array_idRek = json_decode($idRek);  
5.   $key = array_search($id_rekomendasi, $array_idRek);  
6.   $id_next = $key + 1;  
7.   $id_array_next = $array_idRek[$id_next];  
8.   $query_next = mysql_query("select gcm_regid from  
     tentor where id_tentor='$id_array_next'");  
9.   while ($rowx= mysql_fetch_assoc($query_next)) {  
     $reg_next = $rowx["gcm_regid"];  
     }  
10.  $registatoin_ids = array($reg_next);  
11.  $message = array("price" => $message, "idRek" =>  
     $idRek, "id_siswa" => $id_siswa, "id_jadwal_siswa"  
     => $id_jadwal_siswa, 'regId' => $reg_next);  
12.  $this->send_notification($registatoin_ids,  
     $message);  
     }
```

Gambar 5. 5 Implementasi Code Respon Tidak

Setelah notifikasi terkirim ke tentor yang pertama direkomendasikan, maka tentor akan mendapat notifikasi berupa data siswa les beserta jadwalnya. Dalam notifikasi tersebut terdapat tombol untuk merespon tawaran dari admin tersebut. Jika tentor merasa sesuai, maka tentor akan menjawab “ya”, respon “ya” tersebut akan disimpan dalam database berupa penyimpanan id tentor ke jadwal_siswales dan id siswa les dalam jadwal tentor. Sedangkan jika “tidak”, maka dapat dilihat pada Gambar 5.5 bahwa ketika tombol “tidak” ditekan, maka akan dikirim kembali pesan, array yang berisi id tentor rekomendasi, gcm_regid dari tentor yang mengirim, dan id jadwal siswa les yang ditunjukkan pada kode nomor 1. Setelah itu akan di cek gcm_regid tersebut adalah kepunyaan id tentor berapa, lalu pada baris 5 akan dicari id tentor tersebut ada pada index ke berapa di dalam array rekomendasi. Maka untuk mengirimkan notifikasi ke id tentor

berikutnya akan diambil `gcm_regid` dari index berikutnya. Pada baris 11 dan 12, akan dimasukkan ke dalam array pesan yang dikirim melalui variabel `message`, lalu untuk mengirimkannya dipanggil method `send_notification`.

```
Public function
send_notification($registatoin_ids, $message) {
include_once './config.php';
// Set POST variables
1. $url =
'https://android.googleapis.com/gcm/send';
2. $fields = array(
'registration_ids' => $registatoin_ids,
'data' => $message);
3. $headers = array(
'Authorization: key=' . GOOGLE_API_KEY,
'Content-Type: application/json');
// Open connection
4. $ch = curl_init();
// Set the url, number of POST vars, POST data
5. curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
// Disabling SSL Certificate support temporarily
6. curl_setopt($ch, CURLOPT_SSL_VERIFYPEER,
false);
curl_setopt($ch, CURLOPT_POSTFIELDS,
7. json_encode($fields));
// Execute post
8. $result = curl_exec($ch);
9. if ($result === FALSE) {
die('Curl failed: ' . curl_error($ch));
}
// Close connection
10. curl_close($ch);
echo $result;
}
```

Gambar 5. 6 Implementasi Code Mengirim Notifikasi

Gambar 5.6 diatas menjelaskan pengiriman *push notification* kepada tentor. Implementasi method tersebut akan mengirimkan permintaan pada <https://android.googleapis.com/gcm/send> selaku GCM server untuk memberikan persetujuan pengiriman pesan serta meminta GCM server untuk mengirimkan pesan notifikasi kepada perangkat sesuai dengan registration id yang dimiliki masing-masing perangkat.

5.5 Implementasi Antarmuka

Berdasarkan perancangan antarmuka melihat rekomendasi tentor pada Gambar 4.15 dapat diimplementasikan dalam Gambar 5.7. Gambar 5.7 merupakan hasil dari implementasi Gambar 4.15 yang menjelaskan mengenai implementasi antarmuka dari use case melihat ranking tentor yang menampilkan detail siswa les yang terdiri dari nama, alamat, telepon, jenjang, sekolah, hari, jam dan mata pelajaran dan urutan ranking dari tentor yang direkomendasikan untuk jadwal siswa les tersebut. Tendor yang diambil hanya 5 rekomendasi saja dari tentor yang ada di Bimbingan Belajar. Dalam halaman ini juga terdapat tombol “Kirim *Notification*” untuk mengirimkan *notification* kepada tentor pertama yang direkomendasikan.

The screenshot shows a web application interface for 'Bimbingan Belajar'. The navigation bar includes 'Bimbingan Belajar', 'Siswa Les', 'Tendor', and an 'Admin' dropdown menu. The main content area is titled 'Rekomendasi' and is divided into two columns: 'Detail Siswa Les' and 'Rekomendasi'.

Detail Siswa Les	Rekomendasi
Nama : Orlando	Lilis Andriani
Alamat : Jalan Melawai 4B	eric septian
Telepon : 081336069028	Luluk Setiawati Hartono
Jenjang : SMP	Agus Satria
Sekolah : Santa Maria II	Rino sebastian
Hari : Selasa	
Jam : 13.00-14.30	
Mata Pelajaran : Matematika	

At the bottom of the 'Detail Siswa Les' section, there is a blue button labeled 'Kirim Notification'.

Gambar 5. 7 Implementasi Antarmuka Melihat Rekomendasi Tendor

Perancangan antarmuka pada Gambar 4.16 merupakan perancangan antarmuka untuk menjawab penawaran admin. Perancangan tersebut diimplementasikan dalam layout yang dapat memenuhi kebutuhan aktor. Hasil implementasi menjawab penawaran admin dapat dilihat pada Gambar 5.8 yang menampilkan detail siswa les beserta jadwalnya dan terdapat tombol untuk merespon penawaran admin tersebut yaitu “ya” dan “tidak”. Tombol ini akan menentukan respon apa yang akan dijalankan.

 Penawaran Siswa Les

Apakah Anda ingin mengajar siswa les berikut ?

Nama : Putri

Alamat Jalan Kebalen no 40

Telepon 08986382780

Jenjang SMA

Sekolah Cor Jesu

Hari Rabu

Jam 17.00-18.30

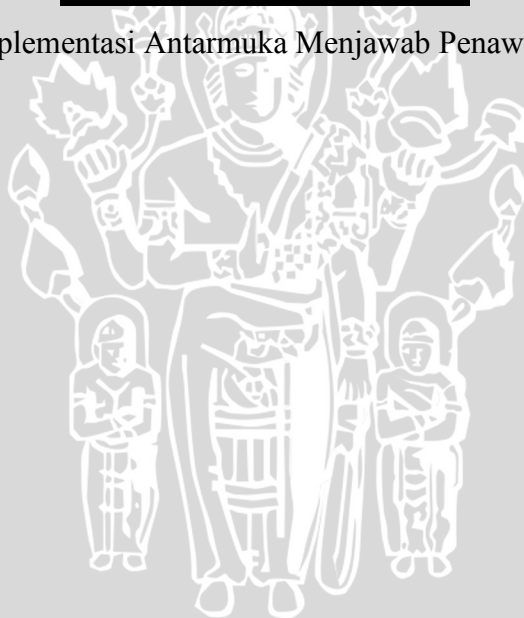
Matpel Bahasa Inggris

Ya

Tidak



Gambar 5. 8 Implementasi Antarmuka Menjawab Penawaran Admin



BAB VI

PENGUJIAN DAN ANALISIS

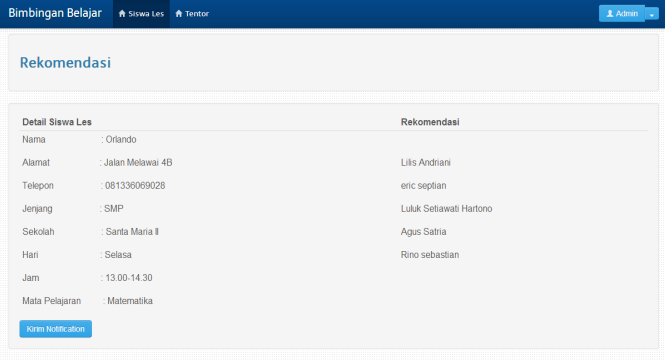
Pada bab ini akan dibahas 2 macam pengujian terhadap sistem yang dibangun, yaitu pengujian *black box* dan pengujian akurasi. Pengujian *black box* ini dilakukan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan perancangan. Pengujian akurasi dilakukan untuk mengetahui tingkat akurasi dari sistem terhadap hasil keluaran yang telah dihitung menggunakan metode *Profile Matching*.

6.1 Pengujian *Black Box*

Berdasarkan kebutuhan fungsional pada Tabel 4.2, kebutuhan untuk melihat rekomendasi tentor dan mengirimkan *push notification* ke tentor yang direkomendasikan, akan diuji fungsionalitasnya. Pengujian dilakukan dengan cara melihat apakah ada daftar rekomendasi yang dihasilkan untuk jadwal siswa les tertentu. Pengujian ini juga akan menguji apakah saat ditekan tombol *push notification*, pesan dapat diterima oleh tentor rekomendasi pertama atau tidak. Hasil yang diharapkan adalah sistem mampu menampilkan rekomendasi sesuai dengan perhitungan metode *Profile Matching* dan mampu mengirimkan notifikasi ke tentor pertama yang direkomendasikan. Tabel 6.1 merupakan detail dari kebutuhan melihat rekomendasi tentor yang diujikan.

Table 6. 1 Tabel Kasus Uji Melihat Rekomendasi Tentor

Kode Uji	P01
Nama Kasus Uji	Kasus Uji Melihat Rekomendasi Tentor
Objek Uji	Kebutuhan Fungsional F03
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan hasil rekomendasi tentor yang telah dihitung menggunakan metode <i>Profile Matching</i> dan mengirimkan notifikasi berdasarkan rekomendasi tentor tadi.
Data	Data Detail Siswa Les


	<p>Nama: Agnes Alamat: Jalan Tokala no 4 Telepon: 08115820199 Jenjang: SMA Sekolah: Dempo Hari: Selasa Jam: 15.00-16.30 Mata Pelajaran: Biologi</p> <p>Data Rekomendasi Tentor</p> <p>Array ([0] => 12 [1] => 1 [2] => 11 [3] => 16 [4] => 17)</p>
<p>Prosedur Uji</p>	<ol style="list-style-type: none"> 1. Admin login ke dalam sistem 2. Admin memilih siswa les yang akan dilihat jadwalnya 3. Admin memilih jadwal siswa les yang ingin dilihat rekomendasinya 4. Admin melihat ranking tentor yang telah direkomendasikan menggunakan metode <i>Profile Matching</i> 5. Admin menekan tombol kirim <i>push notification</i> 6. <i>Notification</i> terkirim ke tentor rekomendasi pertama
<p>Hasil yang diharapkan</p>	<p>Admin melihat rekomendasi tentor yang sesuai atau mendekati kriteria siswa les dan <i>notification</i> dapat terkirim ke tentor rekomendasi pertama.</p>
<p>Hasil Uji</p>	
<p>Analisis Uji</p>	<p>Saat admin menekan tombol rekomendasi, sistem dapat menampilkan detail siswa les beserta</p>

	jadwalnya dan sistem juga dapat menampilkan tentor yang direkomendasikan untuk siswa les dengan jadwal tertentu. Saat admin menekan tombol kirim <i>notification</i> , maka notifikasi yang berisi detail siswa les beserta jadwalnya akan terkirim ke tentor rekomendasi pertama. Berdasarkan dengan hasil uji diatas maka kasus uji melihat rekomendasi tentor yang direkomendasikan dengan metode <i>Profile Matching</i> dan mengirimkan <i>push notification</i> ke tentor pertama 100% valid.
--	---

Berdasarkan kebutuhan fungsional pada Tabel 4.2, kebutuhan untuk menjawab penawaran admin akan diuji fungsionalitasnya. Pengujian dilakukan dengan cara melihat apakah sistem mampu menerima respon dari perangkat tentor atas tawaran yang diberikan oleh admin. Pengujian ini juga akan menguji apakah saat merespon penawaran admin, tindakan yang diberikan atas respon tentor sudah benar atau belum. Hasil yang diharapkan adalah sistem mampu menerima respon dari perangkat tentor dan tindakan yang akan dilakukan sistem adalah jika tentor menjawab “ya”, maka respon akan disimpan ke database, sedangkan jika jawaban “tidak”, maka *notification* akan diteruskan ke tentor rekomendasi kedua dan seterusnya. Tabel 6.2 merupakan detail dari kasus uji dari kebutuhan menjawab penawaran admin.

Table 6. 2 Tabel Kasus Uji Menjawab Penawaran Admin

Kode Uji	P02
Nama Kasus Uji	Kasus Uji Menjawab Penawaran Admin
Objek Uji	Kebutuhan Fungsional F04
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memberikan respon penawaran kepada admin. Pengujian ini juga menguji tindakan yang akan dilakukan dari respon tersebut.
Data	-
Prosedur Uji	<ol style="list-style-type: none"> 1. Tantor menerima notifikasi 2. Tantor membuka notifikasi yang berisi data siswa les beserta jadwalnya. 3. Tantor menjawab penawaran admin tersebut dengan jawaban “ya” atau “tidak”
Hasil yang diharapkan	Jika respon yang diberikan adalah “ya”, maka

	<p>id siswa les akan tersimpan di jadwal tentor dan id tentor akan tersimpan di jadwal siswa les. Jika “tidak”, maka notifikasi akan otomatis terkirim ke rekomendasi tentor berikutnya.</p>
<p>Hasil Uji</p>	 <p>The screenshot shows a mobile phone interface. At the top, the time is 17:54 on Thursday, 21 May 2015. A notification from 'Apakah Anda ingin mengajar...' is visible. Below it, there's a 'BBM Connected' notification and a '16.0.B.216 New system update available' notification. At the bottom, there's a 'Penawaran Siswa Les' notification with the following details: 'Apakah Anda ingin mengajar siswa les berikut?', 'Nama : Putri', 'Alamat Jalan Kebalen no 40', 'Telepon 08986382780', 'Jenjang SMA', 'Sekolah Cor Jesu', 'Hari Rabu', 'Jam 17.00-18.30', and 'Matpel Bahasa Inggris'. At the bottom of the notification, there are two buttons: 'Ya' and 'Tidak'.</p>
<p>Analisis Uji</p>	<p>Saat tentor membuka notifikasi yang telah diterima, tentor akan menjawab penawaran admin yang diberikan berupa respon jawaban “ya” atau “tidak”. Berdasarkan dengan hasil uji tindakan atas respon “ya” maupun “tidak” maka menjawab penawaran admin 100% valid.</p>

Dari detail pengujian fungsional pada Tabel 6.1 dan Tabel 6.2 di atas dapat disimpulkan ke dalam Tabel 6.3 berikut ini.

Table 6. 3 Tabel Kesimpulan Pengujian Fungsional

Test Case Id	Tujuan	Input	Output yang diharapkan	Status
1	Mendapatkan rekomendasi tentor untuk siswa les	Data siswa les dan data tentor	Urutan ranking tentor yang direkomendasikan dan dapat mengirimkan <i>notification</i> sesuai dengan ranking tersebut	Valid
2	Mendapatkan respon tentor yang bersedia mengajar siswa les yang ditawarkan	Respon berupa “ya” atau “tidak”	Ketika tentor menjawab “ya”, maka respon akan disimpan dalam database berupa nama tentor pada jadwal siswa les, tapi jika tentor menjawab “tidak”, maka respon tidak akan disimpan	Valid

6.2 Pengujian Akurasi

Pada pengujian akurasi sistem dilakukan uji coba terhadap aplikasi dengan menggunakan data dari *Sonic Class* Bimbingan Belajar tahun 2014/2015. Pengujian ini akan menggunakan data tentor dan data siswa les yang berasal dari Bimbingan Belajar *Sonic Class*, Malang bulan Mei tahun 2015. Dalam pengujian ini juga akan dilakukan tes pada 3 jadwal siswa les yang akan dilihat rekomendasinya dan juga akan mengirimkan notifikasi tersebut kepada tentor yang direkomendasikan. Pada skenario pengujian ini terdapat beberapa kondisi berikut :

- Jumlah data tentor sejumlah 14 orang
- Jumlah data siswa les sejumlah 41 orang dan terdapat 62 jadwal siswa les
- Dari 14 tentor diambil 5 tentor yang paling mendekati jadwal dan kriteria siswa les
- Jadwal siswa les yang akan diuji adalah sebanyak 3 jadwal

- e. Penentuan tentor yang lolos berdasarkan kriteria jadwal (bobot 40%), kriteria mata pelajaran (bobot 30%), kriteria jenjang (bobot 15%) dan kriteria jenis kelamin (bobot 15%).

Pengujian :

1. Jadwal Siswa Les Pertama

Tabel 6.4 di bawah ini merupakan tabel yang menjelaskan mengenai data siswa les. Siswa les pada pengujian pertama ini bernama Orlando yang mempunyai bobot profil jenis kelamin 1 yang artinya jenis kelamin pria. Jenjang siswa les ini mempunyai bobot profil 2 yang artinya jenjang SMP. Mata pelajaran yang ingin dipelajari di bimbingan belajar ini mempunyai bobot profil 4 yang artinya pelajaran Matematika. Siswa les ini ingin les pada jadwal Selasa dan jam 13.00-14.30 yang mempunyai bobot profil hari 2 dan bobot profil jam 3.

Table 6. 4 Tabel Data Siswa Les Uji Pertama

No	Keterangan Kriteria	Bobot Profil	
1	Nama	Orlando	
2	Jenis Kelamin	1	
3	Jenjang	2	
4	Mata Pelajaran	4	
5	Jadwal les (hari jam)	2	3

Tabel 6.5 di bawah ini merupakan hasil perbandingan keluaran dari sistem dengan keputusan dari admin untuk data uji pertama. Dapat diketahui dari 5 tentor yang direkomendasikan oleh sistem, ada 3 tentor yang cocok dengan keputusan admin. Sehingga dapat dihitung nilai akurasi dan mendapat presentase 60%.

Table 6. 5 Tabel Pengujian Akurasi Data Uji Pertama

No	Sistem	Admin	Status
1	Lilis	Lilis	Valid
2	Eric	Stella	Tidak Valid
3	Luluk	Luluk	Valid
4	Agus	Santi	Tidak Valid
5	Rino	Rino	Valid

$$\begin{aligned}
 \text{Akurasi} &= \frac{\sum \text{test case yang valid}}{\sum \text{test case yang diujikan}} \times 100\% \\
 &= \frac{3}{5} \times 100\% \\
 &= 60\%
 \end{aligned}$$

2. Jadwal Siswa Les Kedua

Tabel 6.6 di bawah ini merupakan tabel yang menjelaskan mengenai data siswa les. Siswa les pada pengujian kedua ini bernama Gladys yang mempunyai bobot profil jenis kelamin 2 yang artinya jenis kelamin wanita. Jenjang siswa les ini mempunyai bobot profil 2 yang artinya jenjang SMP. Mata pelajaran yang ingin dipelajari di bimbingan belajar ini mempunyai bobot profil 6 yang artinya pelajaran Biologi. Siswa les ini ingin les pada jadwal Jumat dan jam 11.00-12.30 yang mempunyai bobot profil hari 5 dan bobot profil jam 2.

Table 6. 6 Tabel Data Siswa Les Uji Kedua

No	Keterangan Kriteria	Bobot Profil
1	Nama	Gladys
2	Jenis Kelamin	2
3	Jenjang	2
4	Mata Pelajaran	6
5	Jadwal les (hari jam)	5 2

Tabel 6.7 di bawah ini merupakan hasil perbandingan keluaran dari sistem dengan keputusan dari admin untuk data uji kedua. Dapat diketahui dari 5 tentor yang direkomendasikan oleh sistem, ada 5 tentor yang cocok dengan keputusan admin. Sehingga dapat dihitung nilai akurasi dan mendapat presentase 100%.

Table 6. 7 Tabel Pengujian Akurasi Data Uji Kedua

No	Sistem	Admin	Status
1	Vita	Vita	Valid
2	Alita	Alita	Valid
3	Philip	Philip	Valid
4	Lilis	Lilis	Valid
5	Luluk	Luluk	Valid

$$\begin{aligned}
 \text{Akurasi} &= \frac{\sum \text{test case yang valid}}{\sum \text{test case yang diujikan}} \times 100\% \\
 &= \frac{5}{5} \times 100\% \\
 &= 100\%
 \end{aligned}$$

3. Jadwal Siswa Les Ketiga

Tabel 6.8 di bawah ini merupakan tabel yang menjelaskan mengenai data siswa les. Siswa les pada pengujian ketiga ini bernama Jeremy yang mempunyai

bobot profil jenis kelamin 1 yang artinya jenis kelamin pria. Jenjang siswa les ini mempunyai bobot profil 3 yang artinya jenjang SMA. Mata pelajaran yang ingin dipelajari di bimbingan belajar ini mempunyai bobot profil 11 yang artinya pelajaran Akuntansi. Siswa les ini ingin les pada jadwal Senin dan jam 11.00-12.30 yang mempunyai bobot profil hari 1 dan bobot profil jam 2.

Table 6. 8 Tabel Data Siswa Les Uji Ketiga

No	Keterangan Kriteria	Bobot Profil	
1	Nama	Jeremy	
2	Jenis Kelamin	1	
3	Jenjang	3	
4	Mata Pelajaran	11	
5	Jadwal les (hari jam)	1	2

Tabel 6.9 di bawah ini merupakan hasil perbandingan keluaran dari sistem dengan keputusan dari admin untuk data uji pertama. Dapat diketahui dari 5 tentor yang direkomendasikan oleh sistem, ada 4 tentor yang cocok dengan keputusan admin. Sehingga dapat dihitung nilai akurasi dan mendapat presentase 80%.

Table 6. 9 Tabel Pengujian Akurasi Data Uji Ketiga

No	Sistem	Admin	Status
1	Rino	Rino	Valid
2	Alita	Alita	Valid
3	Stella	Stella	Valid
4	Vani	Vani	Valid
5	Sugi	Santi	Tidak Valid

$$\begin{aligned}
 \text{Akurasi} &= \frac{\sum \text{test case yang valid}}{\sum \text{test case yang diujikan}} \times 100\% \\
 &= \frac{4}{5} \times 100\% \\
 &= 80\%
 \end{aligned}$$

Dari Tabel 6.5, Tabel 6.7 dan Tabel 6.9 dapat diketahui status data uji yang dibandingkan dengan keputusan administrator bimbingan belajar. Dari ketiga data uji diatas, dapat dihitung rata-rata pengujian akurasi kedua adalah sebagai berikut:

$$\begin{aligned}\text{Rata-rata akurasi} &= \frac{\text{akurasi data 1} + \text{akurasi data 2} + \text{akurasi data 3}}{3} \\ &= \frac{60\%+100\%+80\%}{3} \\ &= 80\%\end{aligned}$$

Dari perhitungan rata-rata akurasi diatas dapat disimpulkan bahwa tingkat akurasi dari perhitungan *Profile Matching* jika dibandingkan dengan keputusan dari Administrator Bimbingan Belajar sebesar 80%. Nilai akurasi 80% ini dikatakan baik karena sistem ini digunakan bukan untuk menggantikan peran administrator tetapi membantu administrator untuk menentukan tentor yang sesuai. Selain itu, perbedaan keputusan yang dihasilkan ini disebabkan karena dalam proses pengambilan keputusan, administrator hanya berfokus pada beberapa kriteria seleksi, tidak semua kriteria diperhitungkan.



BAB VII PENUTUP

7.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang dilakukan maka dapat diambil kesimpulan sebagai berikut:

1. Sistem rekomendasi tutor bimbingan belajar dengan menggunakan metode *Profile Matching* ini diimplementasikan dengan 4 macam kriteria, yaitu jenis kelamin, jenjang, mata pelajaran dan waktu. Pada masing-masing kriteria didapatkan gap untuk tiap kriteria dan dihitung nilai total dari *core factor* dan *secondary factor* untuk mendapatkan nilai akhir metode *Profile Matching*. Setelah mendapatkan nilai total, masing-masing tutor diberi ranking untuk menentukan rekomendasi.
2. Sistem rekomendasi tutor bimbingan belajar telah diimplementasikan pada perangkat bergerak berbasis Android dengan memanfaatkan teknologi *Google Cloud Messaging* dan dapat digunakan sebagai salah satu media untuk menawarkan siswa les dari admin kepada tutor yang direkomendasikan. Selain itu, sistem ini juga dapat membantu proses komunikasi dari admin kepada tutor secara efektif.
3. Pada proses pengujian akurasi sistem dengan menggunakan 3 macam studi kasus jadwal siswa les, tingkat akurasi yang dihasilkan pada perhitungan metode *Profile Matching* adalah 80%.

7.2 Saran

Saran untuk penelitian selanjutnya dengan berdasarkan penelitian ini adalah perlu dilakukan pemeriksaan terhadap kriteria untuk rekomendasi dan kemungkinan adanya penambahan kriteria. Perlu ditambahkan status dari penerima notifikasi penawaran siswa les kepada tutor agar pengiriman notifikasi dapat terkontrol dengan baik oleh administrator.

DAFTAR PUSTAKA

- [ASA-13] Santoso, Agus dan Handojo, Andreas.2013."Pembuatan Aplikasi Mobile Broadcast Informasi Perkuliahan Berbasis Android".Universitas Kristen Petra. Surabaya.
- [AYU-09] Ayuliana.2009. "Blackbox Testing".
<http://rifiana.staff.gunadarma.ac.id>. Diakses pada 27 Januari 2015.
- [COR-13] Janssen, Cory.2013"*Native Mobile App*".
<http://www.techopedia.com/definition/27568/native-mobile-app>.
Diakses pada 29 Desember 2014.
- [DID-09] Saputra, Didi Indra.2009."Bahasa Perograman Java".
<https://didiindra.wordpress.com/tag/pengertian-java/>. Diakses pada 28 Desember 2014.
- [DWI-13] Dwijaya, Ilman Fahma.2013."Sistem Pendukung Keputusan Kenaikan Jabatan pada PT Sysmex Menggunakan Metode *Profile Matching*".Universitas Komputer Indonesia.Bandung
- [HER-14] Dea,Kentia.2014."Notifikasi *Push* Android Menggunakan *Google Cloud Messaging*".
<http://www.hermantolle.com/Class/2014/11/notifikasi-push-android-menggunakan-google-cloud-messaging-gcm-php-dan-mysql/>. Diakses pada 28 Desember 2014.
- [IND-11] Indriyani.2011."Pengaruh Bimbingan Belajar Terhadap Prestasi Belajar Siswa".
<http://psikologi45.blogspot.com/2011/03/pengaruh-bimbingan-belajar-terhadap.html>. Diakses pada 27 Desember 2014.
- [JSO-13] Anonymous.2013."Pengenalan JSON". <http://www.json.org/json-id.html>. Diakses pada 28 Desember 2014.



- [KAS-12] Powersteering.2012. "Cara kerja aplikasi Android pada teknologi 'Cloud' Google (Android Junkies)".
<http://www.kaskus.co.id/thread/50cab9566012430853000053/cara-kerja-aplikasi-android-pada-teknologi-quotcloudquot-google-android-junkies>. Diakses pada 28 Desember 2014.
- [KUS-07] Kusrini.2007."Konsep dan Aplikasi Sistem Pendukung Keputusan". Andi.Yogyakarta.
- [MAR-13] Ardi, Mario.2013."Perbedaan *Native App* dan *Web App*".
<http://multimedia-portal.blogspot.com/2013/04/perbedaan-native-app-dengan-web-app.html>. Diakses pada 28 Desember 2014.
- [NIN-13] Pambayun, Kusumaning Hati dkk.2013." Sistem Pendukung Keputusan Seleksi Penerimaan Asisten Praktikum Menggunakan Metode Profil Matching (Studi Kasus Prodi Teknik Informatika Universitas Brawijaya)". Universitas Brawijaya.Malang.
- [NUR-10] Azizah, Nurul.2010."Apa itu Java?".
<http://menulisjava.blogspot.com/2010/12/berkenalan-dengan-java.html>. Diakses pada 28 Desember 2014.
- [PED-14] Anonymous.2014."Unified Modelling Language".
http://id.wikipedia.org/wiki/Unified_Modeling_Language. Diakses pada 28 Desember 2014.
- [SAP-13] Pandwinata, Vina.2013."Pengertian *Web service*".
<http://saptafina13.blogspot.com/2013/04/pengertian-web-service.html>. Diakses pada 28 Desember 2014.
- [SET-13] Setyaningsih, Wiji.2013."Decision Support Sistem Menggunakan Metode *Profile Matching*".
- [SRO-13] Yusro.2013."Pengertian JSON".
<http://www.mysuro.info/2013/01/pengertian-json-javascript-object.html>. Diakses pada 28 Desember 2014.

- [SUH-13] Dr. Suherman, M.Pd.2013."Bimbingan Belajar".Universitas Pendidikan Indonesia.
- [SUS-13] Marpaung, Susi.2013."Seberapa penting sih Pengujian Perangkat Lunak itu?". <http://susielvina.wordpress.com/2013/11/25/seberapa-penting-sih-pengujian-perangkat-lunak-itu/>. Diakses pada 4 Januari 2015.
- [TRA-13] Trav.2013."Android, MySQL, PHP, dan JSON 1".
<http://www.mybringback.com/tutorial-series/12924/android-tutorial-using-remote-databases-php-and-mysql-part-1/>. Diakses pada 28 Desember 2014.
- [YUL-13] Yulianti, Fariani Dewi, Sarwosari, Wijayanti.
2013."Pengembangan Aplikasi e-Learning Jurusan Teknik Informatika ITS Berbasis Android pada Perangkat Komunikasi Bergerak.Institut Teknologi Sepuluh Nopember.Surabaya.
- [ZAP-13] Gartias, Zapra.2013."Apa itu JSON? Mengenal JSON".
<http://www.tuxcorner.net/apa-itu-json-mengenal-json/>. Diakses pada 28 Desember 2014.

