



**RANCANG BANGUN SISTEM PENENTU JABATAN KARYAWAN
BERDASARKAN KEPERIBADIAN MENGGUNAKAN METODE *PROFILE***

MATCHING

**(STUDI KASUS: PT. PERKEBUNAN NUSANTARA XII KEBUN
KERTOWONO)**

SKRIPSI

Oleh

Antonius Halim Febrianto

NIM 132410101077

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2020



**RANCANG BANGUN SISTEM PENENTU JABATAN KARYAWAN
BERDASARKAN KEPERIBADIAN MENGGUNAKAN METODE *PROFILE***

MATCHING

**(STUDI KASUS: PT. PERKEBUNAN NUSANTARA XII KEBUN
KERTOWONO)**

SKRIPSI

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan pendidikan di Fakultas Ilmu Komputer Program Studi Sistem
Informasi Universitas Jember dan mendapat gelar Sarjana Komputer

Oleh

Antonius Halim Febrianto

NIM 132410101077

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2020

PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Allah SWT yang senantiasa memberikan rahmat dan hidayah-Nya untuk mempermudah dan melancarkan dalam mengerjakan skripsi.
2. Ayahanda Andre Halim Sugiartodan Ibunda tercinta Endang Lilik Asih.
3. Saudara laki-laki ku tersayang Aditya Dwi Sugiarto.
4. Keluarga PM atas dukungan beserta doanya.
5. Sahabatku – sahabatku yang telah memberi semangat.
6. Guru – guruku dari taman kanak-kanak sampai dengan perguruan tinggi.
7. Almamater Fakultas Ilmu Komputer Universitas Jember.

MOTO

“Stop Complaining, Just Do It”
(“Berhenti Mengeluh, Lakukan Saja”)



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Antonius Halim Febrianto

NIM : 132410101077

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Rancang Bangun Sistem Penentu Jabatan Karyawan Berdasarkan Kepribadian Menggunakan Metode *Profile Matching* (Studi Kasus: PT. Perkebunan Nusantara XII Kebun Kertowono)”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember,
Yang menyatakan,

Antonius Halim Febrianto
NIM 132410101077

SKRIPSI

**RANCANG BANGUN SISTEM PENENTU JABATAN KARYAWAN
BERDASARKAN KEPERIBADIAN MENGGUNAKAN METODE *PROFILE*
MATCHING
(STUDI KASUS: PT. PERKEBUNAN NUSANTARA XII KEBUN
KERTOWONO)**

Oleh :

Antonius Halim Febrianto

NIM 132410101077

Pembimbing

Dosen Pembimbing Utama : Drs. Antonius Cahya P., M. App.Sc, Ph. D

Dosen Pembimbing Pendamping : Oktalia Juwita, S. Kom., M.MT

PENGESAHAN PEMBIMBING

Skripsi berjudul “Rancang Bangun Sistem Penentu Jabatan Karyawan Berdasarkan Kepribadian Menggunakan Metode *Profile Matching* (Studi Kasus: PT. Perkebunan Nusantara XII Kebun Kertowono)”, telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Ilmu Komputer Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Drs. Antonius Cahya P., M. App.Sc, Ph. D Oktalia Juwita, S. Kom., M.MT
NIP. 196909281993021001 NIP 198110202014042001

PENGESAHAN PENGUJI

Skripsi berjudul “Rancang Bangun Sistem Penentu Jabatan Karyawan Berdasarkan Kepribadian Menggunakan Metode *Profile Matching* (Studi Kasus: PT. Perkebunan Nusantara XII Kebun Kertowono)”, telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Ilmu Komputer Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Yanuar Nurdiansyah, ST., M.Cs.

Gayatri Dwi Santika, S. SI., M.Kom

NIP. 198201012010121004

NIP. 760017013

Mengesahkan

Dekan Fakultas Ilmu Komputer

Prof. Dr. Saiful Bukhori, S.T., M. Kom

NIP. 196811131994121001

RINGKASAN

Rancang Bangun Sistem Penentu Jabatan Karyawan Berdasarkan Kepribadian Menggunakan Metode *Profile Matching* (Studi Kasus: PT. Perkebunan Nusantara XII Kebun Kertowono); Antonius Halim Febrianto, 132410101077; 2020, 125 HALAMAN; Fakultas Ilmu Komputer Program Studi Sistem Informasi Universitas Jember.

PTPN XII Kebun Kertowono merupakan salah satu anak perusahaan dari PTPN XII yang memproduksi dan mengekspor teh. Salah satu kendala yang dihadapi perusahaan PTPN XII Kebun Kerowono adalah ketidakmampuan karyawan dalam memaksimalkan kinerjanya. Hal tersebut dapat terjadi karena proses perekrutan karyawan yang kurang baik. Penulis menggunakan metode *Profile Matching* dalam melakukan analisa kecocokan kriteria jabatan PT. Perkebunan Nusantara XII Kebun Kertowono dengan kriteria kepribadian manusia.

Penelitian ini dilakukan dalam 3 tahap yaitu, tahap pengumpulan data, tahap analisis, dan tahap pengembangan sistem. Tahap pengumpulan data dilakukan melalui wawancara dengan pihak PT. Perkebunan Nusantara XII Kebun Kertowono dan Psikologi. Tahap analisis dilakukan dengan mencocokkan data kriteria jabatan dengan kepribadian manusia menggunakan metode *Profile Matching*. Tahap pengembangan dilakukan dengan membangun sebuah sistem penentu jabatan karyawan berdasarkan kepribadian manusia yang berbasis web. Hasil dari penelitian ini adalah kecocokan kriteria jabatan perusahaan PT. Perkebunan Nusantara XII Kebun Kertowono dengan kepribadian manusia.

PRAKATA

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Rancang Bangun Sistem Penentu Jabatan Karyawan Berdasarkan Kepribadian Menggunakan Metode *Profile Matching* (Studi Kasus: PT. Perkebunan Nusantara XII Kebun Kertowono)”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Fakultas Ilmu Komputer Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari dukungan berbagai pihak. Oleh karena itu penulis menyampaikan terima kasih kepada :

1. Prof. Dr. Saiful Bukhori, ST., M. Kom., selaku Dekan Fakultas Ilmu Komputer Universitas Jember;
2. Drs. Antonius Cahya P., M. App.Sc, Ph. D, selaku Dosen Pembimbing Utama dan Oktalia Juwita, S. Kom., M.MT., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
3. Prof. Drs. Slamin, M.Comp.Sc., Ph.D., sebagai dosen pembimbing akademik, yang telah mendampingi penulis sebagai mahasiswa;
4. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Fakultas Ilmu Komputer Universitas Jember;
5. Ayahanda tercinta Andre Halim Sugiarto dan Ibunda tercinta Endang Lilik Asih yang selalu mendukung dan mendoakan;
6. Saudara laki-laki ku Aditya Dwi Sugiarto;
7. Keluarga PM yang selalu memberi semangat serta doa;
8. PT. Perkebunan Nusantara XII Kebun Kertowono yang telah bersedia menjadi objek penelitian;
9. Teman-teman seperjuanganku Intention angkatan 2013;
10. Sahabatku – sahabatku yang telah memberi semangat;
11. Teman-Teman Program Studi Sistem Informasi di semua angkatan;

12. Semua pihak yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, oleh sebab itu penulis mengharapkan adanya masukan yang bersifat membangun dari semua pihak. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

Jember,

Penulis



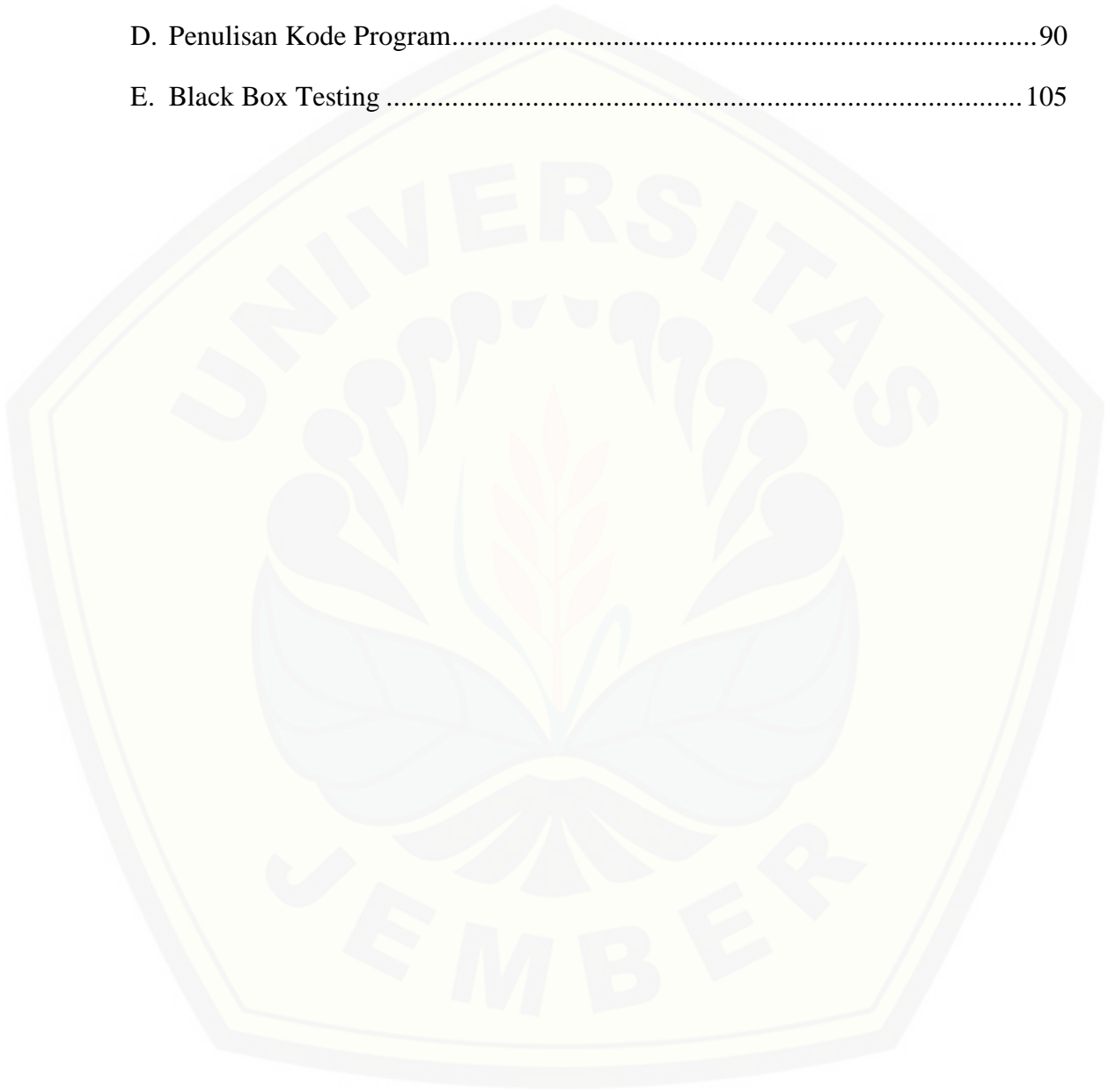
DAFTAR ISI

SKRIPSI.....	i
PERSEMBAHAN.....	ii
MOTO.....	iii
PERNYATAAN.....	iv
PENGESAHAN PEMBIMBING.....	vi
PENGESAHAN PENGUJI.....	vii
RINGKASAN.....	viii
PRAKATA.....	ix
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xv
DAFTAR GAMBAR.....	xvii
BAB 1. PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan.....	3
1.4 Manfaat.....	3
1.5 Batasan Masalah.....	4
1.6 Sistematika Penulisan.....	4
BAB 2. TINJAUAN PUSTAKA.....	6
2.1. Kepribadian.....	6
2.2. Spesifikasi Jabatan.....	6

2.3. PT. Perkebunan Nusantara XII Kebun Kertowono	7
2.4. Sistem Pendukung Keputusan	8
2.5. Metode <i>Profile Matching</i>	9
BAB 3. METODOLOGI PENELITIAN	13
3.1. Jenis Penelitian	13
3.2. Teknik Pengembangan Sistem	13
3.4.1. Analisis Kebutuhan	14
3.4.2. Desain Sistem	15
3.4.3. Implementasi	16
3.4.4. Pengujian	16
3.4.5. Pemeliharaan	17
BAB 4. ANALISIS PERANCANGAN SISTEM	18
4.1. Analisis Kebutuhan Data dan Sistem	18
4.1.1 Kebutuhan Fungsional	25
4.1.2 Kebutuhan Non-Fungsional	25
4.2. Desain Sistem	26
4.2.1 Usecase Diagram	26
4.2.2 <i>Usecase Scenario</i>	30
4.2.3 <i>Sequence Diagram</i>	37
4.2.4 <i>Activity Diagram</i>	41
4.2.5 <i>Class Diagram</i>	45
4.2.6 Entity Relationship Diagram (ERD)	45
4.3. Penulisan Kode Program dan Pengujian Sistem	46

4.3.1. Penulisan Kode Program.....	46
4.3.2. Pengujian Sistem	49
BAB 5. HASIL DAN PEMBAHASAN	50
5.1. Hasil Implementasi Kode Program Pada Sistem Penentu Jabatan Karyawan Di Suatu Perusahaan Berdasarkan Kepribadian Manusia.....	50
5.1.1. Tampilan Halaman Masuk	50
5.1.2. Tampilan Halaman Daftar.....	51
5.1.3. Tampilan Data Karyawan	51
5.1.4. Tampilan Halaman Data Kriteria Jabatan.....	52
5.1.5. Tampilan Halaman Bobot Kepribadian	53
5.1.6. Tampilan Halaman Hasil	53
5.1.7. Tampilan Halaman Profil Perusahaan	55
5.1.8. Tampilan Halaman Hasil	56
5.1.9. Tampilan Halaman Biodata.....	56
5.2. Implementasi Metode <i>Profile Matching</i>	57
5.3. Pembahasan Sistem Penentu Jabatan Karyawan Di Suatu Perusahaan Berdasarkan Kepribadian Manusia.....	61
BAB 6. PENUTUP	62
6.1. Kesimpulan	62
6.2. Saran	63
DAFTAR PUSTAKA	65
LAMPIRAN	67
A. Usecase Skenario	67

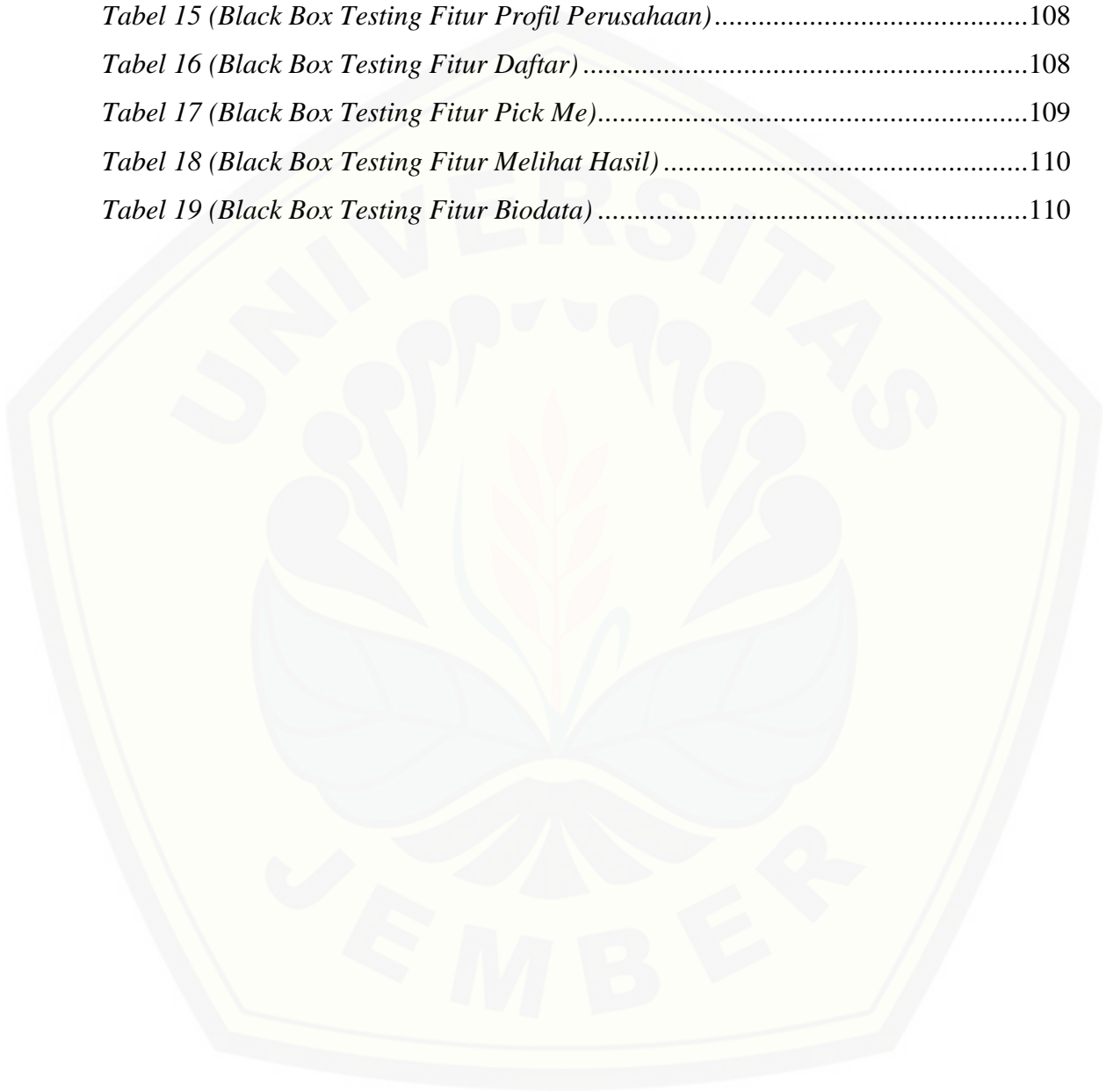
B. Sequence Diagram	79
C. Activity Diagram	83
D. Penulisan Kode Program.....	90
E. Black Box Testing	105



DAFTAR TABEL

Tabel 4. 1 Data Kriteria Jabatan	18
Tabel 4. 2 Subkriteria.....	19
Tabel 4. 3 Definisi Aktor	28
Tabel 4. 4 Definisi <i>Usecase</i>	29
Tabel 4. 5 Skenario <i>Usecase</i> Mengelola Biodata	33
Tabel 4. 6 Skenario <i>Usecase</i> Keluar	37
Tabel 5. 1 Kriteria Jabatan PTPN XII Kebun Kertowono	57
Tabel 5. 2 Kepribadian Sinta.....	58
Tabel 5. 3 Pembobotan Sinta.....	58
Tabel 5. 4 Selisih Kriteria Jabatan <i>Administrasi Keuangan</i>	59
Tabel 5. 5 Pembobotan <i>Gap</i> Sinta	59
<i>Tabel 1 Skenario Usecase Masuk (Admin)</i>	67
<i>Tabel 2 Skenario Usecase Masuk (Karyawan)</i>	68
<i>Tabel 3 (Skenario Usecase Melihat Data Karyawan)</i>	69
<i>Tabel 4 (Skenario Usecase Mengelola Data Kriteria Jabatan)</i>	70
<i>Tabel 5 (Skenario Usecase Melihat Hasil)</i>	73
<i>Tabel 6 (Skenario Usecase Melihat Bobot Kepribadian)</i>	74
<i>Tabel 7 (Skenario Usecase Melihat Profil Perusahaan)</i>	75
<i>Tabel 8 (Skenario Usecase Daftar)</i>	75
<i>Tabel 9 (Skenario Usecase Pick Me)</i>	77
<i>Tabel 10 (Skenario Usecase Melihat Hasil)</i>	78
<i>Tabel 11 (Black Box Testing Fitur Data Karyawan)</i>	105
<i>Tabel 12 (Black Box Testing Fitur Data Kriteria Jabatan)</i>	105

<i>Tabel 13 (Black Box Testing Fitur Hasil)</i>	107
<i>Tabel 14 (Black Box Testing Fitur Bobot Kepribadian)</i>	107
<i>Tabel 15 (Black Box Testing Fitur Profil Perusahaan)</i>	108
<i>Tabel 16 (Black Box Testing Fitur Daftar)</i>	108
<i>Tabel 17 (Black Box Testing Fitur Pick Me)</i>	109
<i>Tabel 18 (Black Box Testing Fitur Melihat Hasil)</i>	110
<i>Tabel 19 (Black Box Testing Fitur Biodata)</i>	110



DAFTAR GAMBAR

Gambar 3. 1 Model Waterfall (Sommerville, 2011).....	14
Gambar 4. 1 Diagram Alir Profile Matching	24
Gambar 4. 2 <i>Usecase</i> Diagram.....	27
Gambar 4. 3 <i>Activity</i> Diagram Mengelola Biodata.....	44
Gambar 4. 4 <i>Activity</i> Diagram Keluar.....	45
Gambar 4. 5 Entity Relationship Diagram.....	46
Gambar 5. 1 Tampilan Halaman Masuk	50
Gambar 5. 2 Tampilan Halaman Daftar.....	51
Gambar 5. 3 Tampilan Data Karyawan	52
Gambar 5. 4 Tampilan Halaman Data Kriteria Jabatan.....	52
Gambar 5. 5 Tampilan Halaman Bobot Kepribadian	53
Gambar 5. 6 Tampilan Halaman Hasil	54
Gambar 5. 7 Tampilan Haaman Profil Perusahaan	55
Gambar 5. 8 Tmpilan Halaman Hasil	56
Gambar 5. 9 Tampilan Halaman Biodata.....	57
<i>Gambar 1 (Sequence Diagram Melihat Data Karyawan)</i>	79
<i>Gambar 2 (Sequence Diagram Mengelola Data Kriteria Jabatan)</i>	80
<i>Gambar 3 (Sequence Diagram Melihat Hasil)</i>	80
<i>Gambar 4 (Sequence Diagram Melihat Bobot Kepribadian)</i>	81
<i>Gambar 5 (Sequence Diagram Melihat Profil Perusahaan)</i>	81
<i>Gambar 6 (Sequence Diagram Daftar)</i>	81
<i>Gambar 7 (Sequence Diagram Pick Me)</i>	82
<i>Gambar 8 (Sequence Diagram Melihat Hasil)</i>	82
<i>Gambar 9 (Sequence Diagram Mengelola Biodata)</i>	82
<i>Gambar 10 (Activity Diagram Masuk Admin)</i>	83

<i>Gambar 11 (Activity Diagram Masuk Karyawan)</i>	84
<i>Gambar 12 (Activity Diagram Melihat Data Karyawan)</i>	84
<i>Gambar 13 (Activity Diagram Mengelola Data Kriteria Jabatan)</i>	86
<i>Gambar 14 (Activity Diagram Melihat Hasil)</i>	86
<i>Gambar 15 (Activity Diagram Melihat Bobot Kepribadian)</i>	87
<i>Gambar 16 (Activity Diagram Melihat Profil Perusahaan)</i>	87
<i>Gambar 17 (Activity Diagram Daftar)</i>	88
<i>Gambar 18 (Activity Diagram Pick Me)</i>	89
<i>Gambar 19 (Activity Diagram Melihat Hasil)</i>	90
<i>Gambar 20 (Kode Program Masuk)</i>	91
<i>Gambar 21 (Kode Program Melihat Data Karyawan)</i>	92
<i>Gambar 22 (Kode Program Mengelola Data Kriteria Jabatan)</i>	93
<i>Gambar 23 (Kode Program Melihat Hasil)</i>	94
<i>Gambar 24 (Kode Program Melihat Bobot Kepribadian)</i>	95
<i>Gambar 25 (Kode Program Melihat Profil Perusahaan)</i>	97
<i>Gambar 26 (Kode Program Daftar)</i>	99
<i>Gambar 27 (Kode Program Pick Me)</i>	100
<i>Gambar 28 (Kode Program Melihat Hasil)</i>	101
<i>Gambar 29 (Kode Program Mengelola Biodata)</i>	104

BAB 1. PENDAHULUAN

Bab ini merupakan langkah awal dari penulisan tugas akhir. Bab ini berisi latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah, metodologi penelitian, dan sistematika penulisan.

1.1 Latar Belakang

Sumber Daya Manusia di dalam suatu perusahaan merupakan hal yang sangat penting untuk mendukung kemajuan dan kualitas perusahaan dalam mencapai tujuan. SDM adalah “aset perusahaan” yang harus dipelihara dan dikembangkan karena SDM faktor penentu keberhasilan aktivitas yang dilakukan dalam suatu perusahaan (Gouzali Saydam, 2000). Dalam hal ini, SDM menjadi salah satu alat pencapai target dan tujuan perusahaan. Perencanaan dan penelitian yang sempurna dalam sebuah perusahaan atau organisasi, bukan hal yang dapat menjamin tercapainya tujuan yang diharapkan, tanpa didukung oleh semangat dan kerja keras dari SDM yang tersedia (Asad D.M, 2003).

Kualitas SDM yang baik juga harus didukung dengan memiliki kepribadian yang baik. Permasalahan yang sering dihadapi yaitu ketidakmampuan suatu perusahaan menentukan apakah karyawan tersebut memiliki kepribadian seperti yang dibutuhkan atau tidak. Kepribadian adalah organisasi dinamis dalam individu, dari sistem psikofik yang menentukan penyesuaian yang unik terhadap lingkungannya (Feist J & Feist G, 2008). Pendapat lainnya dikemukakan oleh McCrae R bahwa kepribadian adalah karakteristik yang menetap dalam diri seseorang yang menggambarkan perilaku individu, yang digunakan untuk berinteraksi dengan lingkungannya (2011).

PTPN XII Kebun Kertowono merupakan salah satu perusahaan yang memproduksi dan mengekspor teh yang terdiri dari Ortodoks dan CTC. Salah satu kendala yang dihadapi perusahaan PT. Perkebunan Nusantara XII Kebun Kertowono adalah ketidakmampuan karyawan dalam memaksimalkan kinerjanya. Hal tersebut dapat terjadi karena proses perekrutan karyawan yang kurang baik.

Hidayat Arif Lukman dan Pinandita Tito pernah melakukan penelitian dengan mengimplementasikan Metode Gap Kompetensi (*Profile Matching*) pada Sistem Pendukung Keputusan Evaluasi Kinerja Karyawan Untuk Promosi Jabatan Struktural Pada Bimbingan Belajar Sciencemaster Menggunakan Metode Gap Kompetensi (*Profile Matching*) sebagai penentu kebijakan dari strategi dalam pengambilan keputusan penilaian kinerja karyawan, sehingga diharapkan karyawan dengan kemampuan terbaik yang akan terpilih untuk promosi jabatan struktural pada Bimbingan Belajar Sciencemaster (2013). Kriteria yang digunakan dalam sistem promosi jabatan berdasar pada 3 aspek yaitu aspek Kecerdasan, Sikap Kerja dan Perilaku. Hasil dari proses tersebut adalah ranking karyawan yang merupakan dasar rekomendasi bagi pengambilan keputusan untuk memilih karyawan yang cocok pada jabatan yang dibutuhkan.

Berdasarkan uraian diatas peneliti menggunakan metode *Profile Matching* karena metode ini dinilai mampu menghitung kecocokan kriteria jabatan dengan kepribadian manusia dengan menggunakan beberapa kriteria yang telah ditentukan. Secara garis besar merupakan proses membandingkan antara kriteria suatu jabatan ke dalam kriteria kepribadian manusia sehingga dapat diketahui perbedaan kriteria (disebut juga gap), semakin kecil gap yang dihasilkan maka bobot nilainya semakin besar.

Perancangan sistem informasi yang digunakan berbasis website. Penggunaan sistem informasi berbasis website dapat menjadi sebuah revolusi publikasi dalam membuka jangkauan informasi yang lebih luas untuk menyampaikan berbagai jenis informasi. Selain itu sistem informasi berbasis website memberikan kemudahan dalam aktivitas-aktivitas akademik tanpa batasan waktu, jarak dan tempat (Riyadi A.S, Retnandi E, & Deddy A, 2012).

1.2 Rumusan Masalah

Berdasarkan uraian yang telah disampaikan dalam latar belakang mendefinisikan permasalahan yang harus diselesaikan dalam penulisan ini, yaitu:

1. Bagaimana cara menentukan jabatan karyawan berdasarkan kepribadian manusia dengan menggunakan metode *Profile Matching*?
2. Bagaimana cara merancang dan membangun system penentu jabatan karyawan di suatu perusahaan berdasarkan kepribadian manusia menggunakan metode *Profile Matching*?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Menerapkan metode *Profile Matching* pada sistem penentu jabatan karyawan di suatu perusahaan berdasarkan kepribadian manusia.
2. Menentukan jabatan karyawan di suatu perusahaan berdasarkan kepribadian manusia.

1.4 Manfaat

Manfaat dari penelitian ini adalah:

1. Bagi Akademis
Penelitian yang dilakukan diharapkan memberikan hasil yang mampu memberikan masukan informasi yang terkait dengan judul penelitian kepada pembaca pada umumnya dan pada Fakultas Ilmu Komputer Universitas Jember pada khususnya.
2. Bagi Peneliti
Mengetahui bagaimana proses penerapan metode *Profile Matching* pada sistem penentu jabatan karyawan di suatu perusahaan berdasarkan kepribadian manusia.

3. Bagi Objek Penelitian

Mencocokkan kriteria jabatan PT. Perkebunan Nusantara XII Kebun Kertowono dengan kepribadian manusia.

1.5 Batasan Masalah

Penulis memberikan batasan masalah untuk objek dan tema yang dibahas sehingga tidak terjadi penyimpangan dalam proses penulisan dan pembuatan aplikasi.

Berikut adalah batasan masalah yang dicantumkan:

1. Sistem yang dibangun berbasis website
2. Jabatan yang ditentukan yaitu Pemasaran, Administrasi Keuangan dan Human Resource Departement (HRD)

1.6 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut:

1. Pendahuluan

Bab kesatu ini memuat uraian tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan skripsi yang masing-masing tertuang secara eksplisit dalam subbab tersendiri.

2. Tinjauan Pustaka

Bab ini memaparkan tinjauan terhadap hasil-hasil penelitian terdahulu berkaitan dengan masalah yang dibahas, landasan materi, dan kajian teori metode analisis data yang berkaitan dengan masalah dalam penelitian.

3. Metodologi Penelitian

Bab ini menguraikan tentang tempat dan waktu penelitian, metode penelitian, metode pengumpulan data, metode analisis data, dan teknik pengembangan sistem yang digunakan dalam penelitian.

4. Analisis dan Perancangan Sistem

Bab ini berisi uraian tentang tentang perancangan desain sistem. Perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, kemudian merancang *usecase diagram*, *scenario*, *activity diagram*, *sequence diagram*, *class diagram* dan *entity relationship diagram (ERD)*.

5. Hasil dan Pembahasan

Bab ini memaparkan secara rinci pemecahan masalah melalui analisis yang disajikan dalam bentuk deskripsi dibantu dengan ilustrasi berupa tabel dan gambar untuk memperjelas hasil penelitian.

6. Penutup

Bab ini terdiri atas kesimpulan atas penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.

BAB 2. TINJAUAN PUSTAKA

Pada bagian ini dipaparkan tinjauan yang berkaitan dengan masalah yang dibahas, kajian teori yang berkaitan dengan masalah serta kajian teori yang dikaitkan dengan permasalahan yang dihadapi.

2.1. Kepribadian

Kepribadian adalah karakteristik seseorang yang menyebabkan munculnya konsistensi perasaan, pemikiran, dan perilaku (Pervin Lawrence A dkk, 2010). Penjelasan mengenai kepribadian dapat kita temukan melalui beberapa ahli Psikologi di dunia. Salah satu pengertian yang dikemukakan, kepribadian merupakan pola sifat yang relatif permanen dan mempunyai karakteristik yang unik yang secara konsisten mempengaruhi perilakunya (Feist J & Feist G.J, 2006).

Menurut Allport menjelaskan bahwa kepribadian merupakan organisasi dinamis dalam individu, dari sistem psikofisik yang menentukan penyesuaian yang unik terhadap lingkungannya (Feist J & Feist G, 2008). McCrae R yang berpendapat bahwa kepribadian adalah karakteristik yang menetap dalam diri seseorang yang menggambarkan perilaku individu, yang digunakan untuk berinteraksi dengan lingkungannya (2011). Penelitian yang dilakukan Robbins dan Judge mendefinisikan kepribadian sebagai jumlah total dari cara seseorang untuk bereaksi dan berinteraksi dengan orang lain (2011).

2.2. Spesifikasi Jabatan

Spesifikasi jabatan (job specification) menunjukkan siapa yang melakukan pekerjaan itu dan faktor-faktor manusia yang diisyaratkan (Handoko, T., Hani, 1998). Spesifikasi jabatan juga merupakan persyaratan minimal yang harus dipenuhi oleh orang yang menduduki suatu jabatan, agar ia dapat melaksanakan tugas yang dibebankan kepadanya dengan baik dan bertanggung jawab. Menurut Alex S.

Nitisemito (1980) Job Spesification (Spesifikasi jabatan) merupakan suatu informasi tentang syarat – syarat yang diperlukan bagi setiap karyawan agar dapat memangku suatu jabatan dengan baik. Oleh sebab itu, perusahaan harus bisa menempatkan karyawan sesuai dengan minat dan kemampuannya serta memperhatikan kebutuhannya.

Selain itu, dengan menempatkan karyawan pada suatu jabatan yang sesuai dengan kualifikasi yang dimilikinya berarti para karyawan telah diberi kesempatan untuk mengembangkan diri dan merealisasikan potensinya seoptimal mungkin. Di dalam spesifikasi jabatan ditentukan kemampuan dan bakat dasar yang harus dimiliki untuk menjalankan pekerjaan.

2.3. PT. Perkebunan Nusantara XII Kebun Kertowono

PTPN XII Kebun Kertowono ini berdiri sejak tahun 1910 yang memiliki luas 2267,97 hektare. Dengan komoditas utama perusahaan ini yaitu berupa teh. Kebun Teh Kertowono mempunyai kepekatan dan the berwarna merah bata. Perusahaan ini juga memiliki alat mesin dari penggilingan hingga sortasi, dan memiliki beberapa bagian pengelolaan mulai dari penerimaan, pelayuan, penggilingan, pengeringan, sortasi, hingga penyimpanan.

Sebagai sebuah perusahaan, PTPN XII Kebun Kertowono juga memiliki Visi dan Misi tersendiri yaitu:

Visi PTPN XII Kebun Kertowono

“Menjadi Perusahaan Agribisnis yang berdaya saing tinggi dan mampu tumbuh-kembang berkelanjutan”

Misi PTPN XII Kebun Kertowono:

1. Melaksanakan reformasi bisnis, strategi, struktur, dan budaya perusahaan untuk mewujudkan profesionalisme berdasarkan prinsip-prinsip good corporate governance.

2. Meningkatkan nilai dan daya saing perusahaan (competitive advantage) melalui inovasi serta peningkatan produktivitas dan efisiensi dalam penyediaan produk berkualitas dengan harga kompetitif dan pelayanan bermutu tinggi.
3. Menghasilkan laba yang dapat membawa perusahaan tumbuh dan berkembang untuk meningkatkan nilai bagi shareholders dan stakeholders lainnya.
4. Mengembangkan usaha agribisnis dengan tata kelola yang baik serta peduli pada kelestarian alam dan tanggung jawab sosial pada lingkungan usaha (community development).

2.4. Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) adalah sistem yang dapat menyelesaikan masalah dengan efektif dan efisien. SPK membantu untuk mengambil suatu keputusan dari berbagai alternatif hasil pengolahan suatu informasi yang diperoleh dari suatu model pengambilan keputusan (Marsono, Boy A.F, & Dari W, 2015).

SPK digunakan dalam pengambilan keputusan pada keadaan semiterstruktur dan tidak terstruktur. Adapun karakteristik dalam SPK (Saragih M.A, 2015), yaitu:

1. Menunjang pembuatan keputusan dalam menangani masalah semi terstruktur dan tidak terstruktur
2. Membantu dari berbagai tingkat manajemen, mulai dari tingkat atas sampai tingkat bawah.
3. Menunjang pembuatan keputusan secara kelompok maupun perorangan, pembuatan keputusan yang saling bergantung dan berurutan.
4. Menunjang berbagai bentuk proses pembuatan keputusan dan jenis keputusan.
5. Mudah dikembangkan oleh pemakai akhir serta memiliki kemampuan pemodelan dan analisis data.

SPK dapat digambarkan sebagai sistem yang berkemampuan mendukung analisis data, pemodelan keputusan, berorientasi keputusan, orientasi perencanaan masa depan yang digunakan pada saat-saat yang tidak biasa. SPK juga merupakan penggabungan sumber-sumber kecerdasan individu dengan kemampuan komponen

untuk memperbaiki kualitas keputusan dan menjadi sistem informasi berbasis komputer untuk manajemen pengambilan keputusan yang menangani masalah-masalah semi

Secara umum dapat dikatakan bahwa kegunaan dari SPK adalah untuk meningkatkan kemampuan para pengambil keputusan dengan memberikan alternatif – alternatif keputusan yang lebih banyak atau lebih baik, sehingga dapat membantu untuk merumuskan masalah dan keadaan yang dihadapi. Dengan demikian SPK dapat menghemat waktu, tenaga dan biaya. Sehingga dapat dikatakan secara singkat bahwa tujuan Sistem Pendukung Keputusan adalah untuk meningkatkan efektivitas dan efisiensi dalam pengambilan keputusan.

2.5. Metode *Profile Matching*

Metode *Profile Matching* merupakan sebuah mekanisme pengambilan keputusan dengan mengasumsikan bahwa terdapat variabel predictor yang ideal yang harus dimiliki, bukannya tingkat minimal yang harus dipenuhi atau dilewati. Metode *Profile Matching* atau pencocokan profil adalah metode yang sering digunakan sebagai mekanisme dalam pengambilan keputusan dengan mengasumsikan bahwa terdapat tingkat variabel prediktor yang ideal yang harus dipenuhi oleh subyek yang diteliti (Hidayat A.L & Pinandita T, 2013).

Penelitian yang dilakukan oleh Hidayat dan Pinandita menggunakan metode *Profile Matching* untuk menentukan kebijakan strategi dalam pengambilan keputusan penilaian kinerja karyawan (2013). Kriteria yang digunakan berdasar pada 3 aspek yaitu aspek Kecerdasan, Sikap Kerja dan Perilaku. Berdasarkan penelitian yang telah dilakukan, penulis mengambil dan mengimplementasikan perhitungan dari metode *Profile Matching* untuk mencocokkan kriteria jabatan PT. Perkebunan Nusantara XII Kebun Kertowono dengan kepribadian manusia.

Metode *Profile Matching* memiliki keunggulan dalam sistem analisisnya diantaranya: mempertimbangkan konsistensi yang logis dalam penilaian yang

digunakan untuk menentukan prioritas sehingga menghasilkan alternatif yang tidak banyak, serta pada metode ini dilakukan proses pembobotan dan sehingga dapat diketahui perbedaan kompetensinya (disebut juga *gap*), semakin kecil *gap* yang dihasilkan maka bobot nilainya semakin besar yang berarti memiliki peluang yang lebih besar juga. Selain itu metode *profile matching* juga memiliki kelemahan yaitu, tidak memperhitungkan daya tahan ketahanan output analisis sensitivitas pengambilan keputusan.

Proses metode *Profile Matching* secara garis besar merupakan proses membandingkan antara nilai data actual dari suatu *profile* yang akan dinilai dengan nilai *profile* yang diharapkan yang disebut *gap*. Gap yang dimaksud adalah perbedaan / selisih value masing – masing aspek / atribut dengan value target. Untuk pengumpulan *gap – gap* yang terjadi itu sendiri pada tiap aspeknya mempunyai perhitungan yang berbeda – beda. Pemetaan Gap Kompetensi dapat dirumuskan pada Persamaan 2.1.

$$\text{Gap} = \text{Value Atribut} - \text{Value Target} \quad \dots\dots (2.1)$$

1. Pembobotan

Setelah diperoleh Gap pada masing – masing pembeli, setiap profil kriteria kualitas pembeli diberi bobot nilai sesuai dengan ketentuan yang sudah ditetapkan. Tabel bobot nilai *gap* dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Bobot Nilai *Gap*

Selisih	Bobot Nilai	Keterangan
0	5	Tidak ada selisih (Kompetensi sesuai yang dibutuhkan)
1	4.5	Kompetensi <i>profile</i> kelebihan 1 tingkat/level
-1	4	Kompetensi <i>profile</i> kekurangan 1 tingkat/level
2	3.5	Kompetensi <i>profile</i> kelebihan 2 tingkat/level
-2	3	Kompetensi <i>profile</i> kekurangan 2 tingkat/level
3	2.5	Kompetensi <i>profile</i> kelebihan 3 tingkat/level
-3	2	Kompetensi <i>profile</i> kekurangan 3 tingkat/level
4	1.5	Kompetensi <i>profile</i> kelebihan 4 tingkat/level
-4	1	Kompetensi <i>profile</i> kekurangan 4 tingkat/level

Sumber: (Kusrini, 2007)

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

Setelah menentukan bobot nilai *gap* untuk semua aspek dengan cara yang sama, setiap aspek dibagi lagi menjadi dua kelompok *Core Factor* (faktor utama) dan *Secondary Factor* (faktor pendukung). Perhitungan *core factor* dapat ditunjukkan pada Persamaan 2.2.

$$NCF = \frac{\sum NC}{\sum IC} \dots\dots (2.2)$$

Keterangan:

NCF : nilai rata-rata core factor

NC : jumlah total nilai core factor

IC : jumlah item core factor

Sedangkan untuk perhitungan *secondary factor* dapat ditunjukkan pada Persamaan 2.3.

$$NSF = \frac{\sum NS}{\sum IS} \quad \dots\dots (2.3)$$

Keterangan:

NSF : nilai rata rata *secondary factor*

NS : jumlah total nilai *secondary factor*

IS : jumlah item *secondary factor*

3. Penghitungan Nilai Total

Dari hasil perhitungan dari tiap aspek di atas kemudian dihitung nilai total berdasar prosentase dari *core* dan *secondary factor* yang diperkirakan berpengaruh terhadap kinerja tiap-tiap *profil*. Penghitungan nilai total dapat dilihat dalam Persamaan 2.4.

$$N = (X)\% NCF + (X)\% NSF \quad \dots\dots (2.4)$$

Keterangan:

N : nilai total tiap aspek

NCF : nilai rata rata *core factor*

NSF : nilai rata rata *secondary factor*

(X)% : nilai presentase yang diinputkan

BAB 3. METODOLOGI PENELITIAN

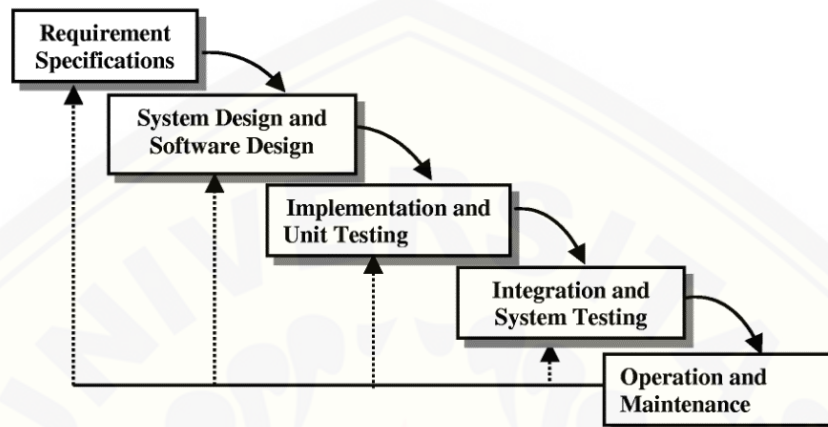
Bab ini menggambarkan tentang penelitian yang akan dilakukan untuk menjawab rumusan masalah sehingga dapat mewujudkan tujuan sebenarnya dari penelitian. Pada metodologi penelitian akan dijelaskan tentang jenis penelitian, alur penelitian dan tahapan pengembangan sistem.

3.1. Jenis Penelitian

Jenis penelitian yang akan dilakukan merupakan penelitian kualitatif. Penelitian kualitatif yang akan dilakukan adalah melakukan wawancara kepada semua pihak yang berkaitan dengan adanya sistem penentu jabatan perusahaan berdasarkan kepribadian manusia menggunakan metode *Profile Matching*.

3.2. Teknik Pengembangan Sistem

Untuk membangun sebuah sistem dibutuhkan sebuah model dalam pengembangannya. Dalam penelitian ini model yang digunakan adalah model waterfall. Model ini dipakai karena umum digunakan dalam pengembangan *software* berskala kecil. Lima tahap dalam model waterfall adalah sebagai berikut, yaitu *Requirement Definition, System and Software Design, Implementation and UnitTesting, Integration and System Testing, Operations and Maintenance*. Model waterfall dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Model Waterfall (Sommerville I, 2011)

3.4.1. Analisis Kebutuhan

Tahap pertama yang harus dilakukan adalah proses analisis kebutuhan terhadap sistem yang akan dirancang. Data kebutuhan dibagi menjadi kebutuhan fungsional dan non-fungsional. Pada sistem ini dibutuhkan juga data-data kriteria yang akan digunakan sebagai pendukung keputusan. Data yang sudah terkumpul akan menentukan bagaimana fitur yang akan dibangun pada sistem. Teknik analisis kebutuhan yang akan dilakukan pada penelitian ini adalah sebagai berikut:

1. Studi pustaka

Studi pustaka merupakan teknik pengumpulan data dengan mengadakan studi penelaahan terhadap buku-buku, literatur-literatur, catatan-catatan, karya ilmiah, dan situs web yang ada hubungannya dengan masalah yang dipecahkan. Bertujuan untuk menyusun dasar teori yang akan digunakan dalam penelitian.

2. Wawancara

Wawancara merupakan cara pengumpulan data yang berhubungan langsung dengan narasumber dengan menyajikan pertanyaan kepada narasumber. Narasumber pada penelitian ini adalah Ibu Tina HRD PT. Perkebunan Nusantara XII Kebun Kertowono dan Ibu Fitria yang berprofesi sebagai psikolog. Wawancara terhadap narasumber bertujuan untuk memperoleh data yang dibutuhkan dalam penyelesaian penelitian. Data yang diperoleh dari hasil wawancara adalah data jenis – jenis jabatan dan kepribadian manusia.

3.4.2. Desain Sistem

Tahapan selanjutnya adalah desain sistem dengan menggunakan Unified Modeling Language (UML) yang dirancang dengan konsep Object-Oriented Programming (OOP). Pemodelan UML yang digunakan sebagai berikut:

1. *Usecase Diagram*

Use Case Diagram bertujuan untuk mengetahui fitur-fitur yang akan dibuat pada sistem.

2. *Skenario*

Skenario bertujuan untuk mengetahui alur kerja dari aksi aktor dan reaksi sistem masing-masing fitur yang ada pada *Use Case Diagram*.

3. *Sequence Diagram*

Sequence Diagram bertujuan untuk menggambarkan urutan proses dari function yang dibuat pada saat pengkodean.

4. *Activity Diagram*

Activity diagram bertujuan untuk memberikan gambaran jalannya sistem gar lebih mudah dibaca oleh pengguna.

5. *Class Diagram*

Class Diagram bertujuan untuk melihat relasi antar *class* yang digunakan dalam sistem.

6. *Entity Relationship Diagram*

Entity Relationship Diagram merupakan diagram yang bertujuan untuk melihat relasi antar data yang ada pada *database*.

3.4.3. Implementasi

Tahap implementasi merupakan tahapan untuk mengimplementasikan desain yang telah dirancang ke dalam kode program. Dengan melakukan penulisan kode program menggunakan bahasa pemrograman *php*. Untuk manajemen basis data menggunakan DBMS MySQL.

3.4.4. Pengujian

Tahap pengujian yang dilakukan oleh penulis dilakukan dengan 2 cara yaitu pengujian sistem dan pengujian algoritma. Tahap pengujian untuk mengetahui kinerja sistem penulis menggunakan *whitebox testing* dan *blackbox testing* dan pada tahap pengujian algoritma penulis membandingkan pengujian manual dengan pengujian pada sistem.

1. Pengujian Sistem

a. *Black Box Testing*

BlackBox Testing digunakan untuk menguji spesifikasi fungsionalitas dari suatu aplikasi atau *software*. Seorang *tester* dengan melakukan proses *black box testing* dapat mendefinisikan kumpulan kondisi input dan fungsionalitas suatu aplikasi ataupun *software*. *Black box testing* merupakan pendukung dan pelengkap yang digunakan untuk menguji hal – hal yang tidak tercakup dalam *White box testing* (Mustaqbal M.S, Firdaus R.F, & Rahmadi H, 2015). Pada pengujian *black box* ini, aplikasi yang dibangun pada penelitian ini akan diuji dengan mengujikan langsung running aplikasi dan melakukan kegiatan pengujian dengan menganalisis proses input dan output yang dihasilkan aplikasi.

3.4.5. Pemeliharaan

Pemeliharaan sistem diperlukan ketika sistem telah digunakan dan dijalankan oleh *user*. Ketika sistem dijalankan mungkin saja masih terjadi kesalahan atau *error* yang tidak ditemukan sebelumnya. Sehingga diperlukan perbaikan pada sistem tersebut.



BAB 4. ANALISIS PERANCANGAN SISTEM

Bab ini akan membahas tentang analisis dan perancangan Sistem Penentu Jabatan Karyawan Berdasarkan Kepribadian Menggunakan Metode *Profile Matching*. Tahap perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, pembuatan desain sistem, penulisan kode program, dan pengujian sistem.

4.1. Analisis Kebutuhan Data dan Sistem

Proses analisis kebutuhan sistem dilakukan dengan melakukan wawancara pada pihak PT. Perkebunan Nusantara XII Kebun Kertowono dan psikolog untuk mendapatkan data kriteria jabatan dan kepribadian manusia. Berikut merupakan data kriteria dan kompetensi yang didapatkan melalui proses pencarian data. Data kriteria jabatan dapat dilihat pada Tabel 4.1

Tabel 4. 1 Data Kriteria Jabatan

No	Data	Keterangan
1	Jabatan	Data Atribut
2	Pikiran	Parameter
3	Energi	Parameter
4	Sifat	Parameter
5	Strategi	Parameter
6	Identitas	Parameter

Penulis juga menggunakan subkriteria pada kriteria jabatan. Pembuatan subkriteria bertujuan untuk mempermudah proses pembobotan. Kriteria ditunjukkan pada Tabel 4.2.

Tabel 4. 2 Kriteria

No	Aspek	Kriteria	Subkriteria	Bobot
1	Pikiran	Ekstrovert	Saya menyukai aktivitas di luar ruangan	5
			Saya menyukai pekerjaan yang berhubungan dengan lingkungan di luar kerja saya	
			Saya mudah bergaul di lingkungan baru	
		Introvert	Tulisan adalah cara saya mengekspresikan diri	4
			Pekerjaan yang tidak melibatkan banyak orang lebih saya gemari	
			Beraktivitas sendirian lebih saya senangi	

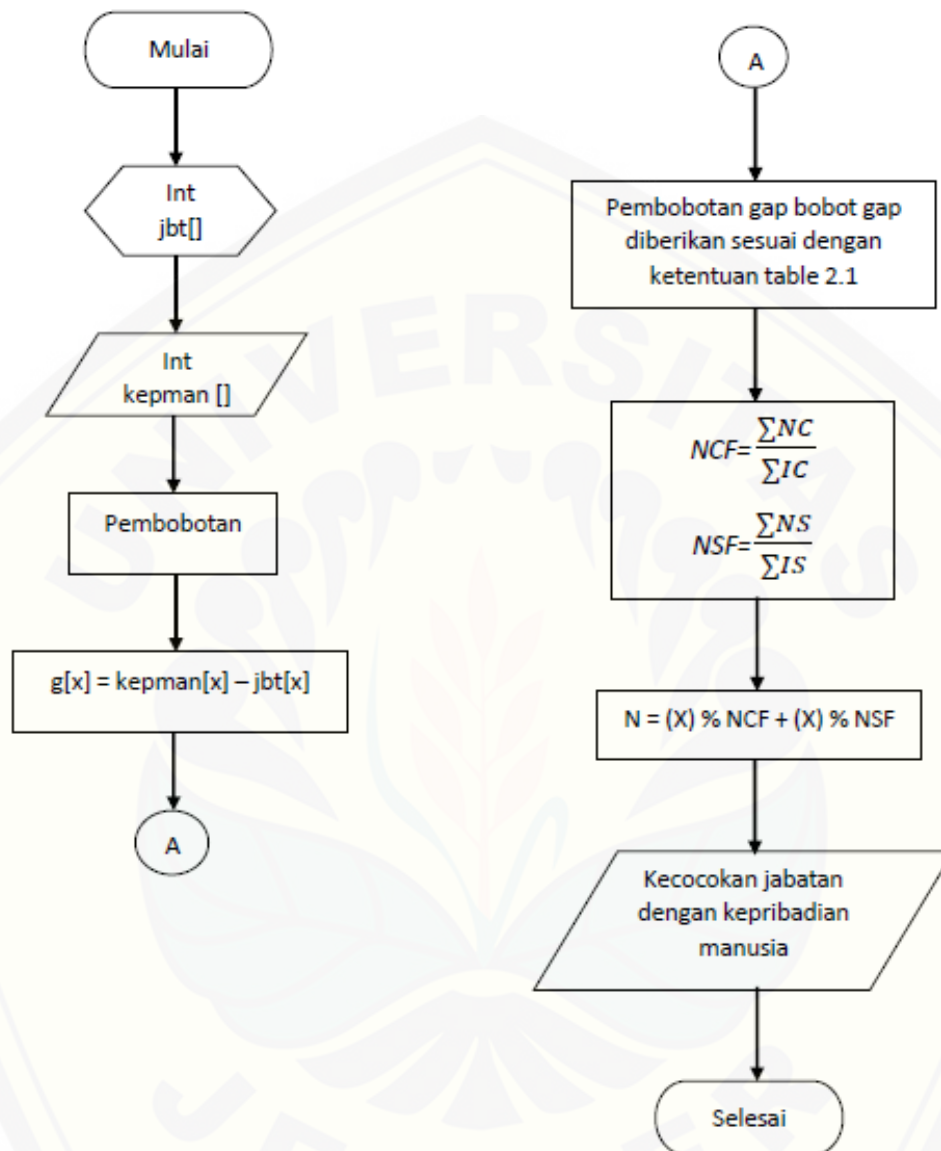
2	Energi	Visioner	Saya senang mengeksplere ide dan rencana	2
			Saya cenderung mendalami pikiran saya dan mengabaikan sekitar	
			Sesuatu yang tidak realistis dan menarik lebih saya senangi	
		Realistik	Saya orang yang praktis di banding kreatif	3
Saya jarang terhanyut dengan oleh fantasi dan ide				
Pengalaman lebih saya utamakan di banding imajinasi				
3	Sifat	Rasional	Pengambilan keputusan harus	2

			secara logika bukan perasaan	
			Melakukan yang benar lebih penting di banding harus kooperatif	
			Saya tidak khawatir terhadap pikiran orang lain	
		Emosional	Pikiran orang lain penting untuk saya renungkan	1
			Loyalitas lebih utama bagi saya	
			Saya lebih sering memberikan dukungan emosioanal di banding solusi	
4	Strategi	Perencanaan	Saya menyukai hal-hal yang terjadwal	3

			Saya menyukai lingkungan kerja yang rapih	
			Saya terbiasa dengan keteraturan	
		Improvisasi	Bekerja secara dinamis lebih saya senangi dibandingkan terorganisir	2
			Saya tidak terbiasa menghabiskan waktu membuat rencana	
			Membuat opsi lain lebih penting buat saya	
5	Identitas	Tegas	Pikiran orang lain tidak mempengaruhi kinerja saya	1
			Saya sulit melakukan hal diluar yang	

			sudah disepakati bersama	
			Saya cenderung bekerja dengan ketentuan yang ada	
		Hati-Hati	Saya bukan orang yang cepat bertindak	2
			Saya selalu melihat dari semua kondisi	
			Melihat hasil kerja kembali penting bagi saya	

Data kriteria dan kompetensi diatas selanjutnya digunakan dalam proses analisis kebutuhan sistem dengan menghitung data yang telah disimpan ke dalam database menggunakan metode *Profile Matching*. Diagram alir metode *profile matching* dapat dilihat pada Gambar 4.1



Gambar 4. 1 Diagram Alir Profile Matching

Keterangan:

- jbt[] = variabel penampung nilai kriteria jabatan
 kepman[] = variabel penampung nilai kriteria kepribadian manusia
 g[] = nilai gap masing – masing kriteria

NCF	= hasil perhitungan nilai <i>core factor</i>
NSF	= hasil perhitungan nilai <i>secondary factor</i>
N[]	= nilai total
\sum NC	= total nilai gap yang masuk <i>core factor</i>
\sum IC	= jumlah item <i>core factor</i>
\sum NS	= total nilai gap yang masuk <i>secondary factor</i>
\sum IS	= jumlah item <i>secondary factor</i>

4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional berisi proses – proses yang akan dilakukan oleh sistem. Berdasarkan hasil wawancara dengan pihak terkait, kebutuhan fungsional dari sistem ini adalah sebagai berikut:

1. Sistem mampu melihat data karyawan
2. Sistem mampu melihat data kriteria jabatan
3. Sistem mampu mengelola data kriteria jabatan (*insert, edit*)
4. Sistem mampu melihat bobot kepribadian
5. Sistem mampu melihat hasil
6. Sistem mampu melihat profil perusahaan
7. Sistem mampu melakukan daftar
8. Sistem mampu melakukan tes
9. Sistem mampu melihat hasil
10. Sistem mampu mengelola biodata (*view, edit*)

4.1.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan hal yang dibutuhkan oleh sistem untuk mendukung aktivitas sistem sesuai dengan kebutuhan fungsional yang telah disusun. Kebutuhan non-fungsional menitikberatkan pada properti perilaku yang dimiliki oleh sistem. Kebutuhan non-fungsional sistem ini yaitu:

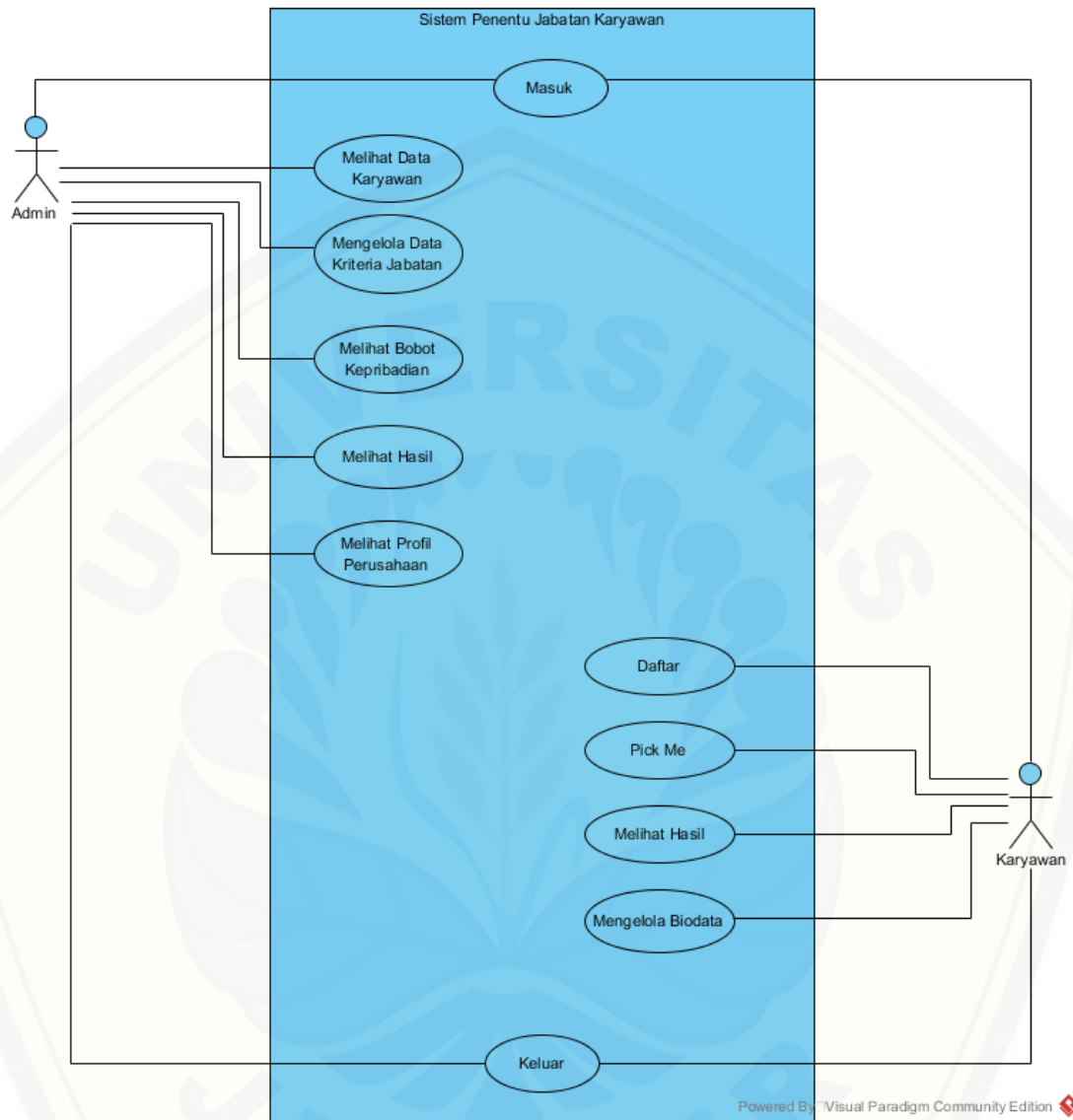
1. Sistem menggunakan *username* dan *password* untuk autentifikasi akses terhadap sistem.
2. Sistem Penentu Jabatan Karyawan Di Suatu Perusahaan Berdasarkan Kerribadian Manusia menggunakan *template* dari *bootstrap* sehingga tampilan web *user friendly* dan *reziseable*.

4.2. Desain Sistem

Tahap desain merupakan tahapan dalam pembuatan desain sistem yang dapat memudahkan developer dalam membangun sistem. Perancangan sistem dengan konsep perancangan berorientasi objek, dengan menggunakan UML (*Unified Modeling Language*) yang digunakan antara lain:

4.2.1 Usecase Diagram

Usecase diagram merupakan diagram yang menggambarkan interaksi antara sistem dan aktor sistem. *Usecase diagram* hanya memberi gambaran singkat hubungan antara *usecase*, aktor, dan sistem. Dari *usecase diagram* ini dapat diketahui fitur-fitur apa saja yang dimiliki oleh sistem yang dibuat. *Usecase diagram* aplikasi ini dapat dilihat pada Gambar 4.2.



Gambar 4. 2 Usecase Diagram

Berikut penjelasan tentang definisi aktor dan definisi *Usecase* dalam *Usecase diagram* pada Gambar 4.2.

1. Definisi Aktor

Definisi aktor merupakan penjelasan tentang aktor – aktor sebagai pengguna dari Sistem Penentu Jabatan Karyawan Di Suatu Perusahaan Berdasarkan Kepribadian Manusia yang akan dibangun. Terdapat 2 aktor seperti yang dijelaskan pada Tabel 4.3.

Tabel 4. 3 Definisi Aktor

No.	Aktor	Deskripsi
1.	Admin	Aktor admin memiliki hak akses secara penuh. Admin dapat melakukan <i>login</i> dan mengelola data sistem secara keseluruhan, meliputi data kriteria jabatan, data karyawan, data kepribadian dan data bobot kepribadian.
2.	Karyawan	Aktor karyawan memiliki hak akses untuk mengelola data biodata, melakukan tes dan melihat hasil perhitungan.

2. Definisi *Usecase*

Definisi *Usecase* merupakan penjelasan dari masing-masing *Usecase* atau fitur-fitur dari Sistem Penentu Jabatan Karyawan Di Suatu Perusahaan Berdasarkan Kepribadian Manusia yang terdapat pada Gambar 4.2. Definisi *Usecase* digambarkan pada Tabel 4.4.

Tabel 4. 4 Definisi *Usecase*

No.	No <i>Usecase</i>	<i>Usecase</i>	Deskripsi
1.	U001	Masuk	Menggambarkan proses autentifikasi untuk masuk ke sistem.
2.	U002	Melihat Data Karyawan	Menggambarkan proses melihat data karyawan
3.	U003	Mengelola Data Kriteria Jabatan	Menggambarkan proses melihat, menambah, mengubah, data kriteria jabatan
4.	U004	Melihat Hasil	Menggambarkan proses melihat hasil
5.	U005	Melihat Bobot Kepribadian	Menggambarkan proses melihat bobot kepribadian
6.	U006	Melihat Profil Perusahaan	Melihat Profil
7.	U007	Daftar	Menggambarkan proses pendaftaran karyawan
8.	U008	Pick Me	Menggambarkan proses penentuan jabatan berdasarkan kepribadian manusia
9.	U009	Melihat Hasil	Melihat hasil
10.	U010	Mengelola Biodata	Menggambarkan proses melihat, mengubah dan mencetak biodata karyawan
11.	U011	Keluar	Menggambarkan proses keluar dari sistem

4.2.2 *Usecase Scenario*

Usecase scenario digunakan untuk menjelaskan alur sistem sesuai dengan *Usecase diagram* pada Gambar 4.2.

4.2.2.1 Skenario *Usecase* Masuk

Skenario *usecase* masuk merupakan alur yang menjelaskan dari proses masuk ke dalam sistem serta alur alternatif yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* masuk adalah *admin* dan karyawan. Prekondisi dari skenario *usecase* masuk adalah aktor harus memiliki *username* dan *password* untuk dapat mengakses sistem. Prakondisi dari skenario *usecase* masuk adalah aktor berhasil masuk ke dalam system. Alur alternatif yang ada pada skenario ini adalah *username* atau *password* yang diinputkan aktor salah dan *field* pada *form* masuk kosong. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sistem pada skenario *usecase* masuk dapat dilihat pada lampiran A Tabel 1 dan Tabel 2.

4.2.2.2 Skenario *Usecase* Melihat Data Karyawan

Skenario *usecase* melihat data karyawan merupakan alur yang menjelaskan aktor pada saat melihat data karyawan yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* melihat data karyawan adalah *admin*. Prekondisi dari skenario *usecase* melihat data karyawan adalah aktor harus berhasil masuk terlebih dahulu untuk dapat melihat data karyawan. Prakondisi dari skenario *usecase* melihat data karyawan adalah data karyawan berhasil dilihat. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sitem pada skenario *usecase* mengelola data karyawan dapat dilihat pada lampiran A Tabel 3.

4.2.2.3 Skenario *Usecase* Mengelola Data Kriteria Jabatan

Skenario *usecase* mengelola data kriteria jabatan merupakan alur yang menjelaskan aktor pada saat melihat, menambah, mengubah data kriteria jabatan serta alur alternatif yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* mengelola data kriteria jabatan adalah *admin*.

Prekondisi dari skenario *usecase* mengelola data kriteria jabatan adalah aktor harus berhasil masuk terlebih dahulu untuk dapat mengelola data kriteria jabatan. Prakondisi dari skenario *usecase* mengelola data kriteria jabatan adalah data kriteria jabatan berhasil ditambahkan, data kriteria jabatan berhasil diubah dan data kriteria jabatan berhasil dihapus. Alur alternatif yang ada pada skenario ini adalah aktor *field* pada *form* tambah atau update data tidak lengkap dan aktor batal menyimpan data. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sitem pada skenario *usecase* mengelola data kriteria jabatan dapat dilihat pada lampiran A Tabel 4.

4.2.2.4 Skenario *Usecase* Melihat Hasil

Skenario *usecase* melihat hasil merupakan alur yang menjelaskan aktor pada saat melihat hasil perhitungan yang telah dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* mengelola hasil adalah *admin*. Prekondisi dari skenario *usecase* melihat hasil adalah aktor harus berhasil masuk terlebih dahulu untuk dapat melihat hasil. Prakondisi dari skenario *usecase* melihat hasil adalah hasil telah dilihat. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sitem pada skenario mengelola hasil dapat dilihat pada lampiran A Tabel 5.

4.2.2.5 Skenario *Usecase* Melihat Bobot Kepribadian

Skenario *usecase* melihat bobot kepribadian merupakan alur yang menjelaskan aktor pada saat melihat bobot kepribadian yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* melihat bobot kepribadian adalah *admin*. Prekondisi dari skenario *usecase* melihat bobot kepribadian adalah aktor harus berhasil masuk terlebih dahulu untuk dapat melihat bobot kepribadian. Prakondisi dari skenario *usecase* melihat bobot kepribadian adalah bobot kepribadian berhasil diliha. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sitem pada skenario *usecase* melihat bobot kepribadian dapat dilihat pada lampiran A Tabel 6.

4.2.2.6 Skenario *Usecase* Melihat Profil Perusahaan

Skenario *usecase* melihat profil perusahaan digunakan untuk menjelaskan alur dari proses melihat profil perusahaan yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* melihat profil perusahaan adalah *admin*. Prekondisi dari skenario *usecase* melihat profil perusahaan adalah aktor harus berhasil masuk terlebih dahulu untuk dapat melihat profil perusahaan. Prakondisi dari skenario *usecase* melihat profil perusahaan adalah profil berhasil dilihat. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sistem skenario *usecase* melihat profil perusahaan dapat dilihat pada lampiran A Tabel 7.

4.2.2.7 Skenario *Usecase* Daftar

Skenario *usecase* daftar digunakan untuk menjelaskan alur proses daftar ke dalam sistem serta alur alternatif yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* daftar adalah *karyawan*. Prekondisi dari skenario *usecase* daftar adalah aktor belum mendaftarkan data dirinya ke dalam sistem. Prakondisi dari skenario *usecase* daftar adalah aktor berhasil daftar dan bisa masuk ke dalam sistem. Alur alternatif yang ada pada skenario ini adalah *field* pada *form* daftar kosong. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sistem pada skenario *usecase* daftar dapat dilihat pada lampiran A Tabel 8.

4.2.2.8 Skenario *Usecase* Pick Me

Skenario *usecase* pick me digunakan untuk menjelaskan alur proses *pick me* yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* pick me adalah *karyawan*. Prekondisi dari skenario *usecase* pick me adalah aktor harus berhasil masuk terlebih dahulu untuk dapat melakukan pick me. Prakondisi dari skenario *usecase* pick me adalah aktor berhasil melakukan pick me. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sistem pada skenario *usecase* pick me dapat dilihat pada lampiran A Tabel 9.

4.2.2.9 Skenario *Usecase* Melihat Hasil

Skenario *use case* melihat hasil digunakan untuk menjelaskan alur dari proses melihat hasil perhitungan yang telah dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* melihat hasil adalah karyawan. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario *usecase* melihat hasil dapat dilihat pada lampiran A Tabel 10.

4.2.2.10 Skenario *Usecase* Mengelola Biodata

Skenario *usecase* mengelola biodata merupakan alur yang menjelaskan aktor pada saat melihat, mengubah atau mencetak biodata serta alur alternatif yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat pada skenario *usecase* mengelola biodata adalah *karyawan*. Prekondisi dari skenario *usecase* mengelola biodata adalah aktor harus berhasil masuk terlebih dahulu untuk dapat mengelola biodata. Prakondisi dari skenario *usecase* mengelola biodata adalah biodata berhasil diubah atau biodata berhasil dicetak. Alur alternatif yang ada pada skenario ini adalah *field* pada *form* ubah kosong dan aktor batal menyimpan atau mencetak biodata. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sistem pada skenario *usecase* mengelola biodata dapat dilihat pada Tabel 4.5.

Tabel 4. 5 Skenario *Usecase* Mengelola Biodata

ID	USC 011
Nama Use Case	Mengelola Biodata
Aktor	Karyawan
Deskripsi	Aktor membuka system untuk mengelola biodata
PreKondisi	Aktor telah melakukan login
PraKondisi	Aktor telah mengelola biodata
Skenario Normal: Melihat Biodata	
Aksi Aktor	Reaksi Sistem

1. Klik menu biodata	
	2. Menampilkan halaman biodata dengan tabel yang menampilkan: <ul style="list-style-type: none"> • Nama • Alamat e-Mail • Asal Kota • Tanggal Lahir • Jenis Kelamin • Jurusan • Universitas • No. Handphone Halaman ini juga menampilkan tombol ubah untuk mengubah biodata
Skenario Normal: Mengubah biodata	
Aksi Aktor	Reaksi Sistem
1. Klik menu biodata	
	2. Menampilkan halaman biodata dengan tabel yang menampilkan: <ul style="list-style-type: none"> • Nama • Alamat e-Mail • Asal Kota • Tanggal Lahir • Jenis Kelamin • Jurusan • Universitas • No. Handphone

	Halaman ini juga menampilkan tombol ubah untuk mengubah biodata
3. Klik tombol ubah pada data yang akan diubah	
	4. Menampilkan form ubah biodata (Asal Kota, Tanggal Lahir, Jurusan, Universitas, No. Handphone)
5. Aktor mengubah form biodata	
6. Klik tombol simpan	
	7. Menyimpan ke <i>database</i>
	8. Menampilkan halaman biodata dengan tabel yang menampilkan: <ul style="list-style-type: none"> • Nama • Alamat e-Mail • Asal Kota • Tanggal Lahir • Jenis Kelamin • Jurusan • Universitas • No. Handphone Halaman ini juga menampilkan tombol ubah untuk mengubah biodata
Skenario Alternatif: Inputan kosong	
Aksi Aktor	Reaksi Sistem
6. Klik tombol simpan	
	7. Menampilkan pesan “Data belum diubah”

	8. Menampilkan form ubah biodata (Asal Kota, Tanggal Lahir, Jurusan, Universitas, No. Handphone)
Skenario Alternatif: Klik tombol batal	
Aksi Aktor	Reaksi Sistem
6. Klik tombol batal	
	<p>7. Menampilkan halaman biodata dengan tabel yang menampilkan:</p> <ul style="list-style-type: none"> • Nama • Alamat e-Mail • Asal Kota • Tanggal Lahir • Jenis Kelamin • Jurusan • Universitas • No. Handphone <p>Halaman ini juga menampilkan tombol ubah untuk mengubah biodata</p>

4.2.2.11 Skenario *Usecase* Keluar

Skenario *use case keluar* digunakan untuk menjelaskan alur dari proses *keluar* dari sistem. Aktor yang terdapat pada skenario *usecase keluar* adalah *admin* dan karyawan. Prekondisi dari skenario *usecase keluar* adalah aktor sudah masuk ke sistem dan akan keluar dari sistem. Prakondisi dari skenario *usecase keluar* adalah aktor berhasil keluar dari sistem. Penjelasan lebih lengkap urutan aksi aktor dan reaksi sistem pada skenario *usecase* keluar dapat dilihat pada Tabel 4.6.

Tabel 4. 6 Skenario *Usecase* Keluar

ID	USC 013	
Nama Use Case	Keluar	
Aktor	Admin, karyawan	
Deskripsi	Aktor keluar dari sistem	
PreKondisi	Aktor akan keluar dari sistem	
PraKondisi	Aktor telah keluar dari sistem	
Skenario Normal: Keluar		
	Aksi Aktor	Reaksi Sistem
	1. Klik tombol keluar	
		2. Menampilkan halaman masuk

4.2.3 *Sequence* Diagram

Sequence diagram adalah diagram yang digunakan untuk menggambarkan interaksi yang terjadi antar objek di dalam sistem yang disusun pada sebuah urutan dan rangkaian waktu.

1.2.3.1 *Sequence* Diagram Melihat Data Karyawan

Sequence diagram melihat data karyawan digunakan untuk menggambarkan skenario *usecase* melihat data karyawan secara teknis. *Sequence* diagram melihat data karyawan berisi *function* dan *method* yang digunakan dalam proses melihat data karyawan. *Class* yang berhubungan dengan *sequence* diagram ini adalah `index.php`, `v_data_karyawan.php`, `DataKaryawan.php`, dan `m_karyawan.php`. *Class* `index.php` merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* `DataKaryawan.php` merupakan *controller* data karyawan. *Class* `m_karyawan.php` merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan

database. *Class* *v_data_karyawan.php* merupakan halaman data karyawan. *Sequence diagram* melihat data karyawan dapat dilihat pada lampiran B Gambar 1.

1.2.3.2 Sequence Diagram Mengelola Data Kriteria Jabatan

Sequence diagram mengelola data kriteria jabatan digunakan untuk menggambarkan skenario *usecase* mengelola data kriteria jabatan secara teknis. *Sequence diagram* mengelola data kriteria jabatan berisi *function* dan *method* yang digunakan dalam proses mengelola data kriteria jabatan. *Class* yang berhubungan dengan *sequence diagram* ini adalah *index.php*, *v_data_jabatan.php*, *DataKriteria.php*, dan *m_jabatan.php*. *Class* *index.php* merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* *DataKriteria.php* merupakan *controller* data kriteria jabatan. *Class* *m_jabatan.php* merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* *v_data_jabatan.php* merupakan halaman data kriteria jabatan. *Sequence diagram* mengelola data kriteria jabatan dapat dilihat pada lampiran B Gambar 2.

1.2.3.3 Sequence Diagram Melihat Hasil

Sequence diagram melihat hasil digunakan untuk menggambarkan skenario *usecase* melihat hasil secara teknis. *Sequence diagram* melihat hasil berisi *function* dan *method* yang digunakan dalam proses melihat hasil. *Class* yang berhubungan dengan *sequence diagram* ini adalah *index.php*, *v_hasil.php*, *Hasil.php*, dan *m_hasil.php*. *Class* *index.php* merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* *Hasil.php* merupakan *controller* hasil. *Class* *m_hasil.php* merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* *v_hasil.php* merupakan halaman hasil. *Sequence diagram* melihat hasil dapat dilihat pada lampiran B Gambar 3.

1.2.3.4 Sequence Diagram Melihat Bobot Kepribadian

Sequence diagram melihat bobot kepribadian digunakan untuk menggambarkan skenario *usecase* melihat data bobot kepribadian secara teknis. *Sequence diagram* melihat bobot kepribadian berisi *function* dan *method* yang digunakan dalam proses

melihat bobot kepribadian. *Class* yang berhubungan dengan *sequence* diagram ini adalah `index.php`, `v_data_jabatan.php`, `DataBobotKepribadian.php`, dan `m_jabatan.php`. *Class* `index.php` merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* `DataBobotKepribadian.php` merupakan *controller* bobot kriteria jabatan. *Class* `m_jabatan.php` merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* `v_data_jabatan.php` merupakan halaman bobot kepribadian. *Sequence diagram* melihat bobot kepribadian dapat dilihat pada lampiran B Gambar 4.

1.2.3.5 *Sequence Diagram* Melihat Profil Perusahaan

Sequence diagram melihat profil perusahaan digunakan untuk menggambarkan skenario *usecase* melihat profil perusahaan secara teknis. *Sequence diagram* melihat profil perusahaan berisi *function* dan *method* yang digunakan untuk melihat profil perusahaan. *Class* yang berhubungan dengan *sequence diagram* ini adalah `index.php`, `v_profil.php`, `DataProfil.php`, `v _profil.php` dan `m_profil.php`. *Class* `index.php` merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* `DataProfilPerusahaan.php` merupakan *controller* profil perusahaan. *Class* `m_profil.php` merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* `v _profil.php` merupakan halaman profil perusahaan. *Sequence diagram* melihat profil perusahaan dapat dilihat pada lampiran B Gambar 5.

1.2.3.6 *Sequence Diagram* Daftar

Sequence diagram daftar digunakan untuk menggambarkan skenario *usecase* daftar secara teknis. *Sequence diagram* daftar berisi *function* dan *method* yang digunakan untuk daftar. *Class* yang berhubungan dengan *sequence diagram* ini adalah `index.php`, `v_daftar.php`, `DataDaftar.php`, `v_daftar.php` dan `m_daftar.php`. *Class* `index.php` merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* `DataDaftar.php` merupakan *controller* daftar. *Class* `m_daftar.php` merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan

database. *Class* v_daftar.php merupakan halaman daftar. *Sequence* diagram daftar dapat dilihat pada lampiran B Gambar 6.

1.2.3.7 *Sequence* Diagram Pick Me

Sequence diagram pick me digunakan untuk menggambarkan skenario *usecase* pick me secara teknis. *Sequence* diagram pick me berisi *function* dan *method* yang digunakan untuk pick me. *Class* yang berhubungan dengan *sequence* diagram ini adalah index.php, v_pick_me.php, DataPickMe.php, v_pick_me.php dan m_pick_me.php. *Class* index.php merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* DataPickMe.php merupakan *controller* pick me. *Class* m_pick_me.php merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* v_pick_me.php merupakan halaman pick me. *Sequence* diagram pick me dapat dilihat pada lampiran B Gambar 7.

1.2.3.8 *Sequence* Diagram Melihat Hasil

Sequence diagram melihat hasil digunakan untuk menggambarkan skenario *usecase* melihat hasil secara teknis. *Sequence* diagram melihat hasil berisi *function* dan *method* yang digunakan untuk melihat hasil. *Class* yang berhubungan dengan *sequence* diagram ini adalah index.php, v_hasil.php, DataHasil.php, v_detail_hasil.php dan m_hasil.php. *Class* index.php merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* DataHasil.php merupakan *controller* hasil. *Class* m_hasil.php merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* v_detail_hasil.php merupakan halaman detail hasil. *Sequence* diagram melihat hasil dapat dilihat pada lampiran B Gambar 8.

4.2.3.9 *Sequence* Diagram Mengelola Biodata

Sequence diagram mengelola biodata digunakan untuk menggambarkan skenario *usecase* mengelola biodata secara teknis. *Sequence* diagram mengelola biodata berisi *function* dan *method* yang digunakan dalam proses mengelola biodata. *Class* yang berhubungan dengan *sequence* diagram ini adalah index.php, v_data_biodata.php, DataBiodata.php, dan m_biodata.php. *Class* index.php

merupakan *class* yang digunakan untuk menyimpan menu-menu utama sistem. *Class* *DataBiodata.php* merupakan *controller* biodata. *Class* *m_biodata.php* merupakan *class* yang digunakan untuk menyimpan *logic* dan berhubungan dengan *database*. *Class* *v_data_biodata.php* merupakan halaman biodata. *Sequence diagram* mengelola biodata dapat dilihat pada lampiran B Gambar 9.

4.2.4 Activity Diagram

Activity Diagram berfungsi untuk menggambarkan alur aktivitas yang akan dijalankan sistem dengan *user* dalam bentuk diagram aktivitas.

4.2.4.1 Activity Diagram Masuk

Activity diagram *masuk* menggambarkan alur aktivitas *masuk* dimulai pada saat aktor membuka halaman website kemudian sistem menampilkan halaman masuk. Selanjutnya, aktor mengisi *username* dan *password* serta klik tombol masuk. Jika gagal maka sistem akan menampilkan pesan error, jika *form* tidak lengkap maka menampilkan span *form* tidak lengkap, dan jika berhasil maka sistem menampilkan halaman dashboard. *Activity* diagram masuk dapat dilihat pada lampiran C Gambar 10 dan Gambar 11.

4.2.4.2 Activity Diagram Melihat Data Karyawan

Activity diagram melihat data karyawan menggambarkan alur aktivitas melihat data karyawan dimulai pada saat aktor memilih menu data karyawan kemudian sistem menampilkan halaman data karyawan. *Activity diagram* mengelola data karyawan dapat dilihat pada lampiran C Gambar 12.

4.2.4.3 Activity Diagram Mengelola Data Kriteria Jabatan

Activity diagram mengelola data kriteria jabatan menggambarkan alur aktivitas mengelola data kriteria jabatan dimulai pada saat aktor memilih menu data kriteria jabatan kemudian sistem menampilkan halaman data kriteria jabatan. Selanjutnya,

aktor bisa memilih untuk menambah, mengubah atau menghapus data kriteria jabatan. Jika data kriteria jabatan berhasil ditambah atau diubah atau dihapus maka sistem akan menampilkan pesan data berhasil ditambah atau diubah atau dihapus. Jika data kriteria jabatan gagal ditambah atau diubah maka sistem akan menampilkan span *form* tidak lengkap. Jika data kriteria jabatan batal ditambah atau diubah atau dihapus maka sistem akan menampilkan halaman data kriteria jabatan. *Activity diagram* mengelola data kriteria jabatan dapat dilihat pada lampiran C Gambar 13.

4.2.4.4 Activity Diagram Melihat Hasil

Activity diagram melihat hasil menggambarkan alur aktivitas melihat hasil dimulai pada saat aktor memilih menu hasil kemudian sistem menampilkan halaman hasil. *Activity diagram* mengelola hasil dapat dilihat pada lampiran C Gambar 14.

4.2.4.5 Activity Diagram Melihat Bobot Kepribadian

Activity diagram melihat bobot kepribadian menggambarkan alur aktivitas melihat bobot kepribadian jabatan dimulai pada saat aktor memilih menu bobot kepribadian kemudian sistem menampilkan halaman bobot kepribadian. *Activity diagram* mengelola bobot kriteria jabatan dapat dilihat pada lampiran C Gambar 15.

4.2.4.6 Activity Diagram Melihat Profil Perusahaan

Activity diagram melihat profil perusahaan menggambarkan alur aktivitas melihat profil perusahaan dimulai pada saat aktor memilih. *Activity diagram* melihat profil perusahaan dapat dilihat pada lampiran C Gambar 16.

4.2.4.7 Activity Diagram Daftar

Activity diagram daftar menggambarkan alur aktivitas untuk melakukan proses daftar dimulai pada saat aktor memilih menu daftar kemudian sistem akan menampilkan halaman daftar. Selanjutnya, aktor bisa mengisi data pribadi dan menyimpan data tersebut dengan klik tombol simpan. Jika data berhasil disimpan maka sistem akan menampilkan pesan data berhasil disimpan. Jika data gagal disimpan maka

sistem akan menampilkan span *form* tidak lengkap. *Activity diagram* daftar dapat dilihat pada lampiran C Gambar 17.

4.2.4.8 Activity Diagram Pick Me

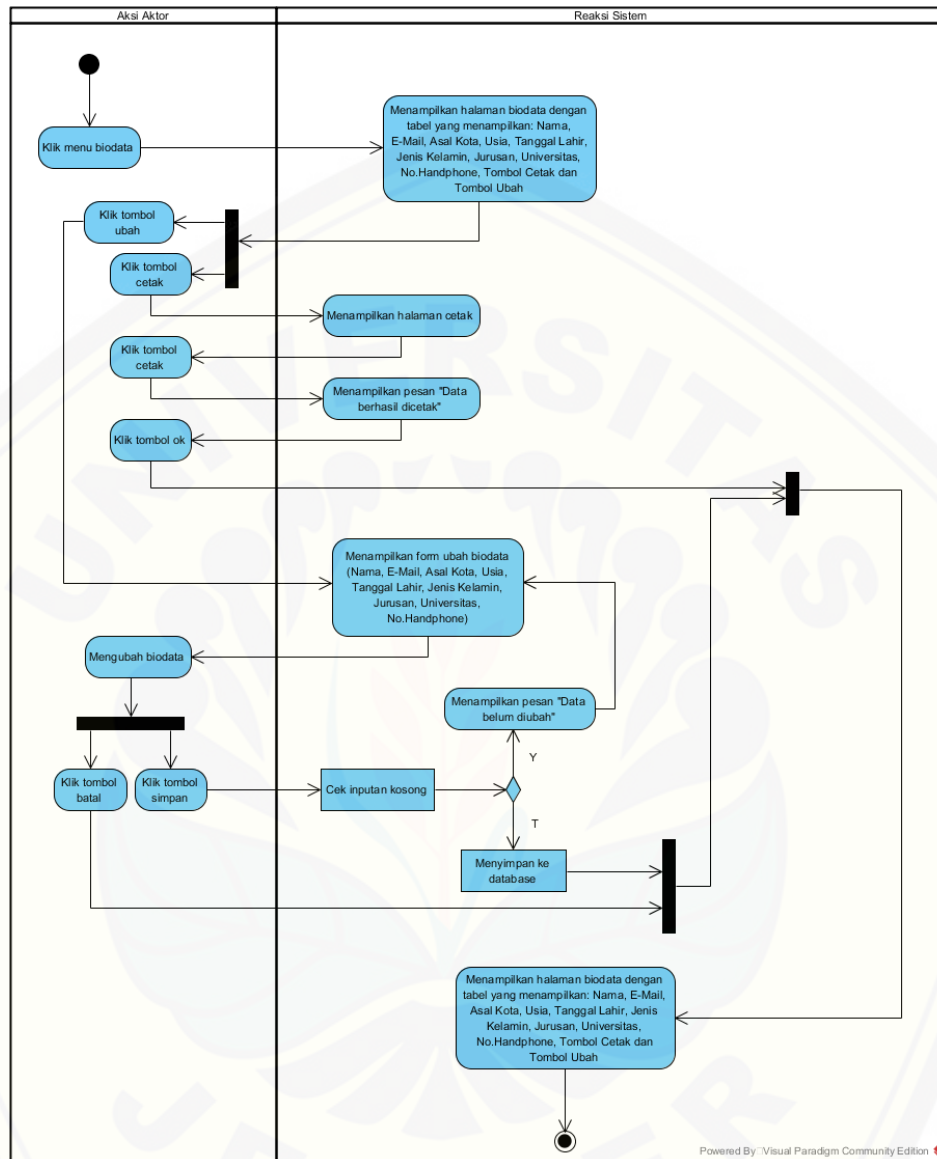
Activity diagram pick me menggambarkan alur aktivitas untuk melakukan pick me dimulai pada saat aktor memilih menu pick me kemudian sistem akan menampilkan halaman pick me. Selanjutnya, aktor bisa melakukan pick me dengan memilih jabatan dan kriteria yang sesuai dengan kepribadiannya. *Activity diagram* pick me dapat dilihat pada lampiran C Gambar 18.

4.2.4.9 Activity Diagram Melihat Hasil

Activity diagram melihat hasil menggambarkan alur aktivitas untuk melihat hasil dimulai pada saat aktor memilih menu hasil kemudian sistem akan menampilkan halaman hasil. Selanjutnya, aktor bisa melihat hasil beserta detailnya. *Activity diagram* melihat hasil dapat dilihat pada lampiran C Gambar 19.

4.2.4.10 Activity Diagram Mengelola Biodata

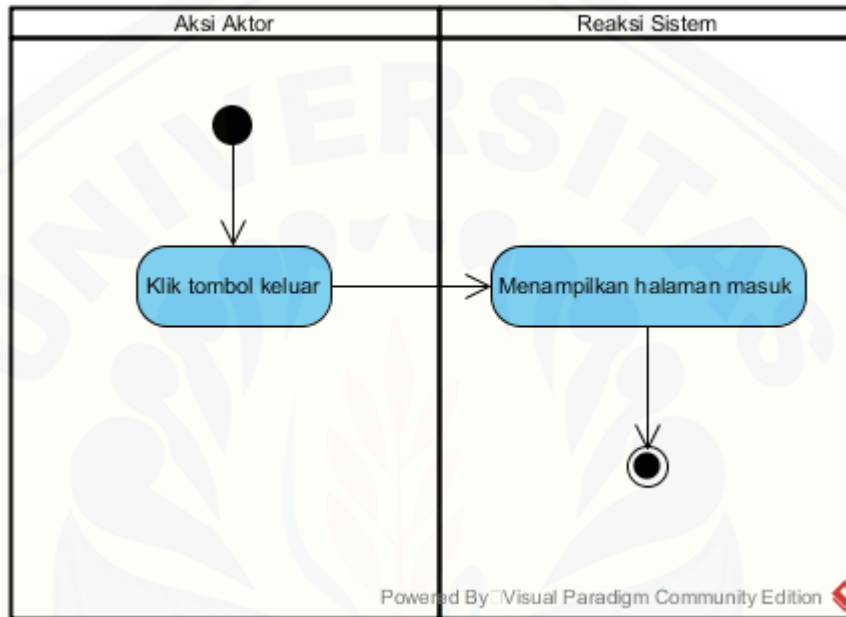
Activity diagram mengelola biodata menggambarkan alur aktivitas mengelola biodata dimulai pada saat aktor memilih menu biodata kemudian sistem menampilkan halaman biodata. Selanjutnya, aktor bisa memilih untuk mengubah atau mencetak biodata. Jika biodata berhasil diubah atau dicetak maka sistem akan menampilkan pesan data berhasil diubah atau dicetak. Jika biodata batal diubah atau dicetak maka sistem akan menampilkan halaman biodata. *Activity diagram* mengelola biodata dapat dilihat pada Gambar 4.3.



Gambar 4. 3 Activity Diagram Mengelola Biodata

4.2.4.11 Activity Diagram Keluar

Activity diagram *keluar* menggambarkan alur aktivitas *keluar* dimulai pada saat aktor memilih tombol keluar kemudian sistem menampilkan halaman masuk. Activity diagram keluar dapat dilihat pada Gambar 4.4.



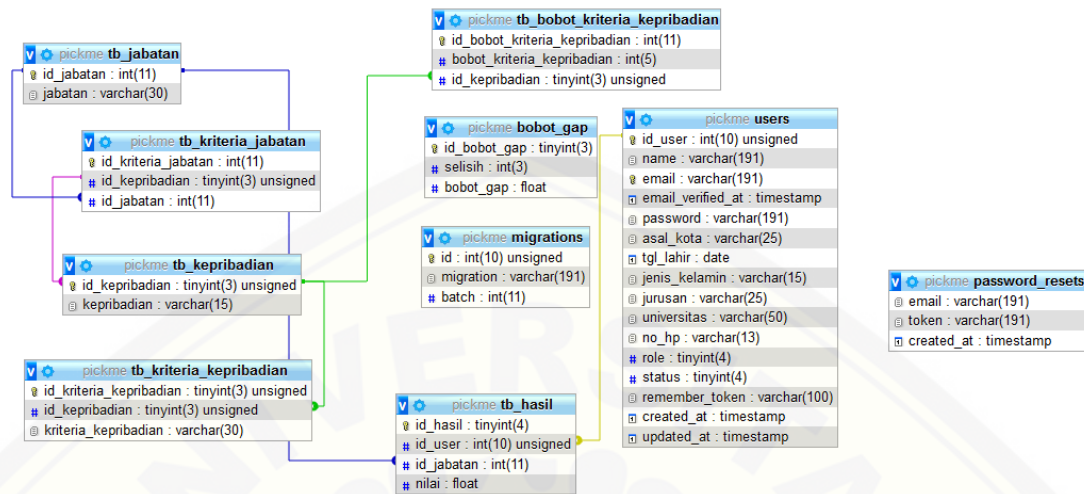
Gambar 4. 4 Activity Diagram Keluar

4.2.5 Class Diagram

Class Diagram menggambarkan hubungan antar kelas yang digunakan untuk membangun suatu sistem. Class Diagram sistem penentu jabatan karyawan di suatu perusahaan berdasarkan kepribadian manusia dapat dilihat pada Lampiran F.

4.2.6 Entity Relationship Diagram (ERD)

ERD merupakan gambaran komponen dan struktur database yang digunakan dalam pembangunan sistem. ERD sistem penentu jabatan karyawan di suatu perusahaan berdasarkan kepribadian manusia dapat dilihat pada Gambar 4.5.



Gambar 4. 5 Entity Relationship Diagram

4.3. Penulisan Kode Program dan Pengujian Sistem

4.3.1. Penulisan Kode Program

Penulisan kode program ini merupakan implementasi dari desain system yang telah dirancang sebelumnya. Pengujian dilakukan untuk mengevaluasi aplikasi yang telah dibuat. Proses pengujian dilakukan dengan pengujian blackbox.

4.3.1.1 Kode Program Masuk

Kode program masuk terletak pada controller MasukController, model User, dan view halaman-masuk. Terdapat pengecekan username dan password yang diinputkan user pada saat proses masuk. Jika username dan password tidak sesuai dengan yang terdapat di database, maka user tidak dapat masuk ke dalam sistem. Penulisan kode program masuk dapat dilihat pada lampiran D Gambar 20.

4.3.1.2 Kode Program Melihat Data Karyawan

Kode program melihat data karyawan terletak pada controller KaryawanController, view karyawan, view form-karyawan. Terdapat function index() yang digunakan untuk menampilkan halaman yang berisi semua data karyawan.

Penulisan kode program melihat data karyawan dapat dilihat pada lampiran D Gambar 21.

4.3.1.3 Kode Program Mengelola Data Kriteria Jabatan

Kode program mengelola data kriteria jabatan terletak pada controller `JabatanController`, view `jabatan`, view `kriteria-jabatan`, dan view `form-ubah-jabatan`. Terdapat function `index()` yang digunakan untuk menampilkan halaman yang berisi semua data kriteria jabatan, function `create()` untuk menampilkan halaman form tambah data kriteria jabatan, function `update()` untuk menampilkan halaman form ubat data kriteria jabatan dan function `delete()` untuk menghapus data kriteria jabatan yang dipilih. Penulisan kode program mengelola data kriteria jabatan dapat dilihat pada lampiran D Gambar 22.

4.3.1.4 Kode Program Melihat Hasil

Kode program melihat hasil terletak pada controller `HasilController`, view `hasil`. Terdapat function `index()` yang digunakan untuk menampilkan halaman yang berisi semua hasil. Penulisan kode program melihat hasil dapat dilihat pada lampiran D Gambar 23.

4.3.1.5 Kode Program Melihat Bobot Kepribadian

Kode program melihat bobot kepribadian terletak pada controller `JabatanController`, view `kepribadian`. Terdapat function `index()` yang digunakan untuk menampilkan halaman yang berisi semua bobot kepribadian. Penulisan kode program melihat bobot kepribadian dapat dilihat pada lampiran D Gambar 24.

4.3.1.6 Kode Program Melihat Profil Perusahaan

Kode program melihat profil perusahaan terletak pada controller `ProfilController`, view `profil`. Terdapat function `index()` yang digunakan untuk

menampilkan halaman yang berisi profil perusahaan. Penulisan kode program mengelola profil perusahaan dapat dilihat pada lampiran D Gambar 25.

4.3.1.7 Kode Program Daftar

Kode program melihat daftar terletak pada controller `DaftarController` dan view `form-daftar`. Terdapat function `index()` yang digunakan untuk menampilkan halaman daftar. Penulisan kode program daftar dapat dilihat pada lampiran D Gambar 26.

4.3.1.8 Kode Program Pick Me

Kode program pick me terletak pada controller `KepribadianController` dan view `pick-me`. Terdapat function `index()` yang digunakan untuk menampilkan halaman pick me dan function `tes()` untuk melakukan pick me. Penulisan kode program pick me dapat dilihat pada lampiran D Gambar 27.

4.3.1.9 Kode Program Melihat Hasil

Kode program melihat hasil terletak pada *controller* `HasilController` dan *view* `hasil`. Terdapat pendeklarasian variabel-variabel yang akan digunakan dalam *function* `index()` pada `HasilController` yang kemudian variabel-variabel tersebut akan diberi nilai dengan data yang tersedia di database. Setelah melakukan tes kepribadian, yang dilakukan adalah melakukan perhitungan menggunakan metode yang digunakan. Penulisan kode program melihat hasil dapat dilihat pada Lampiran D Gambar 28.

4.3.1.10 Kode Program Mengelola Biodata

Kode program mengelola biodata terletak pada controller `BiodataController`, view `biodata`, view `form-biodata`, dan view `form-ubah-biodata`. Terdapat function `index()` yang digunakan untuk menampilkan halaman yang berisi semua biodata, function `create()` untuk menampilkan halaman form tambah biodata, function `update()` untuk menampilkan halaman form ubah biodata dan function `delete()` untuk menghapus

biodata yang dipilih. Penulisan kode program mengelola biodata dapat dilihat pada lampiran D Gambar 29.

4.3.2. Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi aplikasi yang telah dibuat. Proses pengujian dilakukan dengan pengujian whitebox terlebih dahulu kemudian akan dilanjutkan dengan pengujian *blackbox*

4.3.2.1 Pengujian *Black Box*

Pengujian *black box* berfungsi untuk menguji sistem dari sisi fungsionalitas sistem dengan tujuan mengetahui apakah fitur-fitur, inputan dan keluaran sistem sesuai dengan kebutuhan pengguna. Fitur-fitur yang diujikan mencakup fitur melihat, menambah, mengubah dan menghapus data serta keluaran yang diharapkan dari sistem saat fitur tersebut dijalankan.

Namun dibalik semua kelebihan yang dimiliki, system ini juga masih memiliki kekurangan yaitu salah satunya tidak bisa digunakan untuk menentukan jabatan karyawan di perusahaan yang lainnya.

BAB 6. PENUTUP

Bab ini berisi mengenai kesimpulan dan saran dari peneliti tentang penelitian yang telah dilakukan. Kesimpulan dan saran tersebut diharapkan dapat digunakan sebagai acuan pada penelitian selanjutnya.

6.1. Kesimpulan

Kesimpulan dari hasil penelitian yang telah dilakukan adalah sebagai berikut:

1. Dalam melakukan pencocokan menggunakan metode *Profile Matching* untuk system penentu jabatan di suatu perusahaan berdasarkan kepribadian manusia, langkah-langkah yang perlu dilakukan adalah melakukan tes kepribadian manusia dari karyawan tersebut. Setelah itu melakukan perhitungan sesuai dengan metode *Profile Matching* dan pengujian terhadap tingkat kecocokannya. Tingkat kecocokan ini

ditentukan dari nilai *Gap* yang dihasilkan, semakin kecil nilai *Gap* maka tingkat kecocokan semakin besar.

2. Perancangan dan pembangunan sistem penentu jabatan di suatu perusahaan berdasarkan kepribadian manusia dibangun dengan menggunakan *Software Development Life Cycle (SDLC) waterfall* karena model ini memiliki proses yang urut digunakan dan setiap proses tidak tumpang tindih sehingga mudah diterapkan. Analisis kebutuhan dilakukan untuk mendapatkan kebutuhan sistem baik fungsional maupun non fungsional. Analisis kebutuhan dilakukan dengan melakukan wawancara dengan pihak terkait. Desain sistem dilakukan dengan merancang diagram yang akan digunakan sebagai panduan dalam pengkodean sistem. Implementasi sistem dilakukan dengan melakukan pengkodean yang menggunakan menggunakan *framework* Laravel 5.3 yang merupakan *framework* dengan bahasa pemrograman PHP dan menggunakan MySQL sebagai *database management system*. Sedangkan implementasi metode menggunakan metode *Profile Matching*.

6.2. Saran

Beberapa saran yang ditujukan untuk memberikan masukan yang lebih baik pada penelitian selanjutnya yaitu:

1. Pengembangan system pada penelitian ini tidak dapat dilakukan untuk menentukan jabatan karyawan di perusahaan lain nya. Diharapkan pada penelitian selanjutnya, system yang dikembangkan mampu menentukan jabatan karyawan di satu perusahaan secara umum.
2. System yang dikembangkan pada penelitian ini tidak mampu melakukan skala prioritas terhadap karyawan yang memiliki kepribadian yang sama untuk jabatan yang sama. Diharapkan pada penelitian selanjutnya, system yang

dikembangkan dapat melakukan skala prioritas sehingga system yang dikembangkan lebih kompleks.



DAFTAR PUSTAKA

- Alex S. Nitisemito, *Managemen Sumber Daya Manusia*, Sasmito Bross, Jakarta 1980.
- Asad, D. M. (2003). *Psikologi Industri*. Yogyakarta: Liberty.
- Feist, J., & Feist, G. J. (2006). *Theories of Personality*. New York: Mc Graw Hill
- Feist, J., & Feist, G. (2008). *Theories of Personality*. yogyakarta: pustaka pelajar
- Fitria. (2019). Interview Kriteria Kepribadian Manusia (A. H. Febrianto, Pewawancara)
- Gouzali Saydam, (2000), *Manajemen Sumber Daya Manusia (Human Resource) Suatu Pendekatan Mikro*. Jakarta: Djanbatan
- Handoko, T., Hani, 1998, *Manajemen dan Sumber Daya Manusia*, Yogyakarta, Liberty
- Hidayat, A. L., & Pinandita, T. (2013). Sistem Pendukung Keputusan Evaluasi Kinerja Karyawan Untuk Promosi Jabatan Struktural Pada Bimbingan Belajar Sciencemaster Menggunakan Metode GAP Kompetensi. *Jurnal Teknologi Technoscintia*, 211-220
- Kusrini. (2007). *Konsep dan Aplikasi Sistem Pendukung Keputusan*. Yogyakarta: Andi.
- Marsono, Boy, A. F., & Dari, W. (2015). Sistem Pendukung Keputusan Pemilihan Menu Makanan pada Penderita Obesitas dengan menggunakan Metode Topsis. *Jurnal Ilmiah Saindikom Vol.14, No.3, 2*.
- McCrae, R. (2011). 5 Years of Progress: A Reply to Block. *Journal of Research in Personality*, 108-113

- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015, Agustus). Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis. *Jurnal Ilmiah Teknologi Informasi Terapan, Volume I, No. 3*, 31-36.
- Pervin, Lawrence A. dkk. 2010. Psikologi Kepribadian: Teori dan Penelitian (terjemah oleh A.K. Anwar). Jakarta: Kencana Prenada Media Group
- Riyadi, A. S., Retnandi, E., & Deddy, A. (2012). Perancangan Sistem Informasi Berbasis Website Subsistem Guru Di Sekolah Pesantren Persatuan Islam 99 Rancabango. *Jurnal Algoritma Sekolah Tinggi Teknologi Garut*
- Robbins, S. P., & Judge, A. T. (2011). *Organizational Behavioral*. New Jersey: Pearson.
- Saragih, M. A. (2015). Sistem Pendukung Keputusan Penilaian Premises pada Bank CIMB Niaga Medan Menggunakan Metode Analytical Hierarcy Process (AHP). *Pelita Informatika Budi Darma, Vol. 9, No.2*, 2.
- Sommerville, I. (2011). *Software Engineering 9th Edition*. Addison-Wesley.
- Tina, S. (2019). Interview Kriteria Jabatan Perusahaan PT. Perkebunan Nusantara XII Kertowono (A. H. Febrianto, Pewawancara)

LAMPIRAN

A. *Usecase Skenario*

Tabel 1 Skenario Usecase Masuk (Admin)

ID	USC 001
Nama Use Case	Masuk
Aktor	Admin
Deskripsi	Aktor membuka system untuk masuk kehalaman home admin
PreKondisi	Aktor akan melakukan login
PraKondisi	Aktor telah melakukan login
Skenario Normal: Masuk	
Aksi Aktor	Reaksi Sistem
1. Membuka system pick me	
	2. Menampilkan halaman masuk
3. Memasukkan alamat e-mail, password	
4. Klik tombol masuk	
	5. Menampilkan halaman beranda pick me
Skenario Alternatif: alamat e-mail salah	
Aksi Aktor	Reaksi Sistem
4. Klik tombol Masuk	
	5. Menampilkan pesan alamat e-mail atau password salah
	6. Menampilkan halaman masuk
Scenario Alternatif: password salah	
Aksi Aktor	Reaksi Sistem

4. Klik tombol Masuk	
	5. Menampilkan pesan alamat e-mail atau password salah
	6. Menampilkan halaman masuk
Skenario Alternatif: username dan password salah	
Aksi Aktor	Reaksi Sistem
4. Klik tombol masuk	
	5. Menampilkan pesan alamat e-mail atau password salah
	6. Menampilkan halaman masuk

Tabel 2 Skenario Usecase Masuk (Karyawan)

ID	USC 001
Nama Use Case	Masuk
Aktor	Karyawan
Deskripsi	Aktor membuka system untuk masuk kehalaman home karyawan
PreKondisi	Aktor akan melakukan login
PraKondisi	Aktor telah melakukan login
Skenario Normal: Masuk	
Aksi Aktor	Reaksi Sistem
1. Membuka system pick me	
	2. Menampilkan halaman masuk
3. Memasukkan alamat e-mail, password	
4. Klik tombol masuk	
	5. Menampilkan halaman beranda pick me

Skenario Alternatif: username salah	
Aksi Aktor	Reaksi Sistem
4. Klik tombol masuk	
	5. Menampilkan pesan alamat e-mail atau password salah
	6. Menampilkan halaman masuk
Skenario Alternatif: password salah	
Aksi Aktor	Reaksi Sistem
4. Klik tombol masuk	
	5. Menampilkan pesan alamat e-mail atau password salah
	6. Menampilkan halaman masuk
Skenario Alternatif: username dan password salah	
Aksi Aktor	Reaksi Sistem
4. Klik tombol masuk	
	5. Menampilkan pesan alamat e-mail atau password salah
	6. Menampilkan halaman masuk

Tabel 3 (Skenario Usecase Melihat Data Karyawan)

ID	USC 002
Nama Use Case	Melihat Data Karyawan
Aktor	Admin
Deskripsi	Aktor membuka sistem untuk melihat data karyawan
PreKondisi	Aktor akan melihat data karyawan
PraKondisi	Aktor telah melihat data karyawan

Skenario Normal: Melihat Data Karyawan	
Aksi Aktor	Reaksi Sistem
1. Klik menu data karyawan	
	2. Menampilkan halaman karyawan dengan tabel yang menampilkan: <ul style="list-style-type: none"> • No. • Nama • Alamat e-Mail • Asal Kota • Tanggal Lahir • Jenis Kelamin • Jurusan • Universitas • No. Handphone

Tabel 4 (Skenario Usecase Mengelola Data Kriteria Jabatan)

ID	USC 003
Nama Use Case	Mengelola Data Kriteria Jabatan
Aktor	Admin
Deskripsi	Aktor membuka sistem untuk melihat, menambah, mengubah data kriteria jabatan
PreKondisi	Aktor telah melakukan login
PraKondisi	Data kriteria jabatan telah dikelola
Skenario Normal: Melihat data kriteria jabatan	
Aksi Aktor	Reaksi Sistem
1. Klik menu data kriteria jabatan	

	<p>2. Menampilkan halaman kriteria jabatan dengan table yang menampilkan:</p> <ul style="list-style-type: none"> • No. • Jabatan • Kriteria • Tombol ubah • Tombol tambah kriteria
Skenario Normal: Menambah data kriteria jabatan	
Aksi Aktor	Reaksi Aktor
1. Klik menu data kriteria jabatan	
	<p>2. Menampilkan halaman kriteria jabatan dengan table yang menampilkan:</p> <ul style="list-style-type: none"> • No. • Jabatan • Kriteria • Tombol ubah • Tombol tambah kriteria
3. Klik tombol tambah kriteria	
	4. Menampilkan form kriteria jabatan baru (nama jabatan, kriteria jabatan)
5. Klik tombol simpan	
	6. Menyimpan ke <i>database</i>
	<p>7. Menampilkan halaman kriteria jabatan dengan table yang menampilkan:</p> <ul style="list-style-type: none"> • No. • Jabatan • Kriteria • Tombol ubah • Tombol tambah kriteria

Skenario Alternatif: Inputan kosong	
Aksi Aktor	Reaksi Aktor
5. Klik tombol simpan	
	6. Menampilkan pesan “Data belum diisi”
Skenario Normal: Mengubah data kriteria jabatan	
Aksi Aktor	Reaksi Sistem
1. Klik menu data kriteria jabatan	
	2. Menampilkan halaman kriteria jabatan dengan table yang menampilkan: <ul style="list-style-type: none"> • No. • Jabatan • Kriteria • Tombol ubah • Tombol tambah kriteria
3. Klik tombol ubah pada data yang akan diubah	
	4. Menampilkan form ubah kriteria jabatan (nama jabatan, kriteria jabatan)
5. Aktor mengubah form data kriteria jabatan (nama jabatan, kriteria jabatan)	
6. Klik tombol simpan	
	7. Menyimpan ke <i>database</i>
	8. Menampilkan halaman kriteria jabatan dengan table yang menampilkan:

	<ul style="list-style-type: none"> • No. • Jabatan • Kriteria • Tombol ubah • Tombol tambah kriteria
Skenario Alternatif: Inputan kosong	
Aksi Aktor	Reaksi Sistem
6. Klik tombol simpan	
	7. Menampilkan pesan “Data belum diisi”
	8. Menampilkan form ubah kriteria jabatan (nama jabatan, kriteria jabatan)

Tabel 5 (Skenario Usecase Melihat Hasil)

ID	USC 004
Nama Use Case	Melihat Hasil
Aktor	Admin
Deskripsi	Aktor membuka system untuk melihat hasil
PreKondisi	Aktor telah melakukan login
PraKondisi	Aktor telah melihat hasil
Skenario Normal: Melihat hasil	
Aksi Aktor	Reaksi Sistem
1. Klik menu hasil	
	2. Menampilkan jabatan perusahaan
3. Aktor memilih jabatan yang akan dilihat hasilnya	

	4. Menampilkan halaman hasil dengan table yang menampilkan: <ul style="list-style-type: none"> • No. • Nama • Nilai Halaman ini juga menampilkan tombol kembali untuk kembali ke halaman sebelumnya
5. Klik tombol kembali	
	6. Menampilkan jabatan perusahaan

Tabel 6 (Skenario Usecase Melihat Bobot Kepribadian)

ID	USC005
Nama Use Case	Melihat Bobot Kepribadian
Aktor	Admin
Deskripsi	Aktor membuka system untuk melihat bobot kepribadian
PreKondisi	Aktor telah melakukan login
PraKondisi	Aktor telah melihat bobot kepribadian
Skenario Normal: Melihat bobot kepribadian	
Aksi Aktor	Reaksi Sistem
1. Klik menu bobot kepribadian	
	2. Menampilkan halaman bobot kepribadian dengan table yang menampilkan: <ul style="list-style-type: none"> • No. • Kepribadian • Bobot Kepribadian

Tabel 7 (Skenario Usecase Melihat Profil Perusahaan)

ID	USC 006
Nama Use Case	Melihat Profil Perusahaan
Aktor	Admin
Deskripsi	Aktor membuka system untuk melihat profil perusahaan
PreKondisi	Aktor telah melakukan login
PraKondisi	Aktor telah melihat profil perusahaan
Skenario Normal: Melihat profil perusahaan	
Aksi Aktor	Reaksi Sistem
1. Klik menu profil perusahaan	
	2. Menampilkan halaman profil perusahaan

Tabel 8 (Skenario Usecase Daftar)

ID	USC 007
Nama Use Case	Daftar
Aktor	Karyawan
Deskripsi	Aktor membuka system untuk melakukan daftar
PreKondisi	Aktor akan melakukan daftar
PraKondisi	Aktor telah melakukan daftar
Skenario Normal: Masuk	
Aksi Aktor	Reaksi Sistem
1. Membuka system pick me	
	2. Menampilkan halaman masuk
3. Klik tombol daftar	

	<p>4. Menampilkan halaman daftar dengan form yang menampilkan:</p> <ul style="list-style-type: none"> • Nama • Alamat e-mail • Password • Tanggal Lahir • Jenis Kelamin • Asal Kota • Jurusan • Universitas • No. Handphone <p>Halaman ini juga menampilkan tombol daftar</p>
5. Memasukkan nama, alamat e-mail, password, tanggal lahir, jenis kelamin, asal kota, jurusan, universitas, no. handphone	
6. Klik tombol daftar	
	7. Menampilkan pesan “Selamat, Anda sudah terdaftar”
8. Klik tombol masuk	
	9. Menampilkan halaman masuk
Skenario Alternatif: inputan kosong	
Aksi Aktor	Reaksi Sistem
6. Klik tombol daftar	
	7. Menampilkan pesan “Data belum lengkap”
	8. Menampilkan halaman daftar dengan form yang menampilkan: <ul style="list-style-type: none"> • Nama

	<ul style="list-style-type: none"> • Alamat e-mail • Password • Tanggal Lahir • Jenis Kelamin • Asal Kota • Jurusan • Universitas <p>Halaman ini juga menampilkan tombol daftar</p>
--	--

Tabel 9 (Skenario Usecase Pick Me)

ID	USC 009
Nama Use Case	Pick Me
Aktor	Karyawan
Deskripsi	Aktor membuka sistem untuk melakukan proses Pick Me
PreKondisi	Aktor telah melakukan login
PraKondisi	Aktor telah melakukan proses Pick Me
Skenario Normal: Pick Me	
Aksi Aktor	Reaksi Sistem
1. Klik menu Pick Me	
	2. Menampilkan halaman dengan pilihan jabatan perusahaan berupa tombol: <ul style="list-style-type: none"> • Pemasaran • Administrasi Keuangan • HRD
3. Pilih salah satu jabatan	
4. Klik tombol Next	

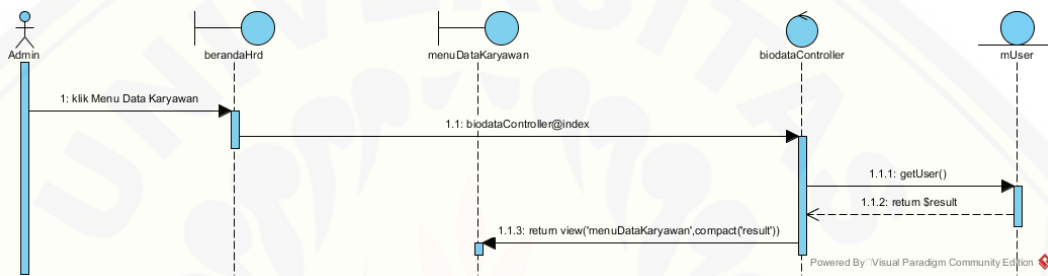
	5. Menampilkan halaman untuk menentukan kepribadian manusia
6. Pilih tiga pilihan yang sesuai dengan kepribadian masing masing	
7. Klik tombol submit	
	8. Menyimpan ke database
	9. Menampilkan halaman dengan tabel yang menampilkan: <ul style="list-style-type: none"> • Nama • Jabatan • Nilai

Tabel 10 (Skenario Usecase Melihat Hasil)

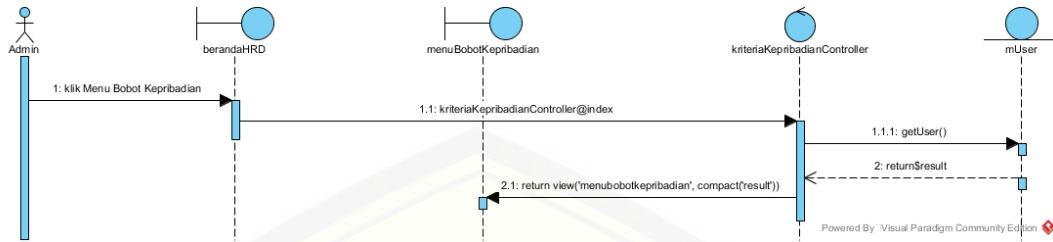
ID	USC 010
Nama Use Case	Melihat Hasil
Aktor	Karyawan
Deskripsi	Aktor membuka system untuk melihat hasil
PreKondisi	Aktor telah melakukan login
PraKondisi	Aktor telah melihat hasil
Skenario Normal: Melihat hasil	
Aksi Aktor	Reaksi Sistem
1. Klik menu hasil	
	2. Menampilkan halaman hasil dengan tabel yang menampilkan: <ul style="list-style-type: none"> • Nama • Jabatan • Nilai

	Halaman ini juga menampilkan tombol cetak untuk mencetak hasil
--	--

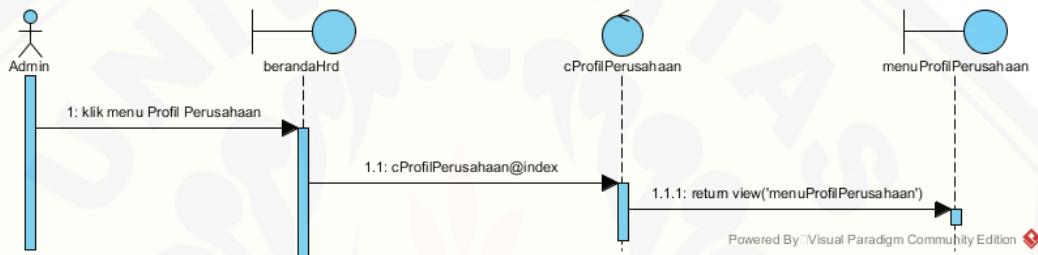
B. Sequence Diagram



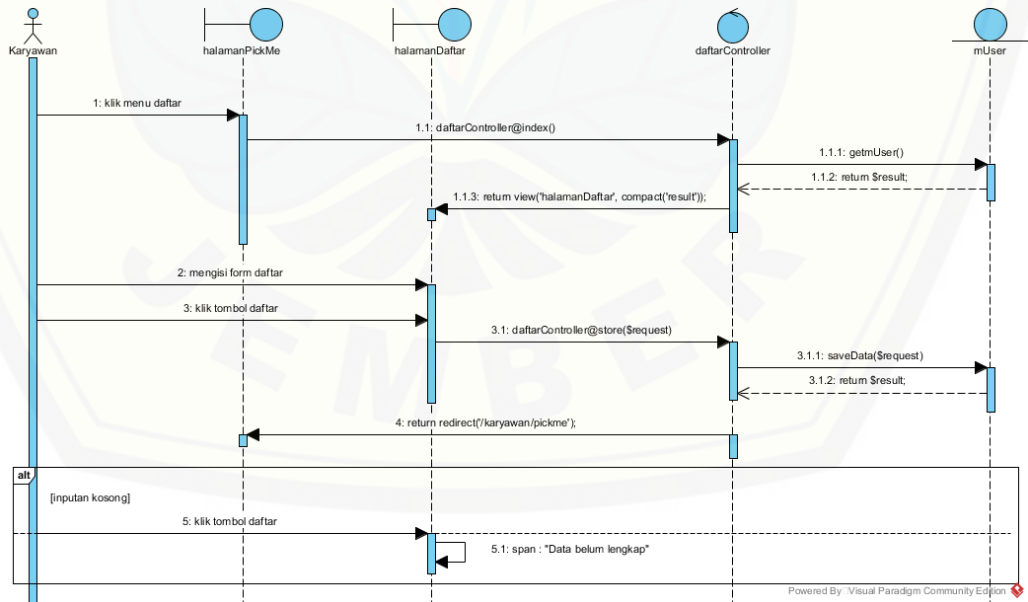
Gambar 1 (Sequence Diagram Melihat Data Karyawan)



Gambar 4 (Sequence Diagram Melihat Bobot Kepribadian)

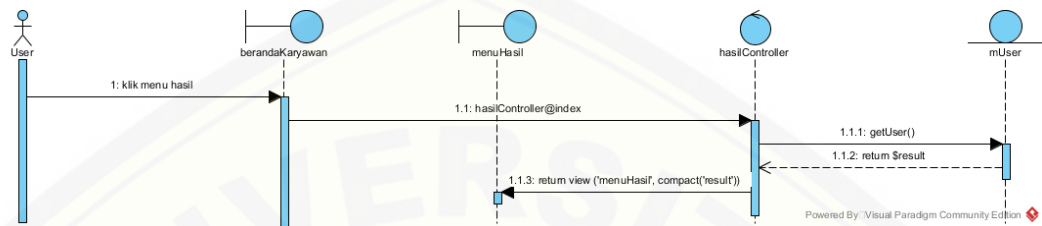


Gambar 5 (Sequence Diagram Melihat Profil Perusahaan)

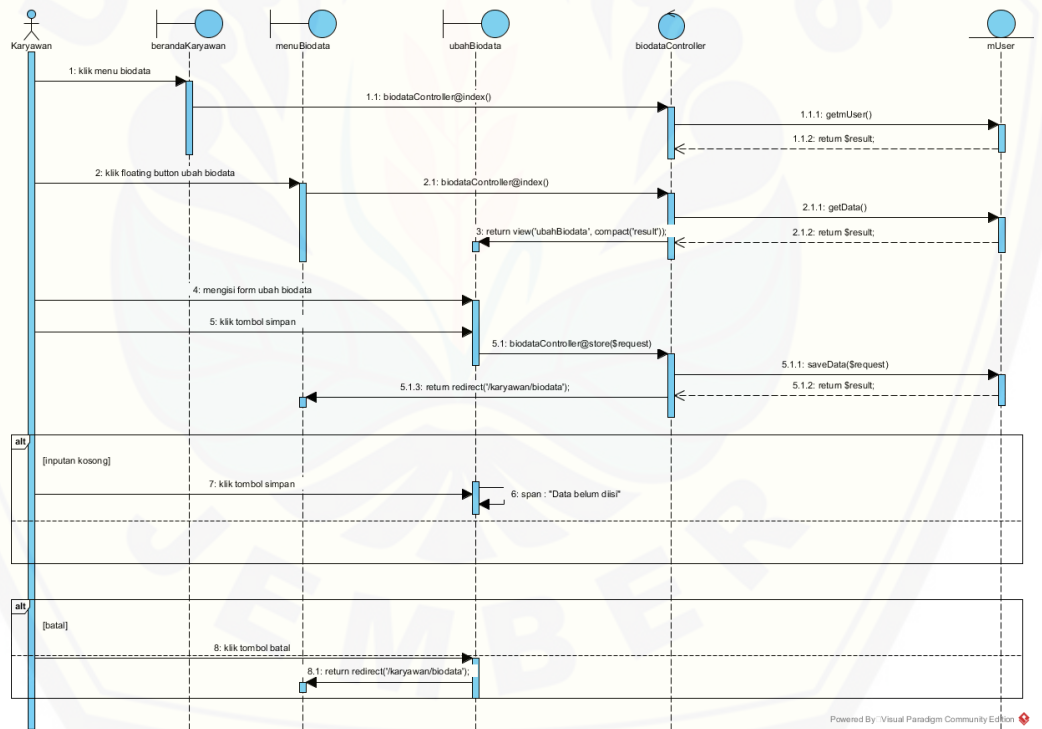


Gambar 6 (Sequence Diagram Daftar)

Gambar 7 (Sequence Diagram Pick Me)

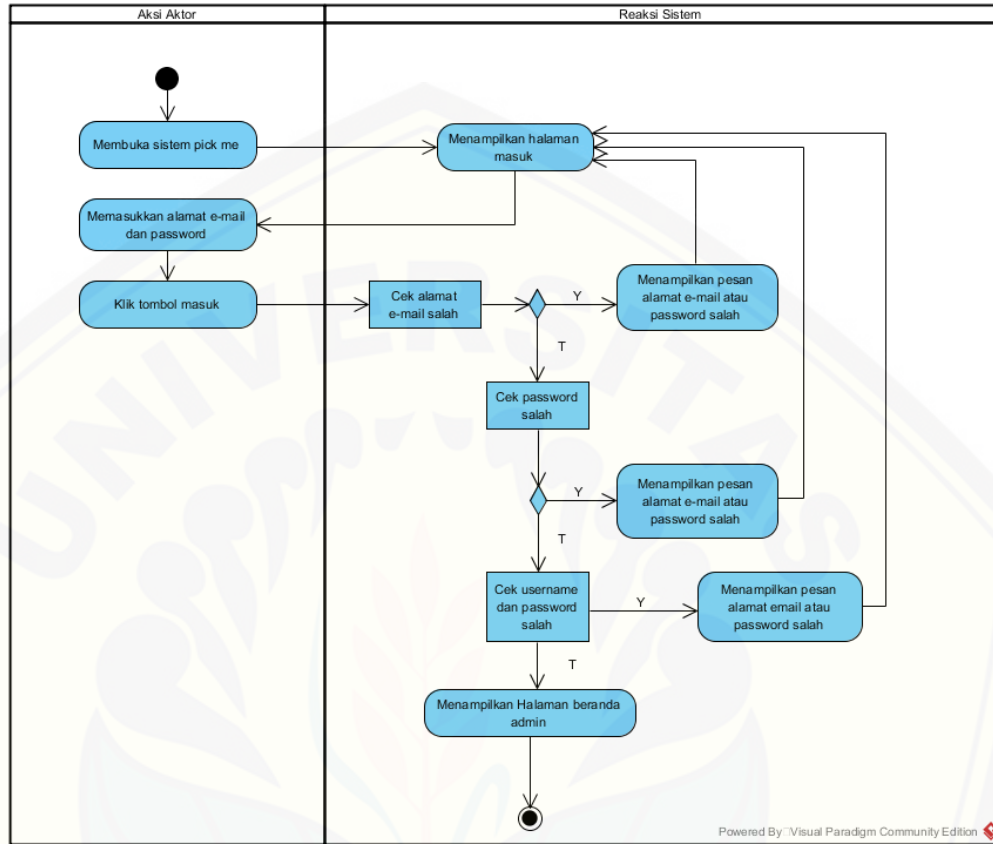


Gambar 8 (Sequence Diagram Melihat Hasil)

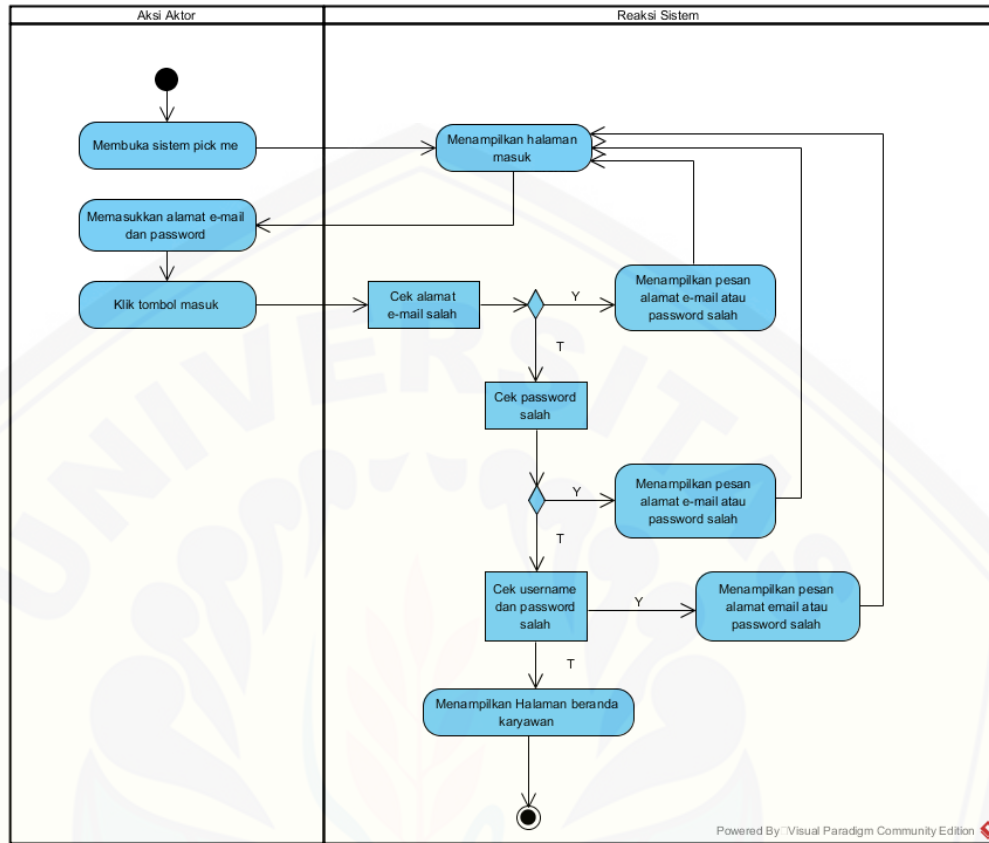


Gambar 9 (Sequence Diagram Mengelola Biodata)

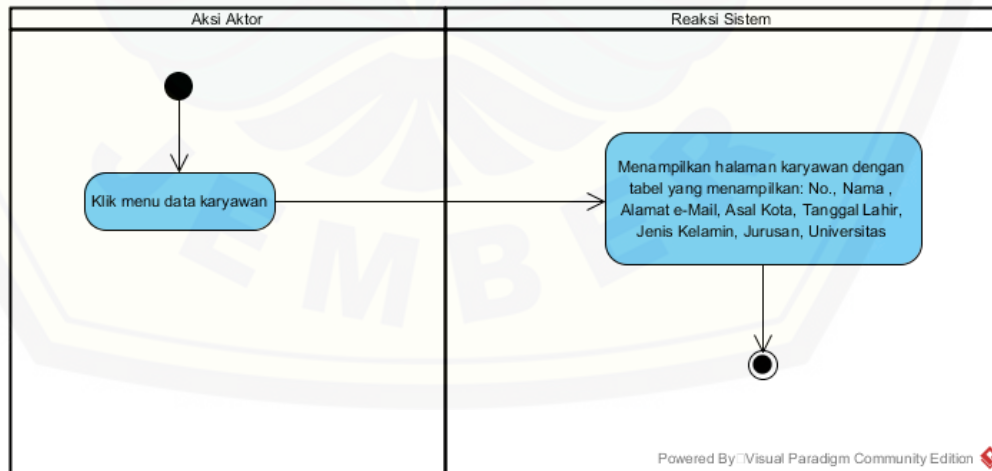
C. *Activity Diagram*



Gambar 10 (Activity Diagram Masuk Admin)

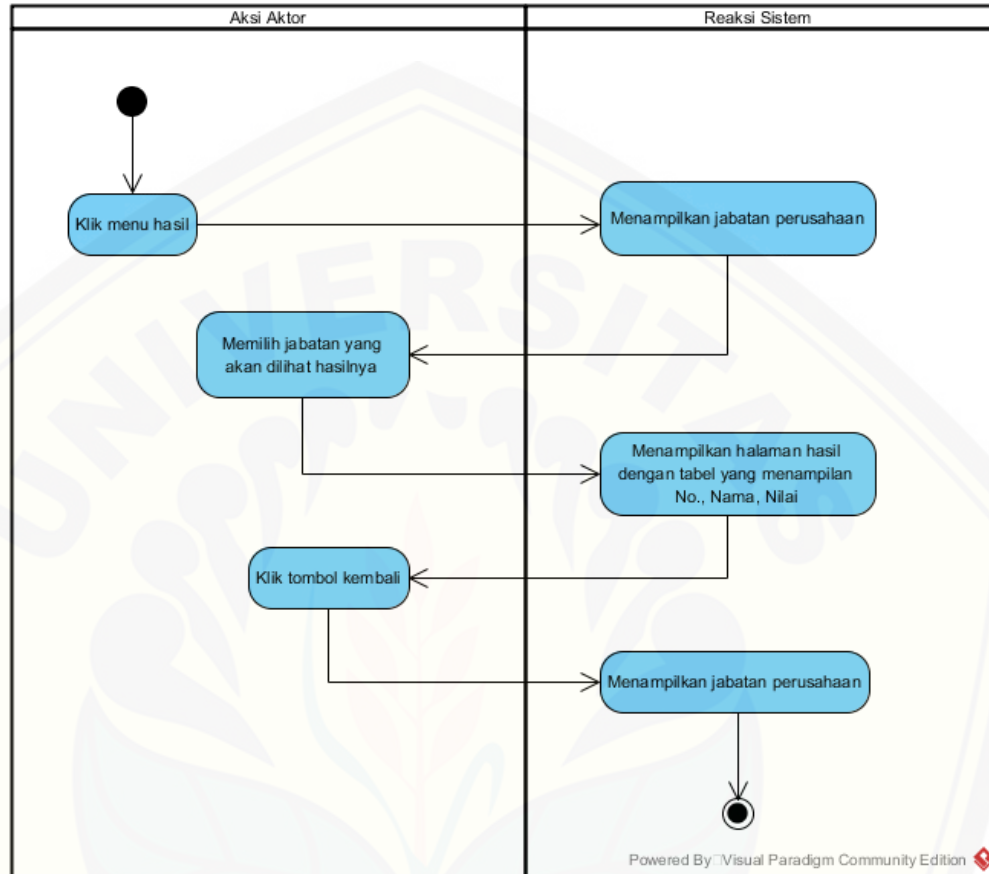


Gambar 11 (Activity Diagram Masuk Karyawan)

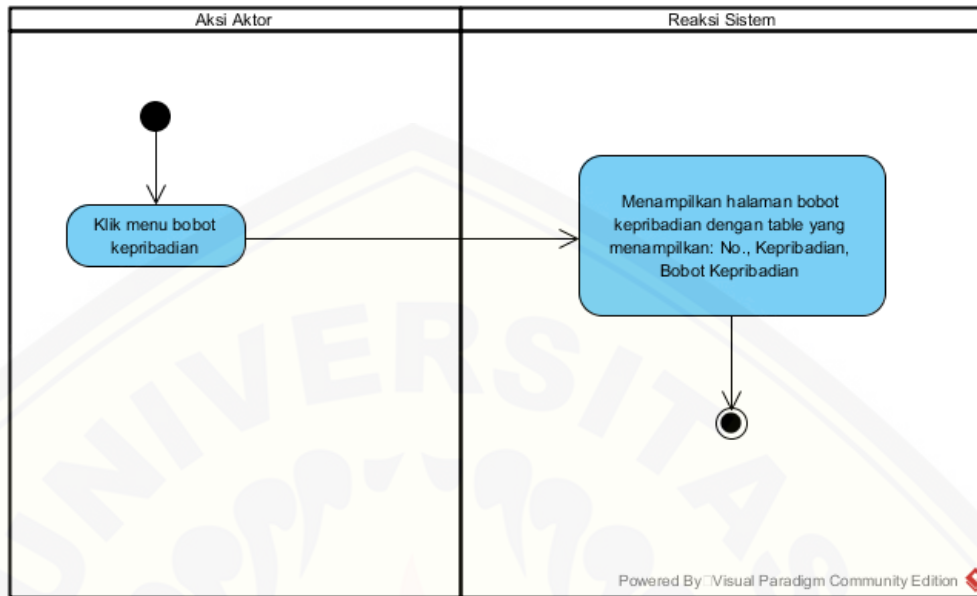


Gambar 12 (Activity Diagram Melihat Data Karyawan)

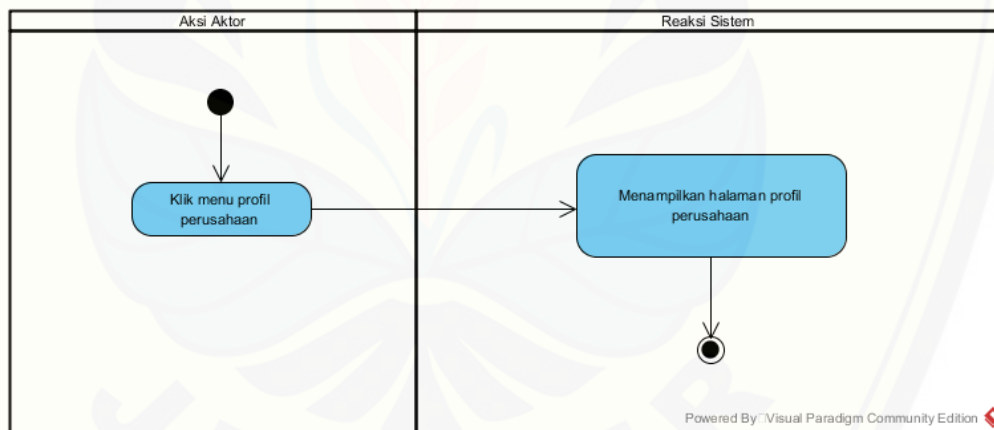
Gambar 13 (Activity Diagram Mengelola Data Kriteria Jabatan)



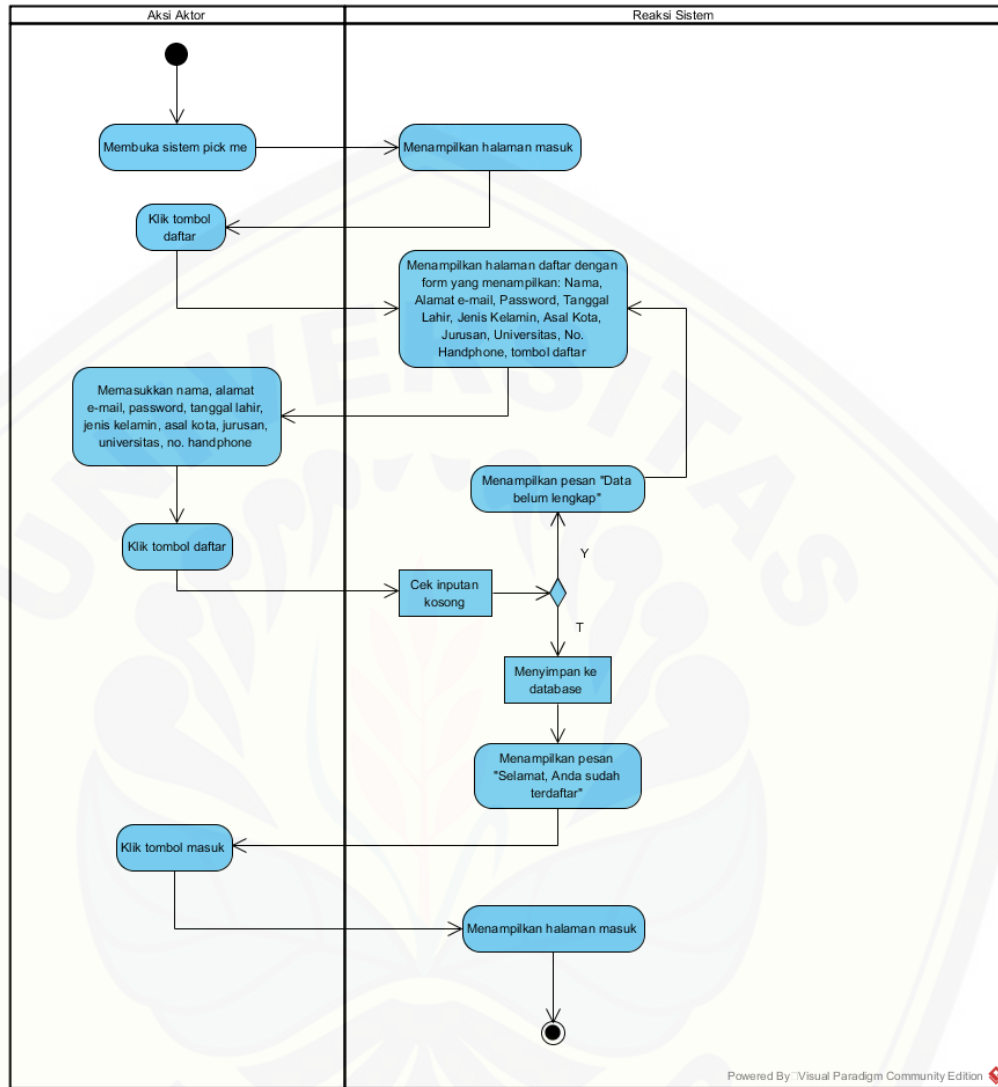
Gambar 14 (Activity Diagram Melihat Hasil)



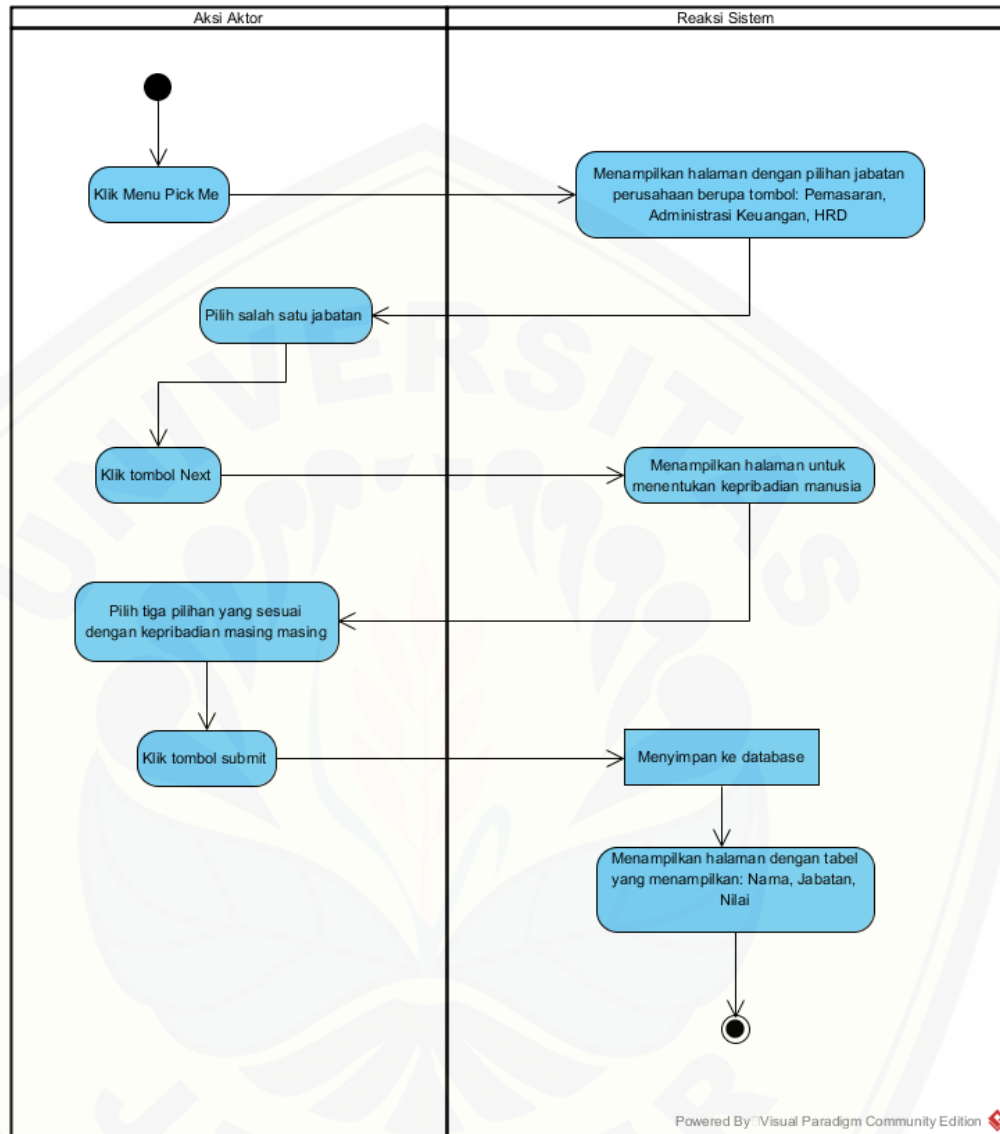
Gambar 15 (Activity Diagram Melihat Bobot Kepribadian)



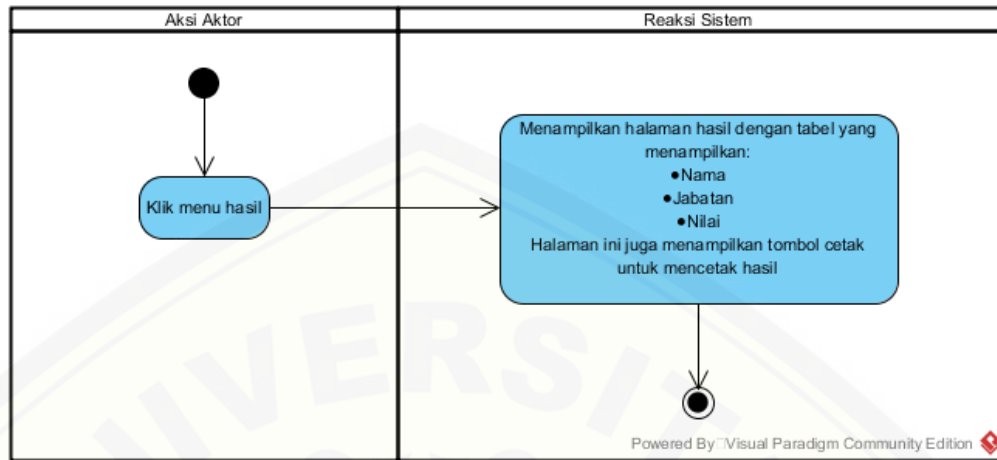
Gambar 16 (Activity Diagram Melihat Profil Perusahaan)



Gambar 17 (Activity Diagram Daftar)



Gambar 18 (Activity Diagram Pick Me)



Gambar 19 (Activity Diagram Melihat Hasil)

D. Penulisan Kode Program

```

1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5   <div class="row justify-content-center">
6     <div class="col-md-8">
7       <div class="card">
8         <div class="card-header">{{ __('Login') }}</div>
9
10        <div class="card-body">
11          <form method="POST" action="{{ route('login') }}">
12            @csrf
13
14            <div class="form-group row">
15              <label for="email" class="col-md-4 col-form-label text-md-right">{{ __('E-Mail Address') }}</label>
16
17              <div class="col-md-6">
18                <input id="email" type="email"
19                  class="form-control{{ $errors->has('email') ? ' is-invalid' : '' }}"
20                  name="email" value="{{ old('email') }}" required autofocus>
21
22                @if ($errors->has('email'))
23                  <span class="invalid-feedback" role="alert">
24                    <strong>{{ $errors->first('email') }}</strong>
25                  </span>
26                @endif
27              </div>
28            </div>
29
30            <div class="form-group row">
31              <label for="password" class="col-md-4 col-form-label text-md-right">{{ __('Password') }}</label>
32
33              <div class="col-md-6">
34                <input id="password" type="password"
35                  class="form-control{{ $errors->has('password') ? ' is-invalid' : '' }}" name="password" required>

```

```
36
37     @if ($errors->has('password'))
38         <span class="invalid-feedback" role="alert">
39             <strong>{{ $errors->first('password') }}</strong>
40         </span>
41     @endif
42 </div>
43 </div>
44
45 <div class="form-group row">
46     <div class="col-md-6 offset-md-4">
47         <div class="form-check">
48             <input class="form-check-input" type="checkbox"
49                 name="remember" id="remember" {{ old('remember') ? 'checked' : '' }}>
50
51             <label class="form-check-label" for="remember">
52                 {{ __('Remember Me') }}
53             </label>
54         </div>
55     </div>
56 </div>
57
58 <div class="form-group row mb-0">
59     <div class="col-md-8 offset-md-4">
60         <button type="submit" class="btn btn-primary">
61             {{ __('Login') }}
62         </button>
63
64         @if (Route::has('password.request'))
65             <a class="btn btn-link" href="{{ route('password.request') }}">
66                 {{ __('Forgot Your Password?') }}
67             </a>
68         @endif
69     </div>
70 </div>
71 </form>
72 </div>
73 </div>
74 </div>
75 </div>
76 </div>
77 @endsection
```

Gambar 20 (Kode Program Masuk)

```

50 @section('konten')
51 <div class="row clearfix">
52 <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
53 <div class="card">
54 <div class="header">
55 <h2>
56 DATA KARYAWAN
57 </h2>
58 </div>
59 <div class="body">
60 <div class="table-responsive">
61 <table class="table table-bordered table-striped table-hover dataTable js-exportable">
62 <thead>
63 <tr>
64 <th>No.</th>
65 <th>Nama</th>
66 <th>e-Mail</th>
67 <th>Asal Kota</th>
68 <th>Tanggal Lahir</th>
69 <th>Jenis Kelamin</th>
70 <th>Jurusan</th>
71 <th>Universitas</th>
72 <th>Nomor Handphone</th>
73 </tr>
74 </thead>
75
76 @php($i=1)
77 <tbody>
78 @foreach($query as $data)
79 <tr>
80 <td>{{ $i }}</td>
81 <td>{{ $data->name }}</td>
82 <td>{{ $data->email }}</td>
83 <td>{{ $data->asal_kota }}</td>
84 <td>{{ $data->tgl_lahir }}</td>
85 @if($data->jenis_kelamin==1)
86 <td>Laki-Laki</td>
87 @elseif($data->jenis_kelamin==2)
88 <td>Perempuan</td>
89 @endif
90 <td>{{ $data->jurusan }}</td>
91 <td>{{ $data->universitas }}</td>
92 <td>{{ $data->no_hp }}</td>
93 </tr>
94 @php($i++)
95 @endforeach
96 </tbody>
97 </table>
98 </div>
99 </div>
100 </div>
101 </div>
102 </div>
103 @endsection

```

Gambar 21 (Kode Program Melihat Data Karyawan)

```
57 @section('konten')
58 <div class="row">
59 <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
60 <div class="card">
61 <div class="header">
62 <h2>
63     DATA KRITERIA JABATAN
64 </h2>
65 <ul class="header-dropdown m-r--5">
66 <li class="dropdown">
67 <li><a href="/tambahKriteria" class="btn btn btn-success waves-effect">TAMBAH KRITERIA</a></li>
68 </li>
69 </ul>
70 </div>
71 <div class="body">
72 <table id="mainTable" class="table table-striped">
73 <thead>
74 <tr>
75 <th>No.</th>
76 <th>Jabatan</th>
77 <th>Kriteria</th>
78 </tr>
79 </thead>
80
81 @php($i=1)
82 <tbody>
83 @foreach($query as $data)
84 <tr>
85 <td>{{ $i }}</td>
86 <td>{{ $data->jabatan }}</td>
87 <td>
88 @php($kriteria=explode(',',$data->kriteria))
89 @foreach($kriteria as $krtr)
90 {{ $krtr }}<br/>
91 @endforeach
92 </td>
93 <td><a href="{{url('/kriteria/'.$data->id_jabatan.'/edit')}}"
94 class="btn btn-warning waves-effect">UBAH</a></td>
95 </tr>
96 @php($i++)
97 @endforeach
98 </tbody>
99 </table>
100 </div>
101 </div>
102 </div>
103 </div>
104 @endsection
```

Gambar 22 (Kode Program Mengelola Data Kriteria Jabatan)


```
52 @section('konten')
53
54 <div class="row clearfix">
55     <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
56         <div class="card">
57             <div class="header">
58                 <h2>
59                     HASIL PERHITUNGAN
60                 </h2>
61             </div>
62             <div class="body table-responsive">
63                 <table class="table">
64                     <thead>
65                         <tr>
66                             <th>No.</th>
67                             <th>Nama</th>
68                             <th>Nilai</th>
69                         </tr>
70                     </thead>
71                     <tbody>
72                         @foreach($query as $data)
73                             <tr>
74                                 <td>{{ $i }}</td>
75                                 <td>{{ $data->name }}</td>
76                                 <td>{{ $data->nilai }}</td>
77                             </tr>
78                         @php($i++)
79                     @endforeach
80                 </tbody>
81             </table>
82             <a href="/hasilPerhitungan" class="btn btn-danger waves-effect">KEMBALI </a>
83         </div>
84     </div>
85 </div>
86 </div>
87
88 @endsection
```

Gambar 23 (Kode Program Melihat Hasil)

```
56 @section('konten')
57
58 <div class="row clearfix">
59     <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
60         <div class="card">
61             <div class="header">
62                 <h2>
63                     DATA BOTOT KEPRIBADIAN
64                 </h2>
65             </div>
66             <div class="body table-responsive">
67                 <table class="table">
68                     <thead>
69                         <tr>
70                             <th>No.</th>
71                             <th>Kepribadian</th>
72                             <th>Botot Kepribadian</th>
73                         </tr>
74                     </thead>
75                     <tbody>
76                         @php($i=1)
77                         @foreach($query as $data)
78                             <tr>
79                                 <td>{{ $i }}</td>
80                                 <td>{{ $data->kepribadian }}</td>
81                                 <td>{{ $data->botot_kepribadian }}</td>
82                             </tr>
83                         @php($i++)
84                         @endforeach
85                     </tbody>
86                 </table>
87             </div>
88         </div>
89     </div>
90 </div>
91
92 @endsection
```

Gambar 24 (Kode Program Melihat Botot Kepribadian)

```

61 @section('konten')
62     <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
63         <div class="card">
64             <div id="carousel-example-generic" class="carousel slide" data-ride="carousel">
65                 <ol class="carousel-indicators">
66                     <li data-target="#carousel-example-generic" data-slide-to="0" class="active"></li>
67                     <li data-target="#carousel-example-generic" data-slide-to="1"></li>
68                     <li data-target="#carousel-example-generic" data-slide-to="2"></li>
69                     <li data-target="#carousel-example-generic" data-slide-to="3"></li>
70                     <li data-target="#carousel-example-generic" data-slide-to="4"></li>
71                 </ol>
72                 <div class="carousel-inner" role="listbox">
73                     <div class="item active">
74                         
75                     </div>
76                     <div class="item">
77                         
78                     </div>
79                     <div class="item">
80                         
81                     </div>
82                     <div class="item">
83                         
84                     </div>
85                     <div class="item">
86                         
87                     </div>
88                 </div>
89                 <a class="left carousel-control" href="#carousel-example-generic" role="button" data-slide="prev">
90                     <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
91                     <span class="sr-only">Previous</span>
92                 </a>
93                 <a class="right carousel-control" href="#carousel-example-generic" role="button" data-slide="next">
94                     <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
95                     <span class="sr-only">Next</span>
96                 </a>
97             </div>
98         </div>
99     </div>

```

```

101 <div class="row clearfix">
102     <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
103         <div class="card">
104             <div class="header">
105                 <h2>
106                     PT. Perkebunan Nusantara XII Kebun Kertowono
107                 </h2>
108             </div>
109             <div class="body">
110                 <p class="lead">
111                     Sejarah
112                 </p>
113                 <p>
114                     PTPN XII Kertowono ini berdiri sejak tahun 1910, untuk komoditas awal yakni kopi dan kina,
115                     namun pada saat era pemerintahan silih berganti, akhirnya ditetapkan komoditas andalanya berupa teh.
116                     Dengan luasan sekitar 2267,97 hektare, terdapat perkebunan teh serta pemandangan yang dapat memikat mata Anda.
117                     Begitu juga dengan aktifitas para pemetik teh disana menjadi suatu keindahan yang tak ada habisnya untuk dinik
118                     Kalau Kebun Teh Kertowono sendiri mempunyai kepekatan dan teh kita berwarna merah bata.
119                     PTPN XII Kertowono ini, merupakan bangunan tua, bisa dibilang alatnya berumur 1 abad lebih.
120                     PTPN XII Kertowono mempunyai alat mesin yang masih asli, jumlahnya sepaket mulai dari penggilingan hingga sort
121                     dan memiliki beberapa bagian pengelolaan mulai dari penerimaan, pelayuan, penggilingan, pengeringan, sortasi,
122                 </p>
123                 <p>
124                     Untuk bisa menuju ke Perkebunan Teh Kertowono ini Anda bisa melewati akses dari arah barat yakni,
125                     Surabaya ke timur ke Probolinggo dan menuju ke arah selatan jalur Jember, atau bisa melewati jalur Malang.
126                     Untuk mendapatkan teh asli Kertowono saat ini tidak sulit, karena sudah memasuki pemasaran di daerah Lumajang,
127                     serta Teh asal Kertowono ini sudah diekspor ke Eropa, Asia serta Timur Tengah, dan juga teh besutan PTPN XII m
128                     peringkat kedua menjadi teh terbaik di dunia setelah India.
129                     Awalnya kita mengelola teh ortodoks. Itu semacam teh hijau. Karena permintaan kurang, kita bikin teh CTC yang
130                     Mesinnya kita datangkan dari India. Kalau mesin ortodoks itu murni dari Belanda.
131                 </p>
132             </div>
133             <div class="header">
134                 <h2>
135                     Visi & Misi

```

```
136     </h2>
137   </div>
138   <div class="body">
139     <p class="lead">
140       Visi
141     </p>
142     <p>
143       "Menjadi Perusahaan Agribisnis yang berdaya saing tinggi dan mampu tumbuh-kembang berkelanjutan"
144     <p class="lead">
145       Misi
146     </p>
147     <p>
148       1. Melaksanakan reformasi bisnis, strategi, struktur, dan budaya perusahaan untuk mewujudkan profesionalisme
149       berdasarkan prinsip-prinsip good corporate governance.<br>
150       2. Meningkatkan nilai dan daya saing perusahaan (competitive advantage) melalui inovasi serta peningkatan
151       produktivitas dan efisiensi dalam penyediaan produk berkualitas dengan harga kompetitif dan pelayanan bermutu
152       dan stakeholders lainnya.<br>
153       3. Menghasilkan laba yang dapat membawa perusahaan tumbuh dan berkembang untuk meningkatkan nilai bagi shareh
154       sosial pada lingkungan usaha (community development).
155     </p>
156   </div>
157 </div>
158 </div>
159 </div>
160 @endsection
```

Gambar 25 (Kode Program Melihat Profil Perusahaan)

```
1 @extends('layouts.app')
2
3 @section('content')
4   <div class="container">
5     <div class="row justify-content-center">
6       <div class="col-md-8">
7         <div class="card">
8           <div class="card-header">{{ __('Register') }}</div>
9
10          <div class="card-body">
11            <form method="POST" action="{{ route('register') }}">
12              @csrf
13
14              <div class="form-group row">
15                <label for="name"
16                  class="col-md-4 col-form-label text-md-right">{{ __('Nama') }}</label>
17
18                <div class="col-md-6">
19                  <input id="name" type="text"
20                    class="form-control{{ $errors->has('name') ? ' is-invalid' : '' }}"
21                    name="name" value="{{ old('name') }}" required
22                    autofocus>
23
24                  @if ($errors->has('name'))
25                    <span class="invalid-feedback" role="alert">
26                      <strong>{{ $errors->first('name') }}</strong>
27                    </span>
28                  @endif
29                </div>
30              </div>
31
32              <div class="form-group row">
33                <label for="email"
34                  class="col-md-4 col-form-label text-md-right">{{ __('Alamat E-Mail ') }}</label>
35
36                <div class="col-md-6">
37                  <input id="email" type="email"
38                    class="form-control{{ $errors->has('email') ? ' is-invalid' : '' }}"
39                    name="email" value="{{ old('email') }}" required>
40                </div>
41              </div>
42            </form>
43          </div>
44        </div>
45      </div>
46    </div>
47  </div>
48 @endsection
```

```
40
41     @if ($errors->has('email'))
42         <span class="invalid-feedback" role="alert">
43             <strong>{{ $errors->first('email') }}</strong>
44             </span>
45     @endif
46 </div>
47 </div>
48
49 <div class="form-group row">
50     <label for="password"
51         class="col-md-4 col-form-label text-md-right">{{ __('Password') }}</label>
52
53     <div class="col-md-6">
54         <input id="password" type="password"
55             class="form-control{{ $errors->has('password') ? ' is-invalid' : '' }}"
56             name="password" required>
57
58         @if ($errors->has('password'))
59             <span class="invalid-feedback" role="alert">
60                 <strong>{{ $errors->first('password') }}</strong>
61             </span>
62         @endif
63     </div>
64 </div>
65 <div class="form-group row">
66     <label for="kota"
67         class="col-md-4 col-form-label text-md-right">{{ __('Asal Kota') }}</label>
68
69     <div class="col-md-6">
70         <input id="kota" type="text"
71             class="form-control{{ $errors->has('kota') ? ' is-invalid' : '' }}"
72             name="kota" value="{{ old('kota') }}" required
73             autofocus>
74 </div>
75 </div>
76
77 <div class="form-group row">
78     <label for="tglame"
79         class="col-md-4 col-form-label text-md-right">{{ __('Tanggal Lahir') }}</label>
80
81     <div class="col-md-6">
82         <div class="form-line" id="bs_datepicker_container">
83             <input type="text" class="form-control"
84                 placeholder="Masukkan Tanggal Lahir"
85                 name="tgl" id="datepicker">
86         </div>
87     </div>
88 </div>
89
90 <div class="form-group row">
91     <label for="gender"
92         class="col-md-4 col-form-label text-md-right">{{ __('Jenis Kelamin') }}</label>
93
94     <div class="col-sm-6">
95         <select class="form-control show-tick"
96             name="gender">
97             <option value="">-- Jenis Kelamin --</option>
98             <option value="1">Laki-Laki</option>
99             <option value="2">Perempuan</option>
100         </select>
101     </div>
102 </div>
```

```
100
101     <div class="form-group row">
102         <label for="jurusan"
103             class="col-md-4 col-form-label text-md-right">{{ __('Jurusan') }}</label>
104
105         <div class="col-md-6">
106             <input id="jurusan" type="text"
107                 class="form-control{{ $errors->has('jurusan') ? ' is-invalid' : '' }}"
108                 name="jurusan" value="{{ old('jurusan') }}"
109                 required autofocus>
110         </div>
111     </div>
112
113     <div class="form-group row">
114         <label for="universitas"
115             class="col-md-4 col-form-label text-md-right">{{ __('Universitas') }}</label>
116
117         <div class="col-md-6">
118             <input id="universitas" type="text"
119                 class="form-control{{ $errors->has('universitas') ? ' is-invalid' : '' }}"
120                 name="universitas"
121                 value="{{ old('universitas') }}" required
122                 autofocus>
123         </div>
124     </div>
125
126     <div class="form-group row">
127         <label for="hp"
128             class="col-md-4 col-form-label text-md-right">{{ __('No. Handphone') }}</label>
129
130         <div class="col-md-6">
131             <input id="hp" type="text"
132                 class="form-control{{ $errors->has('hp') ? ' is-invalid' : '' }}"
133                 name="hp"
134                 value="{{ old('hp') }}" required autofocus>
135         </div>
136     </div>
137
138     <div class="form-group row mb-0">
139         <div class="col-md-6 offset-md-4">
140             <button type="submit" class="btn btn-primary">
141                 {{ __('Register') }}
142             </button>
143         </div>
144     </div>
145 </form>
146 </div>
147 </div>
148 </div>
149 </div>
150 </div>
151 <script type="text/javascript">
152     jQuery(function ($) { // wait until the DOM is ready
153         $("#datepicker").datepicker({format: 'yyyy-mm-dd'});
154     });
155 </script>
156 @endsection
```

Gambar 26 (Kode Program Daftar)


```

42 @section('konten')
43 <div class="row clearfix">
44     <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
45         <div class="card">
46             <div class="header">
47                 <h2>
48                     SILAHKAN PILIH 3 DARI TIAP KRITERIA YANG SESUAI DENGAN PRIBADI ANDA
49                 </h2>
50             </div>
51             <div class="body">
52                 <h2 class="card-inside-title">Basic Examples</h2>
53                 <div class="demo-checkbox">
54                     <form method="post" action="{{route('hasil.store')}}">
55                         @method('put')
56                         @csrf()
57                         @php($i=1)
58                         @php($j=0)
59                         @php($aspek=["PIKIRAN", "ENERGI", "SIFAT", "STRATEGI", "IDENTITAS"])
60                         @foreach($query as $data)
61
62                             @if(in_array($i, [1, 7, 13, 19, 25]))
63
64                                 <h2 class="card-inside-title">ASPEK {{ $aspek[$j] }}</h2>
65
66                                 <div class="aspek-<?=$j+1 ?>">
67                                     @php($j++)
68                                     @endif
69                                     <input type="checkbox" id="checkbox-<?=$i ?>" name="checkbox-<?=$i ?>" value="{{! $data->id !!}}>
70                                     <label for="checkbox-<?=$i ?>">{!! $data->kriteria !!</label><br>
71                                     @php($i++)
72
73                                 @if(in_array($i, [7, 13, 19, 25, 31]))
74                                     </div>
75                                 @endif
76                             @endforeach
77
78                                     <button type="submit" class="btn btn-danger waves-effect">SUBMIT </button>
79                                 </form>
80                             </div>
81                         </div>
82                     </div>
83 </div>
84 <!-- #END# Checkbox -->
85
86
87 {{-- JANGAN DI HAPUS!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!--}}
88 {{-- <select id="kriteria" name="kriteria[]" multiple--}}
89 {{-- <optgroup label="Aspek Pikiran"--}}
90 {{-- @php($i=0)--}}
91 {{-- @foreach($query as $data)--}}
92 {{-- <option value="{{ $data->kriteria }}">{{ $data->kriteria }}</option--}}
93 {{-- @php($i++)--}}
94 {{-- @if($i=6)--}}
95 {{-- <optgroup label="Aspek Energi"></optgroup--}}
96 {{-- @elseif($i=12)--}}
97 {{-- <optgroup label="Aspek Sifat"></optgroup--}}
98 {{-- @elseif($i=18)--}}
99 {{-- <optgroup label="Aspek Strategi"></optgroup--}}
100 {{-- @elseif($i=24)--}}
101 {{-- <optgroup label="Aspek Identitas"></optgroup--}}
102 {{-- @endif--}}
103 {{-- @endforeach--}}
104 {{-- </optgroup--}}
105 {{-- </select--}}
106 {{-- <script--}}
107 {{-- $(document).ready(function () {--}}
108 {{-- $('#kriteria').select2({--}}
109 {{--     placeholder: 'Pilih Kriteria',--}}
110 {{--     maximumSelectionLength: 3,--}}
111 {{-- });--}}
112 {{-- });--}}
113 {{-- </script--}}
114 @endsection

```

Gambar 27 (Kode Program Pick Me)

```
38 @section('konten')
39   <div class="row">
40     <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
41       <div class="card">
42         <div class="header">
43           <h2>
44             HASIL PERHITUNGAN
45           </h2>
46           <ul class="header-dropdown m-r--5">
47             <li class="dropdown">
48           </ul>
49         </div>
50         <div class="body">
51           <table id="mainTable" class="table table-striped">
52             <thead>
53               <tr>
54                 <th>Nama</th>
55                 <th>Jabatan</th>
56                 <th>Nilai</th>
57               </tr>
58             </thead>
59             <tbody>
60               @foreach($query as $data)
61                 <tr>
62                   <td>{!! $data->name !!</td>
63                   <td>{!! $data->jabatan !!</td>
64                   <td>{!! $data->nilai !!</td>
65                 </tr>
66               @endforeach
67             </tbody>
68           </table>
69         </div>
70     </div>
71 @endsection
```

Gambar 28 (Kode Program Melihat Hasil)

```
46 @section('konten')
47     <div class="container-fluid">
48         <div class="row">
49             <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
50                 <div class="card">
51                     <div class="header">
52                         <h2>
53                             BIODATA KARYAWAN
54                         </h2>
55                         <ul class="header-dropdown m-r--5">
56                             <li class="dropdown">
57                                 <a href="{{route('biodata.edit', ['id'=>$query[0]->id_user])}}"
58                                     class="btn btn-warning waves-effect">UBAH</a></li>
59                         </ul>
60                     </div>
61                     <div class="body">
62                         <table class="table table-striped">
63                             <col width="50">
64                             <col width="250">
65                             <tbody>
66                                 <tr>
67                                     <td><strong>Nama</strong></td>
68                                     <td>: {{$query[0]->name}}</td>
69                                 </tr>
70                                 <tr>
71                                     <td><strong>e-Mail</strong></td>
72                                     <td>: {{$query[0]->email}}</td>
73                                 </tr>
74                                 <tr>
75                                     <td><strong>Asal Kota</strong></td>
76                                     <td>: {{$query[0]->asal_kota}}</td>
77                                 </tr>
```

```
78         <tr>
79             <td><strong>Tanggal Lahir</strong></td>
80             <td>: {{{query[0]->tgl_lahir}}}</td>
81         </tr>
82         <tr>
83             <td><strong>Jenis Kelamin</strong></td>
84             @if($query[0]->jenis_kelamin==1)
85                 <td>: Laki-Laki</td>
86             @else
87                 <td>: Perempuan</td>
88             @endif
89         </tr>
90         <tr>
91             <td><strong>Jurusan</strong></td>
92             <td>: {{{query[0]->jurusan}}}</td>
93         </tr>
94         <tr>
95             <td><strong>Universitas</strong></td>
96             <td>: {{{query[0]->universitas}}}</td>
97         </tr>
98         <tr>
99             <td><strong>No. Handphone</strong></td>
100            <td>: {{{query[0]->no_hp}}}</td>
101        </tr>
102    </tbody>
103 </table>
104 </div>
105 </div>
106 </div>
107 </div>
108 </div>
109 @endsection
```

```
39 @section('konten')
40 <div class="row clearfix">
41 <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
42 <div class="card">
43 <div class="header">
44 <h2>
45     UBAH BIODATA
46 </h2>
47 </div>
48 <div class="body">
49 <form method="post" action="{{route('biodata.update', ['id'=> $data->id_user])}}">
50     {{ method_field('PUT') }}
51     {{ csrf_field() }}
52     <label for="email_address">Asal Kota</label>
53     <div class="form-group">
54         <div class="form-line">
55             <input type="text" class="form-control" placeholder="Masukkan Asal Kota"
56                 value="{{ $data->asal_kota }}" name="kota">
57         </div>
58     </div>
59     <label for="password">Tanggal Lahir</label>
60     <div class="form-group">
61         <div class="form-line">
62             <input type="text" class="form-control" placeholder="Masukkan Tanggal Lahir"
63                 value="{{ $data->tgl_lahir }}" name="tgl" id="tgl">
64         </div>
65     </div>
66     <label for="password">Jurusan</label>
67     <div class="form-group">
68         <div class="form-line">
69             <input type="text" class="form-control" placeholder="Masukkan Jurusan"
70                 value="{{ $data->jurusan }}" name="jurusan">
71         </div>
72     </div>
73     <label for="password">Universitas</label>
74     <div class="form-group">
75         <div class="form-line">
76             <input type="text" class="form-control" placeholder="Masukkan Universitas"
77                 value="{{ $data->universitas }}" name="universitas">
78         </div>
79     </div>
80     <label for="password">No. Handphone</label>
81     <div class="form-group">
82         <div class="form-line">
83             <input type="text" class="form-control" placeholder="Masukkan No. Handphone"
84                 value="{{ $data->no_hp }}" name="hp">
85         </div>
86     </div>
87     <input type="hidden" value="{{ $data->id_user }}" name="id">
88     <input type="hidden" value="{{ $data->name }}" name="name">
89     <input type="hidden" value="{{ $data->email }}" name="email">
90     <br>
91     <button type="submit" class="btn btn-primary m-t-15 waves-effect">SIMPAN</button>
92 </form>
93 </div>
94 </div>
95 </div>
96 </div>
97 <script type="text/javascript">
98     jQuery(function ($) { // wait until the DOM is ready
99         $("#tgl").datepicker({format: 'yyyy-mm-dd'});
100     });
101 </script>
102 @endsection
```

Gambar 29 (Kode Program Mengelola Biodata)

E. Black Box Testing*Tabel 11 (Black Box Testing Fitur Data Karyawan)*

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat data karyawan	Main flow: klik menu data karyawan	Membuka halaman data karyawan	Halaman data karyawan terbuka	✓	

Tabel 12 (Black Box Testing Fitur Data Kriteria Jabatan)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat data kriteria jabatan	Main flow: klik menu data kriteria jabatan	Membuka halaman data kriteria jabatan	Halaman data kriteria jabatan terbuka	✓	
Tambah data kriteria jabatan	Main flow: klik tombol tambah	Membuka form jabatan baru	Form tambah jabatan baru ditampilkan	✓	

	Main flow: isi dan klik tombol simpan	Insert data jabatan baru ke database	Data jabatan baru berhasil ditambahkan ke database	✓	
	Alternatif flow: inputan kosong	Menampilkan pesan “Data belum diisi”	Pesan data harus diisi ditampilkan	✓	
Mengubah data kriteria jabatan	Main flow: klik tombol ubah pada data yang akan diubah	Membuka form ubah kriteria jabatan	Form ubah kriteria jabatan ditampilkan	✓	
	Main flow: isi dan klik tombol simpan	Insert data jabatan yang telah diubah ke database	Data jabatan baru berhasil diubah dan disimpan di database	✓	
	Alternatif flow: inputan kosong	Menampilkan pesan “Data belum diisi”	Pesan data belum diisi ditampilkan	✓	

	Alternatif flow: klik tombol batal	Menampilkan halaman data kriteria jabatan	Halaman data kriteria jabatan ditampilkan	✓	
--	--	--	--	---	--

Tabel 13 (Black Box Testing Fitur Hasil)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat hasil	Main flow: klik menu hasil	Membuka halaman hasil	Halaman hasil terbuka	✓	

Tabel 14 (Black Box Testing Fitur Bobot Kepribadian)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat bobot kepribadian	Main flow: klik menu bobot kepribadian	Membuka halaman bobot kepribadian	Halaman bobot kepribadian terbuka	✓	

Tabel 15 (Black Box Testing Fitur Profil Perusahaan)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat profil perusahaan	Main flow: klik menu profil perusahaan	Membuka halaman profil perusahaan	Halaman profil perusahaan terbuka	✓	

Tabel 16 (Black Box Testing Fitur Daftar)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Daftar	Main flow: klik menu daftar	Membuka halaman daftar	Halaman daftar terbuka	✓	
	Main flow: isi dan klik tombol simpan	Insert data baru ke database	Data baru berhasil ditambahkan ke database	✓	
	Alternatif flow:	Menampilkan pesan "Data	Pesan data belum	✓	

	inputan kosong	belum lengkap”	lengkap ditampilkan		
--	-------------------	-------------------	------------------------	--	--

Tabel 17 (Black Box Testing Fitur Pick Me)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Pick Me	Main flow: klik menu pick me	Membuka halaman pick me	Halaman pick me terbuka	✓	
	Main flow: pilih dan klik tombol next	Membuka halaman kepribadian	Halaman kepribadian terbuka	✓	
	Main flow: pilih dan klik tombol simpan	Insert data kepribadian yang telah dipilih ke database	Data kepribadian baru berhasil disimpan di database	✓	

Tabel 18 (Black Box Testing Fitur Melihat Hasil)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat hasil	Main flow: klik menu hasil	Membuka halaman hasil	Halaman hasil terbuka	✓	

Tabel 19 (Black Box Testing Fitur Biodata)

Nama Fitur	Flow	Hasil Yang Diharapkan	Hasil Sebenarnya	Kesimpulan	
				Sukses	Tidak
Melihat biodata	Main flow: klik menu biodata	Menampilkan halaman biodata	Halaman biodata terbuka	✓	
	Main flow: klik tombol ubah pada data yang akan diubah	Membuka form ubah biodata	Form ubah biodata ditampilkan	✓	

	Main flow: isi dan klik tombol simpan	Insert data biodata yang telah diubah ke database	Data biodata baru berhasil diubah dan disimpan di database	✓	
	Alternatif flow: inputan kosong	Menampilkan pesan "Data belum diisi"	Pesan data belum diisi ditampilkan	✓	
	Alternatif flow: klik tombol batal	Menampilkan halaman biodata	Halaman biodata ditampilkan	✓	