**Implementasi Metode *The Distance To The Ideal Alternative***

**Pada Sistem Informasi Perencanaan SDM**

**di PT. XYZ**

**TUGAS AKHIR**



**ADDINA NURIYANTI RAHMI**

**1112001012**

**FAKULTAS TEKNIK DAN ILMU KOMPUTER**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS BAKRIE**  
**JAKARTA**  
**2016**

**PERNYATAAN PEMBIMBING**

Yang bertanda tangan di bawah ini saya:

Pembimbing

Nama : Yusuf Lestanto, S.T., M.Sc.

Dengan ini menyatakan bahwa penelitian tugas akhir skripsi dengan judul:

“Implementasi Metode *The Distance To The Ideal Alternative* Pada Sistem Informasi Perencanaan SDM di PT.XYZ”

Yang disusun oleh:

Nama : Addina Nuriyanti Rahmi

NIM : 1112001012

Telah selesai dan siap untuk diajukan dan dipertahankan di hadapan sidang tugas akhir.

Demikian pernyataan ini dibuat untuk dapat dipergunakan sebagai syarat kelengkapan mengikuti sidang tugas akhir di Universitas Bakrie.

|  |
| --- |
| Jakarta, Agustus 2016  Yang menyatakan  Dosen Pembimbing  Yusuf Lestanto, S.T., M.Sc. |

# HALAMAN PENGESAHAN

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | Judul Penelitian | : | Implementasi Metode *The Distance To The Ideal Alternative* Pada Sistem Informasi Perencanaan SDM di PT.XYZ |
| 2. | Bidang Penelitian | : | Informatika |
| 3. | Peneliti Utama | : | Addina Nuriyanti Rahmi |
| 4. | Jenis Kelamin | : | Perempuan |
| 5. | Unit Kerja | : | Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Bakrie |
| 6. | Alamat/Telp. Unit kerja | : | Gelanggang Mahasiswa Soemantri Brojonegoro Suite GF-22. Jl. H.R. Rasuna Said Kav. C-22, Jakarta Selatan, Tel: 021-526 1448 Ext 243 |
| 7. | Alamat/Telp.Rumah | : | Jl. Kenari 1 No.16 RT 004 RW 03, Kelurahan Kenari, Kecamatan Senen, Jakarta Pusat. |
| 8. | Alamat e-mail | : | [addina.nuriyanti@gmail.com](mailto:addina.nuriyanti@gmail.com) |
| 9. | Telepon seluler | : | 08988159757 |
| 10. | Usulan Penelitian Tahun | : | 2016 |

|  |  |  |
| --- | --- | --- |
|  |  | Jakarta, |
| Menyetujui,  Dosen Pembimbing, |  | Peneliti, |
|  |  |  |
|  |  |  |
| Yusuf Lestanto, S.T., M.Sc. |  | Addina Nuriyanti Rahmi |

**Implementasi Metode *The Distance To The Ideal Alternative***

**Pada Sistem Informasi Perencanaan SDM di PT. XYZ**

**Addina Nuriyanti Rahmi**

# ABSTRAK

Pegawai, sebagai aset SDM perusahaan, merupakan salah satu elemen krusial yang menjaga berjalannya proses bisnis perusahaan. Rekrutmen merupakan sebuah proses penting yang dilakukan perusahaan untuk melakukan pencarian serta pengadaan SDM. Dalam proses ini terdapat beberapa situasi yang membutuhkan pengambilan keputusan, contohnya saat rekrutmen. Pada tahap awal, pengambilan keputusan dilakukan saat penyortiran CV dengan membandingkan kualifikasi pelamar dengan spesifikasi posisi. PT. XYZ merupakan sebuah perusahaan yang bergerak di bidang pembiayaan konsumen. Berdasarkan wawancara yang dilakukan dengan Manajer SDM di perusahaan, diketahui bahwa masih terdapat pegawai dengan ketidaksesuaian antara kualifikasi diri dan spesifikasi kebutuhan dari pekerjaan yang dilakukannya. Penelitian ini bertujuan untuk mengimplementasi metode DiA pada sistem informasi perencanaan SDM. Implementasi metode DiA dikhususkan pada modul rekrutmen. DiA akan melakukan penilaian pada pelamar berdasarkan beberapa kriteria yang digunakan dalam penyaringan lamaran. Kemudian, metode tersebut akan memberikan hasil penilaian dari masing-masing pelamar. Hasil penilaian ini dapat membantu staf SDM untuk membuat keputusan pemilihan pelamar yang lolos ke tahap selanjutnya. Pendekatan DiA akan diterapkan di dalam sistem tersebut dengan harapan dapat memberikan dukungan dalam pengambilan keputusan untuk penyaringan awal pelamar.

**Kata Kunci:** Rekrutmen, DiA, MADM*,* Perencanaan SDM, Sistem Penunjang Keputusan, Penyaringan Lamaran

**The Distance To The Ideal Alternative (DiA) Method Implementation On HR Planning Information System**

**In PT. XYZ**

**Addina Nuriyanti Rahmi**

# ABSTRACT

Manpower, as the enterprise’s HR assets, is one of the crucial elements which keep enterprise’s business process running. Recruitment is an important process enterprises do for searching and procuring human resources. In this process there are some situations that need decision-making, such as when doing recruitment. In the initial step, a decision-making happens when sorting CV by comparing the applicant’s qualification with the job specification. PT. XYZ is an enterprise which runs in a consumer finance field. Based on the interview with HR Manager of the company, it’s known that there are still some employees with mismatch between their personal qualification and the job specification. This research intends to implement DiA method into a HR Planning information system. The DiA method will be particularly implemented into recruitment module of the system. DiA will score the applicants based on some criterion that used in the application screening. After that, the method will provide the scoring result of each applicant. The scoring result may help HR staff to establish a decision when choosing the qualified applicant to the next step. DiA approach will be implemented to the system with expectation to support the decision-making in the application screening of the applicants.

**Keywords:** Recruitment, DiA, MADM*,* HR Planning, Decision Support System, Application Screening

**DAFTAR ISI**

[PERNYATAAN PEMBIMBING ii](#_Toc459865156)

[HALAMAN PENGESAHAN iii](#_Toc459865157)

[ABSTRAK iv](#_Toc459865158)

[ABSTRACT v](#_Toc459865159)

[DAFTAR ISI vi](#_Toc459865160)

[DAFTAR GAMBAR viii](#_Toc459865161)

[DAFTAR TABEL ix](#_Toc459865162)

[DAFTAR SINGKATAN x](#_Toc459865163)

[BAB 1 PENDAHULUAN 1](#_Toc459865164)

[1.1. Latar Belakang 1](#_Toc459865165)

[1.2. Identifikasi Masalah 4](#_Toc459865166)

[1.3. Rumusan Masalah 5](#_Toc459865167)

[1.4. Batasan Masalah 5](#_Toc459865168)

[1.5. Tujuan Penelitian 5](#_Toc459865169)

[1.6. Manfaat Penelitian 6](#_Toc459865170)

[1.7. Sistematika Penulisan 6](#_Toc459865171)

[BAB 2 TINJAUAN PUSTAKA 7](#_Toc459865172)

[2.1. Penelitian Terdahulu 7](#_Toc459865173)

[2.2. Perencanaan Sumber Daya Manusia 9](#_Toc459865174)

[2.3. Rekrutmen 11](#_Toc459865175)

[2.4. Metode DiA 13](#_Toc459865176)

[2.5. *Rapid Application Development* (RAD) 16](#_Toc459865177)

[BAB 3 METODOLOGI PENELITIAN 18](#_Toc459865178)

[3.1 Kerangka Kerja Penelitian 18](#_Toc459865179)

[3.2 Metode Pengumpulan Data 19](#_Toc459865180)

[3.2.1 Studi Literatur 19](#_Toc459865181)

[3.2.2 Wawancara 19](#_Toc459865182)

[3.3 Metode Pengembangan Sistem dan Penyusunan Laporan 20](#_Toc459865183)

[3.3.1 Perencanaan Kebutuhan 20](#_Toc459865184)

[3.3.2 Perancangan Sistem 24](#_Toc459865185)

[3.3.3 Pembangunan Sistem 24](#_Toc459865186)

[3.3.4 Implementasi 24](#_Toc459865187)

[3.3.5 Penyusunan Laporan 25](#_Toc459865188)

[3.4 Rencana Kegiatan 25](#_Toc459865189)

[BAB 4 IMPLEMENTASI DAN PEMBAHASAN 26](#_Toc459865190)

[4.1 Perancangan Sistem 26](#_Toc459865191)

[4.1.1 *Use Case Diagram* 26](#_Toc459865192)

[4.1.2 *Activity Diagram* 28](#_Toc459865193)

[4.1.3 *Class Diagram* 29](#_Toc459865194)

[4.1.4 *Data Model* 30](#_Toc459865195)

[4.2 Implementasi 31](#_Toc459865196)

[4.2.1 Implementasi Metode DiA 31](#_Toc459865197)

[4.2.2 Implementasi Antarmuka 44](#_Toc459865198)

[4.3 Pengujian 49](#_Toc459865199)

[4.3.1 Pengujian *White Box* 49](#_Toc459865200)

[4.3.2 Pengujian *Black Box* 49](#_Toc459865201)

[BAB 5 SIMPULAN DAN SARAN 49](#_Toc459865202)

[5.1 Simpulan 51](#_Toc459865203)

[5.2 Saran 51](#_Toc459865204)

[DAFTAR PUSTAKA 52](#_Toc459865205)

[Lampiran 1 – Wawancara 53](#_Toc459865206)

[Lampiran 2 – *Software Requirement Specification* (SRS) 58](#_Toc459865207)

[Lampiran 3 – Rencana Kegiatan 82](#_Toc459865208)

**DAFTAR GAMBAR**

[Gambar 2.1 Proses Perencanaan SDM (Mathis & Jack, 2010) 10](#_Toc453082217)

[Gambar 2.2 Fase RAD (Kendall & Kendall, 2011)](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082218) 17

[Gambar 3.1 Alur Pelaksanaan Penelitian 18](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082219)

[Gambar 3.2 Diagram *Use Case* 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082220)2

[Gambar 3.3 Diagram Aktivitas Proses Pemeringkatan Kualifikasi Pelamar 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082220)4

[Gambar 3.4 *Class Diagram* Sistem 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082220)5

[Gambar 3.5 *Conceptual Data Model* 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082220)6

[Gambar 3.6 *Logical Data Model* 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082220)8

[Gambar 3.7 *Physical Data Model* 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082220)9

**DAFTAR TABEL**

Tabel 2.1 Tabel Representasi Masalah MADM (Tran & Boukhatem, 2008) 14

Tabel 2.2 Tabel Perbandingan Metode MADM 15

[Tabel 3.1 Deskripsi *Use Case* Modul Rekrutmen 2](file:///D:\Δ%20Dokumen%20Δ\random\Personal\skripsi\Implementasi%20Metode%20Weighted%20Product%20pada%20Sistem%20Perencanaan%20SDM.docx#_Toc453082218)3

**DAFTAR SINGKATAN**

SDM Sumber Daya Manusia

RAD *Rapid Application Development*

KSA *Knowledge, Skills,* dan *Abilities*

CV *Curriculum Vitae*

DiA *The Distance To The Ideal Alternative*

WP *Weighted Product*

SAW *Simple Additive Weighting*

TOPSIS *Technique for Order Preference By Similarity To Ideal Solution*

MADM *Multi-Attribute Decision Making*

HR *Human Resource*

MPP *Manpower Planning*

# BAB 1 PENDAHULUAN

## Latar Belakang

Pegawai, sebagai aset SDM perusahaan, merupakan salah satu elemen krusial yang menjaga berjalannya proses bisnis perusahaan. Tanpa adanya pegawai, perusahaan belum mampu melaksanakan proses bisnis dengan baik karena ia laksana roda yang menggerakkan perusahaan. Selain sebagai penggerak perusahaan, aset SDM dapat menciptakan kompetensi inti (*core competencies*) bagi perusahaan yang membedakan dirinya dengan pesaing lainnya. Beberapa contohnya yang dipengaruhi oleh keberhasilan SDM ialah produktivitas, kualitas/pelayanan, kemampuan pegawai, serta operasi inovatif (Mathis & Jack, 2010).

Rekrutmen merupakan sebuah proses penting yang dilakukan perusahaan untuk melakukan pencarian serta pengadaan SDM. Dalam proses ini terdapat beberapa situasi yang membutuhkan pengambilan keputusan. Salah satu situasi yang melibatkan pengambilan keputusan saat rekrutmen ialah ketika melakukan seleksi pelamar. Ketika sebuah perusahaan mengumumkan pembukaan rekrutmen, sejumlah pelamar akan mengirimkan aplikasi lamaran terhadap posisi tertentu. Biasanya pelamar akan menyertakan CV / data diri mereka pada aplikasi lamaran. Dibutuhkan pengambilan keputusan untuk menentukan siapa pelamar yang akan dipanggil untuk melanjutkan ke tahap seleksi yang lebih ketat. Pada tahap awal, pengambilan keputusan dilakukan saat penyortiran CV dengan membandingkan kualifikasi pelamar dengan spesifikasi posisi. Proses ini dilakukan dengan mempertimbangkan bahwa untuk dapat sepenuhnya mampu mendukung perusahaan, dibutuhkan SDM dengan kualitas baik dan talenta untuk mencapai tujuan perusahaan (Armstrong & Taylor, 2014). Maka dari itu, proses pencarian dan penyeleksian bibit-bibit penggerak perusahaan ini sebaiknya dilakukan dengan perencanaan serta eksekusi yang baik agar perusahaan mendapatkan orang yang tepat.

Selain untuk mendapatkan orang yang tepat, perencanaan dalam rekrutmen juga dapat memperkecil kemungkinan terjadinya kegagalan rekrutmen. Kegagalan

dalam proses rekrutmen dapat mengakibatkan masalah serius bagi perusahaan (Chack, 2013). Salah satu akibat gagalnya proses rekrutmen membuat biaya yang telah dikeluarkan oleh perusahaan untuk penyelenggaraan rekrutmen menjadi sia-sia. Perusahaan juga gagal memenuhi kebutuhan atas pegawai yang mungkin sangat mereka perlukan saat itu. Hal tersebut dapat berimbas pada penurunan produktivitas perusahaan yang akan menghambat kesuksesan bisnis (Bulmash, Chhinzer, & Speers, 2010). Selain itu, kegagalan rekrutmen juga dapat menyebabkan kesalahan sasaran pemilihan kandidat pegawai sehingga perusahaan gagal mendapatkan kandidat pegawai yang berkualifikasi sesuai standar. Tentu saja hal ini bukanlah sesuatu yang diharapkan perusahaan karena ketidakcocokan antara KSAs pekerja dengan kualifikasi pekerjaan dapat berujung pada *turnover* pegawai. *Turnover* merupakan istilah yang merujuk pada tingkat pegawai yang berhenti dan meninggalkan perusahaan (Armstrong & Taylor, 2014) (Bulmash, Chhinzer, & Speers, 2010).

Perencanaan SDM dapat menjadi solusi dalam membantu perusahaan merencanakan rekrutmen dengan baik. Perencanaan SDM juga membantu perusahaan mendapatkan SDM yang “tepat”. Tepat yang dimaksud di sini adalah jumlah orang yang tepat, dengan kemampuan yang tepat, pada waktu dan tempat yang tepat. Kegiatan ini mencakup analisa dan identifikasi ketersediaan serta kebutuhan akan SDM dalam perusahaan untuk jangka waktu panjang di masa yang akan datang (Mathis & Jack, 2010). Oleh karena itu, perencanaan SDM sangat erat hubungannya dengan rekrutmen, seleksi, serta pemeliharaan keahlian pegawai. Apabila di kemudian hari perusahaan menghadapi situasi darurat yang berhubungan dengan pengambilan keputusan terkait SDM, perusahaan dapat menghadapinya secara sigap dengan berpatokan kepada perencanaan SDM yang telah dibuat sebelumnya.

PT. XYZ merupakan sebuah perusahaan yang bergerak di bidang pembiayaan konsumen. Perusahaan ini telah memiliki sekitar 4700 pegawai bekerja pada kurang lebih 160 cabang yang tersebar di seluruh Indonesia (Lampiran 1). Berdasarkan wawancara yang dilakukan dengan Manajer SDM di perusahaan, diketahui bahwa masih terdapat pegawai dengan ketidaksesuaian antara kualifikasi diri dan spesifikasi kebutuhan dari pekerjaan yang dilakukannya. Hal ini dikenal dengan deviasi. Sesungguhnya, deviasi merupakan hal yang sebisa mungkin dihindari oleh perusahaan dalam mempekerjakan pegawai (Armstrong & Taylor, 2014). Namun, pada kenyataannya deviasi masih ditemui dalam perusahaan. Contoh kasus deviasi adalah ketika spesifikasi kebutuhan dari posisi IT *Programmer* mengharuskan lulusan S1, sedangkan pegawai yang dipekerjakan masih berstatus lulusan SMA. Hal ini menandakan bahwa perusahaan masih kesulitan untuk mendapatkan orang yang tepat dalam proses rekrutmennya.

Perencanaan SDM yang tepat akan menghasilkan proses rekrutmen yang sukses. Dalam kasus ini, perusahaan mungkin mampu menyusun perencanaan SDM di atas kertas. Namun, untuk penerapannya terutama pada proses rekrutmen belum begitu berhasil. Hal ini ditunjukkan dengan adanya deviasi. Sebenarnya perusahaan memiliki pilihan untuk menanggulangi deviasi dengan melaksanakan program pengembangan karyawan. Tetapi, apabila perekrutan kandidat dengan KSA di bawah kualifikasi pekerjaan terus terjadi, investasi waktu serta biaya yang disisihkan perusahaan untuk pengembangan karyawan akan membesar akibat akumulasi. Pada akhirnya, hasil dari pilihan tersebut hanya menyebabkan kerugian bagi perusahaan.

Di awal telah dijelaskan sebelumnya bahwa dibutuhkan SDM yang berkualitas sesuai dengan kebutuhan perusahaan untuk meningkatkan performa perusahaan. Hal ini membuat proses penyortiran CV menjadi sama pentingnya dengan proses lain dalam rekrutmen. Proses pengambilan keputusan dalam penyortiran CV akan menentukan daftar awal pelamar dengan kualifikasi yang sesuai terhadap spesifikasi kebutuhan pekerjaan dan mengeliminasi pelamar tidak sesuai kualifikasi (deviasi) (Information and Communications Technology Council, 2012). Hingga saat ini, proses penyortiran CV yang dilakukan perusahaan masih manual, yaitu dengan mengandalkan pengamatan mata untuk membandingkan kualifikasi pelamar dengan kualifikasi yang dibutuhkan posisi tersebut. Proses ini membutuhkan waktu maksimal 5 menit untuk membandingkan satu CV. Sedangkan jumlah CV yang diterima perusahaan dalam proses rekrutmen dapat mencapai rata-rata 250 berkas per bulan (Lampiran 1).

Dalam penelitian yang dilakukan oleh (Chack, 2013) menyebutkan bahwa sistem penunjang keputusan memiliki fungsi penting dalam perusahaan. Penerapan sistem ini akan meningkatkan daya guna serta budaya perusahaan. *The Distance To The Ideal Alternative* (DiA)merupakan salah satu pendekatan yang dapat diterapkan dalam sistem penunjang keputusan. Pendekatan ini telah terbukti memiliki performa yang lebih baik dari pada SAW (Purnamasari, 2015). Penerapan metode ini bisa menjadi jalan keluar bagi perusahaan dalam mendukung pembuatan keputusan pemilihan pelamar pada tahap penyortiran CV yang sesuai dengan spesifikasi pekerjaan. DiA akan melakukan penilaian pada pelamar berdasarkan beberapa kriteria yang digunakan dalam penyaringan lamaran. Kemudian, metode tersebut akan memberikan hasil penilaian dari masing-masing pelamar. Hasil penilaian ini dapat membantu staf SDM untuk membuat keputusan pemilihan pelamar yang lolos ke tahap selanjutnya, yaitu wawancara.

Berdasarkan penjabaran masalah di atas, diperlukan adanya sebuah sistem yang mampu menilai kecocokan kualifikasi pelamar dalam proses penyaringan lamaran. Oleh karena itu, penelitian yang berjudul “Implementasi Metode *The Distance To The Ideal Alternative* (DiA)Pada Sistem Informasi Perencanaan SDM di PT. XYZ” akan dilakukan. Penelitian bertujuan untuk mengimplementasi metode DiA pada sistem informasi perencanaan SDM. Implementasi metode DiA dikhususkan pada modul rekrutmen untuk melakukan penilaian dan pemeringkatan pelamar yang memiliki kesesuaian kualifikasi diri dengan spesifikasi pekerjaan.

## Identifikasi Masalah

Dari uraian pada latar belakang dapat diidentifikasi beberapa masalah yang dihadapi oleh perusahaan, antara lain adalah :

1. Perusahaan mempekerjakan pegawai dengan kualifikasi yang tidak sesuai dengan spesifikasi pekerjaan yang telah ditentukan.
2. Proses penyaringan lamaran pada perusahaan masih dilakukan secara manual.
3. Perusahaan mengalami kesulitan memantau kesesuaian antara kualifikasi pelamar dengan kualifikasi standar pada posisi tertentu selama proses penyaringan lamaran.

## Rumusan Masalah

Penelitian ini dilakukan dengan berpatokan pada pertanyaan sebagai berikut:

1. Bagaimana implementasi metode DiA dalam modul rekrutmen untuk memberikan rekomendasi pelamar pada tahap seleksi administratif?
2. Bagaimana implementasi modul rekrutmen sistem informasi perencanaan SDM dengan menerapkan model RAD?

## Batasan Masalah

Adapun batasan masalah dalam pelaksanaan penelitian ini ialah :

1. Sistem yang dibangun merupakan aplikasi berbasis *web.*
2. Metode DiA diterapkan pada modul rekrutmen untuk memberikan pemeringkatan serta rekomendasi pelamar.
3. Penyeleksian lamaran calon pegawai dengan metode DiAdilakukan berdasarkan pada kriteria umur, tingkat pendidikan, jenis kelamin, pengalaman kerja, kemampuan, dan gaji yang diharapkan.
4. Pengalaman kerja yang dimasukkan ke dalam sistem hanya pengalaman yang memiliki kaitan dengan posisi yang dilamar.
5. Penelitian ini dibatasi hingga tahap pengujian aplikasi.

## Tujuan Penelitian

Adapun tujuan dilaksanakannya penelitian ini ialah :

1. Mengimplementasi metode DiA ke dalam sistem informasi perencanaan SDM.
2. Mengimplementasi modul rekrutmen dalam sistem informasi perencanaan SDM dengan menerapkan model RAD.

## Manfaat Penelitian

Adapun manfaat dari dilaksanakannya penelitian ini ialah :

1. Mempermudah akses atas informasi yang dibutuhkan dalam rekrutmen melalui basis data terpusat pada sistem informasi perencanaan SDM.
2. Mempermudah pengelolaan informasi terkait rekrutmen, yaitu penyimpanan dan pencarian data lowongan pekerjaan, pelamar, serta aplikasi lamaran yang diajukan ke perusahaan.
3. Membantu pengambilan keputusan dalam proses penyaringan lamaran calon pegawai melalui implementasi metode DiA.
4. Penelitian ini dapat menjadi referensi bagi penulis lain yang ingin mengimplementasikan metode DiA dalam sistemnya.

## Sistematika Penulisan

Sistematika penulisan penelitian ini dibagi dalam 5 (lima) bab. Berikut adalah penjabaran dari sistematika penulisan dalam penelitian ini :

**BAB I PENDAHULUAN**

Latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat dalam penelitian ini akan dijelaskan pada bab ini.

**BAB II TINJAUAN PUSTAKA**

Landasan teori membahas beberapa dasar-dasar teori yang mendukung penulisan skripsi ini. Penelitian terdahulu menjelaskan beberapa penelitian yang terkait dengan rekrutmen, rancang bangun sistem, dan metode DiA.

**BAB III METODOLOGI PENELITIAN**

Bab ini menjelaskan daftar tahapan dalam alur pelaksanaan penelitian.

**BAB IV IMPLEMENTASI DAN PEMBAHASAN**

Pada bab ini akan membahas penerapan model *RAD* yang digunakan dalam pengembangan sistem serta penjabaran mengenai penerapan metode DiA untuk memberikan rekomendasi pelamar pada tahap seleksi administratif penyaringan pelamar.

**BAB V SIMPULAN DAN SARAN**

Simpulan menjelaskan hasil dari penelitian. Saran, sesuai dengan maknanya, berisi beberapa saran yang dapat diterapkan untuk penelitian selanjutnya.

# BAB 2 TINJAUAN PUSTAKA

1. Penelitian Terdahulu

Penelitian ini dilaksanakan dengan tujuan untuk mengimplementasi metode DiA dalam sistem informasi perencanaan SDM, khususnya modul rekrutmen. Beberapa penelitian yang bersinggungan dengan topik ini telah dilakukan, di antaranya ialah sebagai berikut :

1. Penelitian berjudul “Penentuan Penerima Beasiswa Dengan Metode *Simple Additive Weighting* (SAW)dan Metode *The Distance To The Ideal Alternative* (DiA)*”* oleh Rachmah Ayu Purnamasari*.* Penelitian ini dimaksudkan untuk menjawab permasalahan terkait penentuan penerima beasiswa dengan mengimplementasi dua buah metode, yaitu SAW dan DiA. Penilaian penerima beasiswa dilakukan terhadap 15 mahasiswa dengan menilai 3 kriteria. Kriteria yang digunakan dalam penilaian ialah nilai IPK, jumlah penghasilan orang tua, dan jumlah saudara kandung. Pengujian dilakukan untuk mengetahui metode yang lebih sesuai dalam kasus ini. Sebuah program dibangun dengan menggunakan *software* MATLAB untuk mempermudah proses pengujian. Pengujian yang disebut uji sensitivitas menunjukkan bahwa metode DiA lebih efektif daripada metode SAW (Purnamasari, 2015).
2. Penelitian berjudul “Uji Sensitivitas Metode WP, SAW, dan TOPSIS Dalam Menentukan Titik Lokasi *Repeater* *Internet Wireless*” oleh David Ahmad Effendy dan Rony Heri Irawan. Penelitian ini mengangkat masalah seringnya terjadi pendirian *repeater* pada lokasi yang kurang tepat oleh penyedia jasa jaringan internet. Penelitian mengusung tiga buah metode untuk memecahkan masalah pemilihan lokasi yang tepat untuk pendirian jaringan internet, yaitu WP*,* SAW*,* dan TOPSIS. Penilaian dilakukan berdasarkan 6 kriteria, yaitu jarak dengan pemancar, tingkat halangan, kepadatan pemukiman penduduk, perizinan pendirian *repeater*, kebutuhan internet, gangguan sinyal dan keamanan. Metode tersebut melewati uji sensitivitas untuk melihat metode mana yang paling cocok. Hasil dari penelitian adalah metode WP, SAW, dan

TOPSIS dapat diimplementasikan dengan cepat dan akurat. Hasil pengujian menunjukkan bahwa metode yang paling relevan dengan kasus ini ialah SAW dengan nilai perubahan terbesar dibandingkan dengan metode WP dan TOPSIS. Nilai perubahan SAW, TOPSIS, dan WP berturut-turut adalah 9%, 0.38%, dan 0,10% (Effendy & Irawan, 2015).

1. Penelitian berjudul “Sistem Pendukung Keputusan Rekrutmen Karyawan Produksi Menggunakan Metode *Weighted Product* Pada PT. Ploss Asia Semarang” oleh Ardi Kusumaning Diah R. Penelitian dilatarbelakangi oleh pengambilan keputusan yang sering kali dipengaruhi oleh subjektivitas. Oleh karena itu, penelitian dilakukan bertujuan untuk membuat sebuah sistem dengan implementasi metode WP untuk mempermudah pemilihan karyawan sesuai dengan kriteria. Kriteria yang digunakan untuk penilaian terdiri dari penampilan (fisik dan busana), kepribadian, pengalaman kerja, motivasi kerja, ambisi dalam pekerjaan, kemampuan kerja sama, kemampuan berkomunikasi, dan potensi untuk berkembang. Metode pengembangan sistem menerapkan model *waterfall* dengan bahasa pemrograman *Microsoft Visual Basic* 6.0 dan *MySQL* sebagai basis data. Sistem berbasis *desktop* tersebut mampu memberikan rekomendasi dan pertimbangan dalam memutuskan penerimaan kandidat. Metode WP berhasil di implementasi (Diah R., 2013).

Berdasarkan uraian di atas, dapat dirangkum beberapa perbedaan antara penelitian ini dengan penelitian terdahulu. Perbedaan yang ada dengan penelitian oleh (Purnamasari, 2015) ialah dari segi bidang penerapannya. Penelitian yang akan dilakukan membahas penerapan metode DiA pada penyaringan lamaran di proses rekrutmen dalam perusahaan, sedangkan (Purnamasari, 2015) menerapkan metode DiA dan SAW pada penentuan penerima beasiswa. Dengan perbedaan pada bidang penerapannya, otomatis kriteria yang digunakan oleh kedua penelitian juga berbeda. Kemudian, sistem yang dibangun oleh (Purnamasari, 2015) berbasis *desktop*, sedangkan sistem dari penelitian ini berbasis *web.*

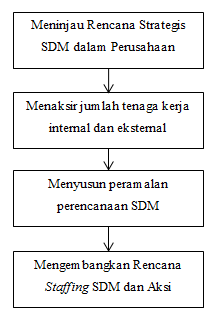
Penelitian yang dilakukan oleh (Effendy & Irawan, 2015) membahas penerapan metode SAW, WP, dan TOPSIS untuk menentukan lokasi pendirian *repeater wireless.* Di lain sisi, penelitian yang akan dilakukan menerapkan metode DiA dalam proses rekrutmen. Kemudian, metode dan kriteria yang digunakan dalam penilaian juga berbeda.

Penelitian yang dilakukan oleh (Diah R., 2013) membahas penerapan metode WP untuk penerimaan pegawai. Bidang penerapan penelitian tersebut sama dengan penelitian ini, namun metode yang digunakan berbeda. Penelitian ini menerapkan metode DiA untuk melakukan penilaian. Kemudian kriteria yang digunakan untuk penilaian juga memiliki perbedaan. Kriteria yang oleh (Diah R., 2013) terdiri dari penampilan (fisik dan busana), kepribadian, pengalaman kerja, motivasi kerja, ambisi dalam pekerjaan, kemampuan kerja sama, kemampuan berkomunikasi, dan potensi untuk berkembang. Berbeda dengan penelitian oleh (Diah R., 2013), penelitian yang akan dilakukan menggunakan kriteria umur, tingkat pendidikan, jenis kelamin, pengalaman kerja, kemampuan, dan gaji yang diharapkan.

1. Perencanaan Sumber Daya Manusia

Sebagai salah satu kegiatan inti dalam manajemen SDM, perencanaan SDM merupakan sebuah proses untuk menganalisis dan mengidentifikasi kebutuhan serta ketersediaan sumber daya manusia dalam sebuah perusahaan. Proses ini dilakukan dengan tujuan untuk mendapatkan tenaga kerja dengan jumlah, kemampuan, serta pada waktu dan tempat yang tepat (Armstrong & Taylor, 2014) (Mathis & Jack, 2010). Maka dari itu, Perumusan perencanaan SDM akan membantu perusahaan untuk menjawab pertanyaan seperti berapa banyak, kemampuan apa saja, kapan, dan di mana, terkait SDM dan kebutuhan serta ketersediaannya dalam perusahaan.

1. **Meninjau Rencana Strategis SDM Perusahaan.** Dalam pelaksanaannya, proses perencanaan SDM melalui beberapa tahapan dapat dilihat pada Gambar 2.1. Pada gambar tersebut terlihat bahwa proses perencanaan SDM dimulai dengan melakukan peninjauan terhadap rencana strategis perusahaan. Tujuan dari rencana strategis perusahaan merupakan hal utama yang mendasari serta membentuk perencanaan SDM (Armstrong & Taylor, 2014).



Gambar 2.1 Proses Perencanaan SDM (Mathis & Jack, 2010)

1. **Menaksir Jumlah Tenaga Kerja Internal dan Eksternal.** Setelah peninjauan rencana strategis perusahaan dilakukan, penaksiran jumlah tenaga kerja internal dan eksternal akan dilaksanakan. Kegiatan ini mencakup identifikasi kondisi SDM internal dan eksternal perusahaan. Ada beberapa hal yang perlu ditinjau ketika akan melakukan identifikasi kondisi SDM internal, beberapa di antaranya ialah daftar pekerjaan yang ada saat itu, jumlah individu yang bertugas melaksanakan pekerjaan tersebut, dan KSA yang dibutuhkan untuk menyelesaikan pekerjaan tersebut. Sedangkan untuk mengidentifikasi kondisi SDM eksternal perusahaan perlu diperhatikan beberapa hal seperti faktor ekonomi, politik, dan pemerintahan.
2. **Menyusun Peramalan Perencanaan SDM.** Pada tahap ini dilakukan identifikasi kebutuhan serta ketersediaan SDM perusahaan. Berdasarkan data taksiran jumlah tenaga kerja internal dan eksternal, dapat diperoleh berapa banyak kebutuhan perusahaan akan tenaga kerja. Selain itu, ketersediaan tenaga kerja dalam perusahaan juga akan diketahui.
3. **Mengembangkan Rencana *Staffing* SDM dan Aksi.** Selanjutnya, akan dikembangkan sebuah rencana *staffing* SDM. *Staffing* SDM merupakan kegiatan menyangkut merekrut, menyeleksi, dan mempertahankan pegawai (Mathis & Jack, 2010). Rencana ini dirancang untuk membantu mengarahkan perusahaan dalam menghadapi hal-hal tidak terduga terkait kepegawaian dalam perusahaan. Salah satu contoh dari hal yang tidak terduga adalah pengunduran diri karyawan potensial di masa kritis perusahaan.
4. Rekrutmen

Rekrutmen adalah sebuah proses mencari dan mengajak tenaga kerja potensial untuk melamar ke lowongan pekerjaan yang dibuka oleh perusahaan (Armstrong & Taylor, 2014) (Mathis & Jack, 2010). Rekrutmen memiliki hubungan yang erat dengan perencanaan SDM. Pada perencanaan SDM dirumuskan strategi untuk pelaksanaan rekrutmen, sehingga untuk pelaksanaan rekrutmen sendiri dapat merujuk pada formulasi perencanaan SDM.

Rekrutmen terdiri dari beberapa tingkatan, yaitu menentukan kebutuhan, menarik kandidat, menyaring lamaran, wawancara, pengujian, menilai kandidat, pengecekan referensi, pengecekan lamaran, penawaran kerja, *follow-up* (Armstrong & Taylor, 2014).

1. **Menentukan Kebutuhan.** Pada tingkatan ini dilakukan analisis kebutuhan terkait spesifikasi pekerjaan (*job specification*). Spesifikasi pekerjaan akan dilihat berdasarkan pengetahuan, kemampuan dan keahlian, pengalaman, pelatihan, atau kebutuhan khusus jika memang diperlukan perusahaan.
2. **Menarik Kandidat.** Sebelum bisa melakukan seleksi kandidat, tentunya diperlukan adanya sekumpulan orang yang melamar lowongan pekerjaan yang dipasang oleh perusahaan terlebih dahulu. Ada beberapa hal yang dapat dilakukan perusahaan untuk menarik kandidat, yaitu dengan menyebutkan kelebihan yang dimiliki perusahaan (contoh : penawaran keuntungan jika menjadi pegawai) dan melakukan promosi lowongan kerja melalui media o*nline* atau cetak.
3. **Menyaring Lamaran.** Setelah promosi lowongan kerja, orang-orang yang tertarik akan memasukkan CV mereka untuk melamar lowongan tersebut. Dari seluruh lamaran yang masuk akan dilakukan penyaringan untuk mendapatkan daftar-pendek (*shortlisted*) calon kandidat untuk diwawancara. Lamaran disaring dengan melakukan komparasi antara informasi kandidat dan spesifikasi pekerjaan yang telah dibuat sebelumnya. Umumnya, daftar-pendek kandidat hanya terdiri dari empat hingga delapan orang. Jika penyaringan menghasilkan lebih dari delapan orang, maka sebaiknya dilakukan penyaringan ulang dengan menggunakan kriteria yang lebih ketat.
4. **Wawancara.** Wawancara merupakan salah satu metode seleksi saat rekrutmen. Hal ini dilakukan untuk mengukur sejauh apa kualifikasi pelamar cocok dengan spesifikasi pekerjaan.
5. **Pengujian.** Pada tahap ini, pelamar diberikan ujian yang dapat berupa ujian psikologi, kecerdasan, atau kemampuan. Dengan dilakukan pengujian akan menghasilkan informasi objektif mengenai bidang yang diujikan terhadap pelamar.
6. **Menilai Kandidat.** Penilaian kandidat dilakukan berdasarkan pada wawancara dan pengujian yang telah dilakukan kepada pada pelamar.
7. **Pengecekan Referensi.** Pengecekan referensi memiliki maksud untuk mencari tahu informasi faktual atas pegawai prospektif terkait masa kerjanya di perusahaan terdahulu. Beberapa pertanyaan yang perlu diajukan ialah lama masa kerja, apa jabatannya, pekerjaan apa yang dilaksanakan, besaran gaji, jumlah hari ketidakhadiran setahun terakhir, dan maukan perusahaan tersebut untuk mempekerjakan kembali pegawai prospektif ini.
8. **Pengecekan Lamaran.** Survei yang dilakukan oleh CareerBuilder terhadap 2.811 manajer SDM dari seluruh bagian Amerika Serikat menunjukkan bahwa 58% manajer menemukan kebohongan dalam resume yang diterima (Career Builder, 2014). Maka dari itu diperlukan pengecekan lamaran untuk menghindari hal ini.
9. **Penawaran Kerja.** Setelah didapatkan kandidat yang cocok, akan dilakukan penawaran kerja serta penandatanganan kontrak.
10. ***Follow-Up.*** *Follow-up* dilakukan untuk memastikan bahwa pegawai yang telah direkrut mampu beradaptasi dan untuk mengecek seberapa baik kerja mereka.

Pada tingkatan penyaringan lamaran, dilakukan perbandingan kualifikasi pelamar dengan spesifikasi pekerjaan yang ditentukan pada tahapan penentuan kebutuhan. Pada tingkat ini, PT. XYZ masih mengalami masalah dimana mereka belum mampu mengetatkan penyaringan lamaran. Hal itu menyebabkan perusahaan mempekerjakan pegawai dengan kualifikasi yang tidak sesuai dengan spesifikasi pekerjaan. Salah satu alasan mengapa hal tersebut bisa terjadi adalah karena sulitnya mendapatkan orang yang tepat untuk posisi tersebut. Sedangkan, mempekerjakan tenaga kerja dengan kualifikasi yang tidak sesuai spesifikasi pekerjaan sebaiknya dihindari (Armstrong & Taylor, 2014).

Penelitian ini dilakukan dengan bertujuan salah satunya untuk mengimplementasikan metode DiA pada tingkatan penyaringan lamaran. Implementasi metode ini diharapkan mampu memberikan daftar pelamar yang cocok dengan spesifikasi pekerjaan.

1. Metode DiA

Metode *The Distance To The Ideal Alternative* merupakan salah satu pendekatan yang dapat digunakan untuk memecahkan permasalahan pemilihan berdasarkan banyak kriteria dan banyak alternatif atau biasa disebut dengan permasalahan MADM (Tran & Boukhatem, 2008). Pendekatan metode DiA bertujuan untuk menentukan alternatif terbaik dari sekumpulan alternatif yang ada. Pada penelitian ini, alternatif ialah para pelamar. Penilaiannya dilakukan berdasarkan beberapa kriteria yang dibutuhkan. Metode ini dibangun dengan mempertimbangkan beberapa kekurangan yang dimiliki oleh metode TOPSIS, yaitu keakuratan dan abnormalitas ranking.

Permasalahan MADM dapat direpresentasikan ke dalam sebuah tabel untuk mempermudah pemahaman pada Tabel 2.1. Dapat dilihat pada Tabel 2.1 terdiri dari 3 elemen, yaitu A, C, dan w. Elemen A merupakan sejumlah alternatif yang merepresentasikan masing-masing pelamar. Elemen C merupakan sejumlah atribut yang merepresentasikan kriteria penilaian penyaringan lamaran. Elemen W merupakan kepentingan relatif / bobot dari masing-masing atribut.

**Tabel 2.1 Tabel Representasi Masalah MADM (Tran & Boukhatem, 2008)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | C1  (w1) | C2  (w2) | ... | Cm  (wm) |
| A1 | x11 | x12 | ... | x1m |
| A2 | x21 | x22 | ... | x2m |
| ... | ... | ... | ... | ... |
| An | xn1 | xn2 | ... | xnm |

Berikut adalah langkah-langkah utama dalam implementasi metode DiA (Tran & Boukhatem, 2008; Purnamasari, 2015):

1. Buat matriks keputusan X.
2. Buat matriks keputusan R ternormalisasi. Matriks ini terdiri dari elemen rij.

Nilai masing-masing rij bisa didapatkan dengan rumus jarak *Euclidean* :

1. Buat matriks normalisasi keputusan V terbobot. Matriks ini terdiri dari elemen vij.

Nilai masing-masing vij bisa didapatkan dengan rumus :

1. Tentukan nilai solusi ideal positif dan negatif dari tiap atribut.

* Untuk kriteria *benefit*
* Untuk kriteria *cost*

1. Hitung jarak *Manhattan* dari nilai atribut positif dan negatif.
2. Tentukan “*Positive Ideal Alternative*” (PIA).
3. Hitung jarak alternatif ke PIA dengan rumus :
4. Lakukan pemeringkatan alternatif. Semakin kecil nilai Ri menandakan alternatif tersebut lebih disarankan.

Selain metode DiA terdapat beberapa metode lain untuk menyelesaikan masalah MADM seperti, TOPSIS, SAW, dan WP. Pada Tabel 2.2 terdapat perbandingan antara metode TOPSIS, SAW, WP, dan DiA.

Tabel 2.2 Tabel Perbandingan Metode MADM

|  |  |  |
| --- | --- | --- |
| **Metode** | **Formula Akhir** | **Karakteristik** |
| TOPSIS |  | Memiliki konsep dasar bahwa alternatif terbaik ialah yang memiliki jarak terpendek relatif ke solusi ideal (Tran & Boukhatem, 2008). |
| SAW |  | Memiliki konsep dasar yaitu mencari penjumlahan terbobot dari nilai atribut seluruh alternatif. Pembobotan dilakukan dengan mengalikan nilai terhadap bobot kriteria (Effendy & Irawan, 2015; Tran & Boukhatem, 2008) |
| WP |  | Memiliki konsep dasar yaitu mencari perkalian terbobot dari nilai atribut seluruh alternatif. Pembobotan dilakukan dengan memangkatkan nilai dengan bobot kriteria (Tran & Boukhatem, 2008; Effendy & Irawan, 2015). |
| DiA |  | Memiliki konsep dasar bahwa alternatif terbaik adalah yang memiliki jarak terdekat dengan alternatif ideal positif (Tran & Boukhatem, 2008). |

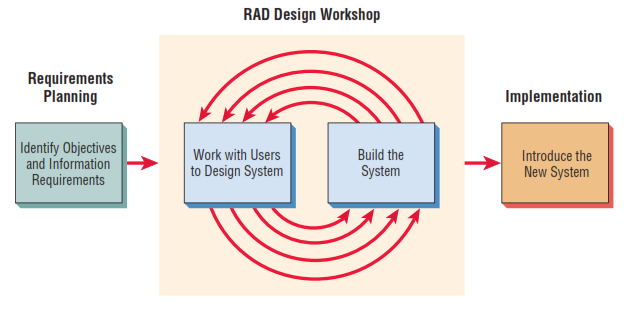
Meski terdapat begitu banyak metode MADM, penelitian ini akan mengusung metode DiA untuk diterapkan dalam sistem perencanaan SDM. Pemilihan ini didasari oleh penelitian yang telah dilakukan (Tran & Boukhatem, 2008) dan (Purnamasari, 2015). Pengujian yang dilakukan oleh (Tran & Boukhatem, 2008) menunjukkan bahwa metode DiA mengalahkan TOPSIS dalam segi abnormalitas ranking dan memiliki akurasi yang lebih baik dari pada SAW dan WP. Kemudian penelitian yang dilakukan oleh (Purnamasari, 2015) menunjukkan bahwa DiA memiliki sensitivitas yang lebih tinggi dari pada SAW.

1. *Rapid Application Development* (RAD)

Dalam membangun sebuah sistem informasi dibutuhkan sebuah metode pengembangan sistem agar pengembangan dapat dilakukan dengan baik. Metode pengembangan sistem yang digunakan akan berbeda tergantung beberapa situasi. RAD merupakan salah satu metode pengembangan sistem yang mampu memperpendek waktu pembangunan sistem. Maka dari itu, RAD cocok diterapkan dalam sebuah proyek pengembangan sistem dengan ketersediaan waktu yang sangat terbatas (Kendall & Kendall, 2011). Model RAD mengenal beberapa fase, yaitu *Requirements Planning*, RAD *Design Workshop*, dan *Implementation* (Kendall & Kendall, 2011).

Diawali dengan *Requirements Planning* atau perencanaan kebutuhan, dilaksanakan identifikasi akan kebutuhan dari sistem informasi yang akan dibangun. Daftar kebutuhan didapatkan dengan berdasar pada tujuan dibuatnya sistem. Kegiatan tersebut dilakukan oleh sistem analis serta calon pengguna.

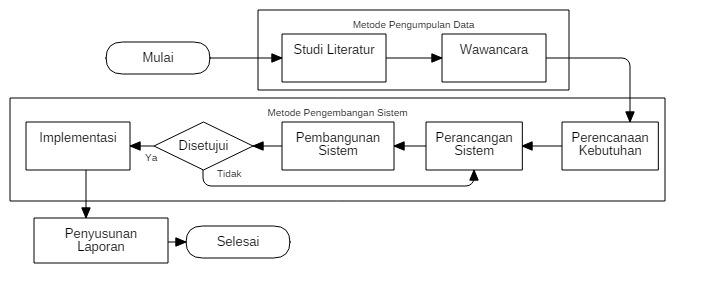
Setelah kebutuhan sistem informasi diketahui, dilakukan fase kedua yang dinamakan RAD *Design Workshop.* Tahap ini terdiri dari dua bagian, yaitu perancangan sistem dan pembangunan sistem. Seperti yang dapat dilihat pada Gambar 2.2, terdapat beberapa panah yang berputar berulang kali menunjuk kepada bagian perancangan sistem dan pembangunan sistem secara bergantian. Panah yang berputar berulang tersebut menandakan bahwa terdapat iterasi antara perancangan dan pembangunan sistem. Proses iterasi ini merupakan poin utama yang berperan penting dalam pemendekan waktu pembuatan sistem. Selama fase *design workshop* ini, calon pengguna akan memberikan tanggapan atas *prototype* yang telah dibangun berdasarkan kebutuhan calon pengguna. Perbaikan yang disarankan calon pengguna akan di akomodasi oleh pengembang untuk pembuatan *prototype* selanjutnya. Kegiatan ini akan terus berulang hingga terdapat persetujuan antara pihak analis dan calon pengguna.

Pada fase terakhir terdapat fase implementasi. Pada akhirnya dalam fase ini dibangun sebuah atau sebagian sistem yang sebenarnya, tidak lagi berbentuk *prototype*. Pengujian sistem dilakukan pada tahap ini. Setelah lolos pengujian, sistem akan diperkenalkan ke para pengguna untuk kemudian digunakan (Kendall & Kendall, 2011).

Gambar 2.2 Fase RAD (Kendall & Kendall, 2011)

# BAB 3 METODOLOGI PENELITIAN

1. **Kerangka Kerja Penelitian**

Penelitian ini akan dilakukan secara bertahap dengan alur seperti yang dapat dilihat pada Gambar 3.1.

Metode pengumpulan data yang diterapkan dalam penelitian adalah studi literatur dan wawancara. Deskripsi kegiatan dari studi literatur dan wawancara akan dipaparkan pada subbab Metode Pengumpulan Data tepat setelah subbab ini. Untuk pengembangan sistem, penelitian mengadopsi metode pengembangan RAD yang terdiri dari empat tahapan, yaitu perencanaan kebutuhan, perancangan sistem, pembangunan sistem, dan implementasi. Pada Gambar 3.1, tahapan metode RAD dikelompokkan menjadi satu ke dalam bagian Metode Pengembangan Sistem. Pemaparan kegiatan pada tiap tahapan RAD serta tahap penyusunan laporan akan dilakukan pada subbab berjudul Metode Pengembangan Sistem dan Penyusunan Laporan.

Gambar 3.1 Alur Pelaksanaan Penelitian

* 1. **Metode Pengumpulan Data**

Data yang akan digunakan dalam penelitian ini didapatkan melalui dua cara, yaitu studi literatur dan wawancara.

* + 1. **Studi Literatur**

Dalam studi literatur dilakukan pencarian dan penyaringan referensi. Kegiatan ini menghasilkan daftar referensi yang akan digunakan untuk memperkuat analisis dalam penelitian ini. Referensi yang digunakan adalah referensi dengan topik-topik yang berhubungan dengan penelitian yang akan dilakukan. Sumber referensi dapat berupa buku, artikel dari internet, serta jurnal dan *paper* penelitian. Selain untuk mencari referensi sebagai penunjang analisis, studi literatur juga dilakukan untuk mencari penelitian-penelitian terdahulu yang telah dilakukan oleh orang lain.

* + 1. **Wawancara**

Dalam wawancara dilakukan tanya-jawab dengan pihak yang memiliki pengetahuan berkaitan masalah yang sedang dialami perusahaan, yaitu Manajer *Human Capital* di PT. XYZ. Wawancara dilakukan bertujuan untuk mendapatkan gambaran terkini kondisi perusahaan terkait rekrutmen dan penyaringan lamaran. Dari wawancara diketahui bahwa pelaksanaan rekrutmen perusahaan dibedakan menjadi 3, yaitu untuk Cabang, Kantor Pusat, dan *Management Trainee*. Rata-rata jumlah orang yang melamar ke perusahaannya tiap bulan sebanyak 250 orang. Penyaringan lamaran atau seleksi administratif yang dilakukan perusahaan masih menggunakan cara manual.

Dalam tahap perancangan serta pembangunan sistem, diskusi dilakukan dengan Manajer IT serta Manajer *Human Capital* untuk mengumpulkan data yang diperlukan. Hasil wawancara dapat dilihat pada Lampiran 1, sedangkan hasil diskusi yang dilakukan pada penelitian ini tidak didokumentasikan.

* 1. **Metode Pengembangan Sistem dan Penyusunan Laporan**

Data yang telah dikumpulkan akan digunakan sebagai patokan dalam mengembangkan sistem. Pada penelitian ini, pengembangan sistem menerapkan pendekatan model RAD. Model RAD terdiri dari beberapa fase dengan iterasi di dalamnya. Model ini dipilih sebagai model pengembangan sistem karena karakteristiknya yang sesuai dengan keadaan di perusahaan. Keadaan memaksa perusahaan untuk tidak dapat menunggu lama hingga sistem selesai dibangun.

Subbab berikut ini akan memaparkan kegiatan dilakukan pada tiap-tiap tahapan RAD serta penyusunan laporan.

* + 1. **Perencanaan Kebutuhan**

Penelitian ini dilakukan sejak awal tahun 2016. Pada tahap awal ini dilakukan diskusi untuk mencari tahu apa tujuan dibuatnya sistem serta informasi apa saja yang akan digunakan di dalamnya. Dengan merujuk pada tujuan dibuatnya sistem, dapat disusun daftar kebutuhan fungsional dan non-fungsional sistem. Kebutuhan fungsional dan non-fungsional sistem disertakan pada dokumen SRS di lampiran 2.

Kebutuhan informasi sistem dibedakan menjadi dua, yaitu kebutuhan informasi secara umum dan kebutuhan informasi untuk Metode DiA. Metode DiA membutuhkan tiga elemen utama untuk melakukan penilaian, yaitu alternatif, kriteria, dan kepentingan relatif masing-masing atribut.

1. **Kebutuhan Informasi Secara Umum**

Eratnya kaitan sistem dengan proses rekrutmen mengandung arti bahwa sebagian besar informasi yang terdapat dalam sistem merupakan data yang memiliki hubungan dengan rekrutmen. Kebutuhan informasi sistem secara umum didapatkan dengan mempertimbangkan kebutuhan fungsional dari sistem yang telah disebutkan. Berikut merupakan daftar kebutuhan informasi sistem secara umum :

* *Job Title* : Informasi daftar jabatan pegawai dalam perusahaan.
* Posisi : Informasi daftar posisi dalam perusahaan.
* *Requirement* : Informasi kebutuhan kualifikasi tiap *Job Title* dan Posisi.
* Kantor : Informasi daftar kantor yang dimiliki perusahaan.
* Lowongan Kerja: Informasi daftar posisi yang sedang lowong pada kantor.
* Pelamar : Informasi daftar orang yang melamar ke perusahaan.

1. **Kebutuhan Informasi Metode DiA**

Permasalahan dalam penyaringan lamaran merupakan salah satu masalah dengan banyak kriteria dan banyak alternatif (MADM). Dalam penyaringan lamaran, sekumpulan lamaran yang diterima perusahaan akan dinilai terhadap beberapa kriteria tertentu dari perusahaan. Hasil penilaian digunakan untuk menentukan pelamar yang layak melanjutkan ke tahap berikutnya. Hingga saat ini perusahaan melakukan penyaringan lamaran dengan cara manual, yaitu mengandalkan pengamatan mata untuk membandingkan kualifikasi pelamar dengan kualifikasi posisi. Hal ini cukup memakan waktu karena proses *screening* sebuah CV membutuhkan waktu hingga 5 menit.

Penelitian ini menerapkan metode DiA ke dalam sistem untuk membantu pelaksanaan proses penyaringan lamaran. Sistem akan menggunakan model matematis dari metode DiA untuk melakukan penilaian masing-masing pelamar. Setelah nilai didapatkan, maka akan dilakukan pemeringkatan. Melalui pemeringkatan tersebut dapat dilihat daftar pelamar yang lebih unggul dari yang lain berdasarkan penilaian sistem. Mereka adalah daftar pelamar yang direkomendasikan oleh sistem untuk lanjut ke tahap berikutnya. Namun, perlu diperjelas bahwa sistem hanya sekedar memberikan rekomendasi pelamar. Pengambilan keputusan tetaplah berada di bawah kuasa pengambil keputusan.

Kebutuhan informasi dari sistem, khususnya untuk implementasi metode DiA, didapatkan melalui diskusi yang dilakukan bersama Manajer *Human Capital* perusahaan. Berikut merupakan beberapa kriteria yang dibutuhkan dalam penilaian :

* 1. Jenis Kelamin () data jenis kelamin pelamar akan dibandingkan dengan kualifikasi jenis kelamin dari posisi yang dilamar. Penilaian dilakukan dengan melihat kesesuaian kualifikasi tersebut. Hasil yang sesuai akan menjadi prioritas untuk direkomendasikan. Tabel 3.1 menunjukkan konversi nilai untuk kriteria ini.

**Tabel 3.1 Nilai Jenis Kelamin**

|  |  |
| --- | --- |
| Hasil Perbandingan | Nilai |
| Sesuai | 1 |
| Tidak Sesuai | 0 |

* 1. Usia () data usia pelamar akan dibandingkan dengan kualifikasi usia dari posisi yang dilamar. Penilaian dilakukan dengan melihat kesesuaian kualifikasi tersebut. Hasil yang sesuai akan menjadi prioritas untuk direkomendasikan. Tabel 3.2 menunjukkan konversi nilai untuk kriteria ini.

**Tabel 3.2 Nilai Usia**

|  |  |
| --- | --- |
| Hasil Perbandingan | Nilai |
| Sesuai | 1 |
| Selisih 1 Tahun | 0,75 |
| Lainnya | 0 |

* 1. Pendidikan () data pendidikan terakhir pelamar akan dibandingkan dengan kualifikasi pendidikan terakhir dari posisi yang dilamar. Nilai pendidikan yang lebih tinggi akan diprioritaskan. Tabel 3.3 menunjukkan konversi nilai untuk kriteria ini.

**Tabel 3.3 Nilai Pendidikan**

|  |  |
| --- | --- |
| Nilai Pendidikan | Nilai |
| S3 | 100 |
| S2 | 90 |
| S1 | 80 |
| D3 | 70 |
| D2 | 60 |
| D1 | 50 |
| SMA | 40 |
| SMP | 30 |

* 1. Pengalaman Kerja () data pengalaman kerja pelamar akan dibandingkan dengan kualifikasi pengalaman kerja dari posisi yang dilamar. Nilai pengalaman kerja yang lebih tinggi akan diprioritaskan. Karena nilai kriteria ini bersifat kuantitatif dan tidak memiliki batasan tertentu seperti pada usia, maka data akan disajikan sebagaimana adanya tanpa konversi nilai.
  2. Kemampuan () data kemampuan pelamar akan dibandingkan dengan kualifikasi kemampuan dari posisi yang dilamar. Penilaian dilakukan dengan melihat kesesuaian kualifikasi tersebut. Hasil yang sesuai akan menjadi prioritas untuk direkomendasikan. Tabel 3.4 menunjukkan konversi nilai untuk kriteria ini.

**Tabel 3.4 Nilai Kemampuan**

|  |  |
| --- | --- |
| Hasil Perbandingan | Nilai |
| Sesuai | 1 |
| Sesuai sebanyak *n* | 1/*n* |

* 1. Gaji yang Diharapkan () Nilai yang lebih rendah akan diprioritaskan pada kriteria ini. Karena nilai kriteria ini bersifat kuantitatif dan tidak memiliki batasan tertentu seperti pada usia, maka data akan disajikan sebagaimana adanya tanpa konversi nilai.

Masing-masing kriteria yang telah dijelaskan di atas memiliki nilai prioritas. Nilai prioritas tersebut diukur dalam skala, yaitu tidak penting, kurang penting, normal, penting, dan sangat penting. Nilai prioritas akan menentukan nilai bobot atau kepentingan relatif kriteria tersebut. Nilai prioritas kriteria dapat disesuaikan oleh pengguna untuk mengantisipasi apabila seiring berjalannya waktu terjadi perubahan tingkat kepentingan masing-masing kriteria dalam pelaksanaan penyaringan lamaran. Berikut merupakan bobot dari masing-masing kriteria berdasarkan nilai prioritas yang diterapkan :

**Tabel 3.5 Bobot Kriteria**

|  |  |  |
| --- | --- | --- |
| Kriteria Penilaian | Tingkat Kepentingan | Bobot |
| Jenis Kelamin (C1) | Kurang Penting | 2 |
| Usia (C2) | Sangat Penting | 5 |
| Pendidikan (C3) | Penting | 4 |
| Pengalaman (C4) | Normal | 3 |
| Kemampuan (C5) | Sangat Penting | 5 |
| Gaji yang Diharapkan (C6) | Normal | 3 |

* + 1. **Perancangan Sistem**

Setelah perencanaan kebutuhan selesai dilakukan, selanjutnya adalah perancangan sistem. Hasil yang didapatkan dari perencanaan kebutuhan akan digunakan sebagai dasar perancangan sistem. Selanjutnya akan dibuat diagram *use case, activity diagram, class diagram,* serta rancangan basis data. Hal ini dilakukan untuk mempermudah proses pemahaman akan alur kerja dan struktur dasar dari sistem. Untuk melaksanakan perancangan sistem, diskusi dilakukan bersama dengan Manajer IT. Perencanaan sistem dilakukan berulang kali. Hal ini bertujuan untuk memastikan bahwa rancangan sistem sudah cocok dengan kebutuhan pengguna.

* + 1. **Pembangunan Sistem**

Dengan didasarkan pada rancangan yang telah dibuat, sebuah *prototype* dari sistem akan dibuat. *Prototype* merupakan sebuah sistem bayangan yang dibangun dengan tujuan untuk merepresentasikan sistem yang sebenarnya. Namun, yang lebih ditekankan pada *prototype* ialah dari segi fungsional di mana ia harus mampu melaksanakan fungsi utama dari sistem yang sesungguhnya. Setelah *prototype* dibangun, tahap perencanaan sistem kembali dilakukan. Hal ini bertujuan untuk memberi gambaran pada pengguna, seperti apa sistem yang akan dibangun nantinya. Pengguna melakukan evaluasi terhadap *prototype* yang telah dibangun di tahap Pembangunan Sistem. Apabila tidak ada perubahan yang diusulkan oleh calon pengguna, maka sistem yang sebenarnya dibuat dengan berdasarkan pada *prototype* yang disetujui.

* + 1. **Implementasi**

Pada tahap ini dilakukan pembangunan sistem sesungguhnya berdasarkan pada *prototype* yang disetujui. Pengujian sistem juga dilakukan pada tahap ini. Pengujian sistem dilakukan dengan tujuan untuk memastikan bahwa sistem bekerja sesuai dengan yang diharapkan. Penyelidikan *error* atau *bug* pada sistem juga berpeluang besar ditemukan melalui pengujian. Pada penelitian ini, akan dilakukan dua pengujian yaitu *black-box* dan *white-box.*

* + 1. **Penyusunan Laporan**

Hasil yang didapatkan dari tahap sebelumnya akan digunakan sebagai bahan penulisan laporan. Analisa akan dilakukan terhadap implementasi metode DiA pada sistem*.*

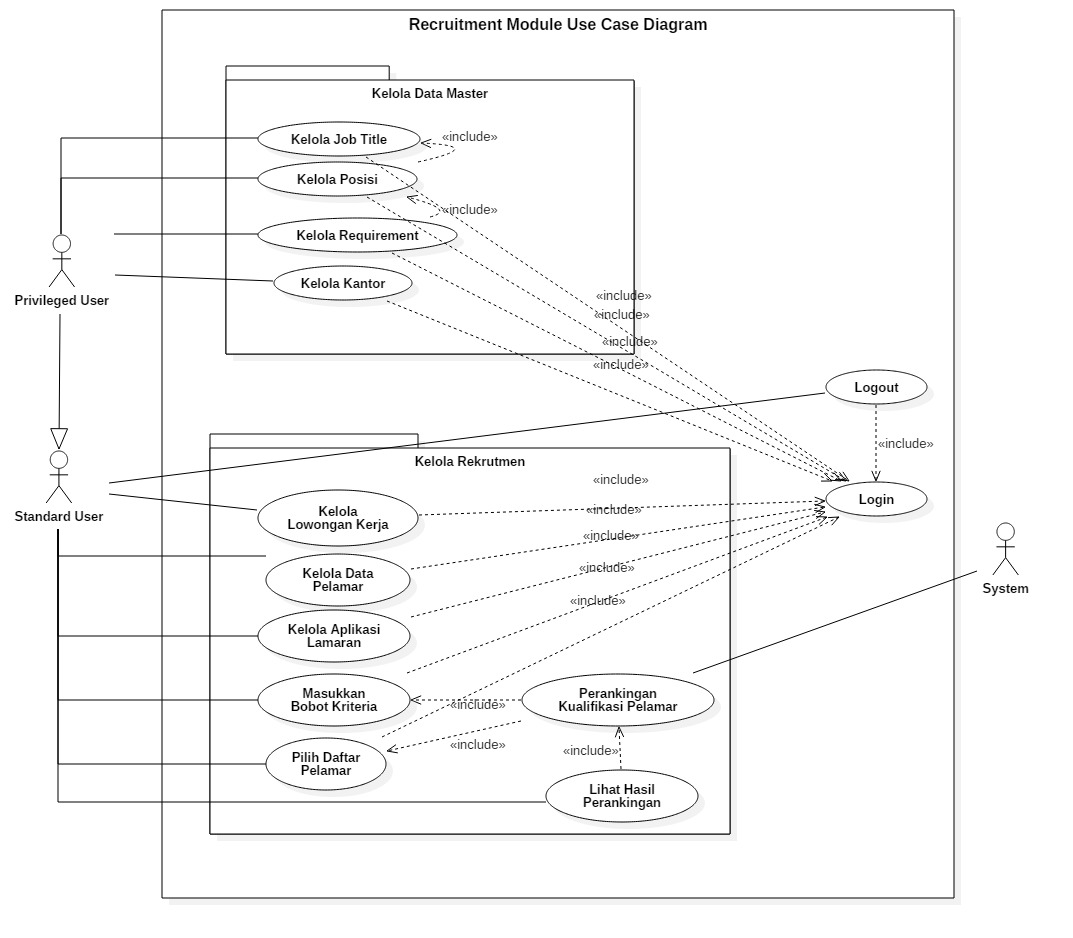
* 1. **Rencana Kegiatan**

Penelitian ini dilakukan sejak awal tahun 2016, tepatnya pada bulan Januari. Untuk lebih jelas, rencana kegiatan dituangkan dalam bentuk gambar yang dapat ditemukan pada Lampiran 3.

# BAB 4 IMPLEMENTASI DAN PEMBAHASAN

* 1. **Perancangan Sistem**

Pada penelitian ini, rancangan sistem tersusun dari beberapa elemen seperti *use case diagram, activity diagram, class diagram,* dan rancangan basis data. Pemaparan tiap elemen dapat ditemukan pada subbab setelah ini secara berurutan.

* + 1. ***Use Case Diagram***

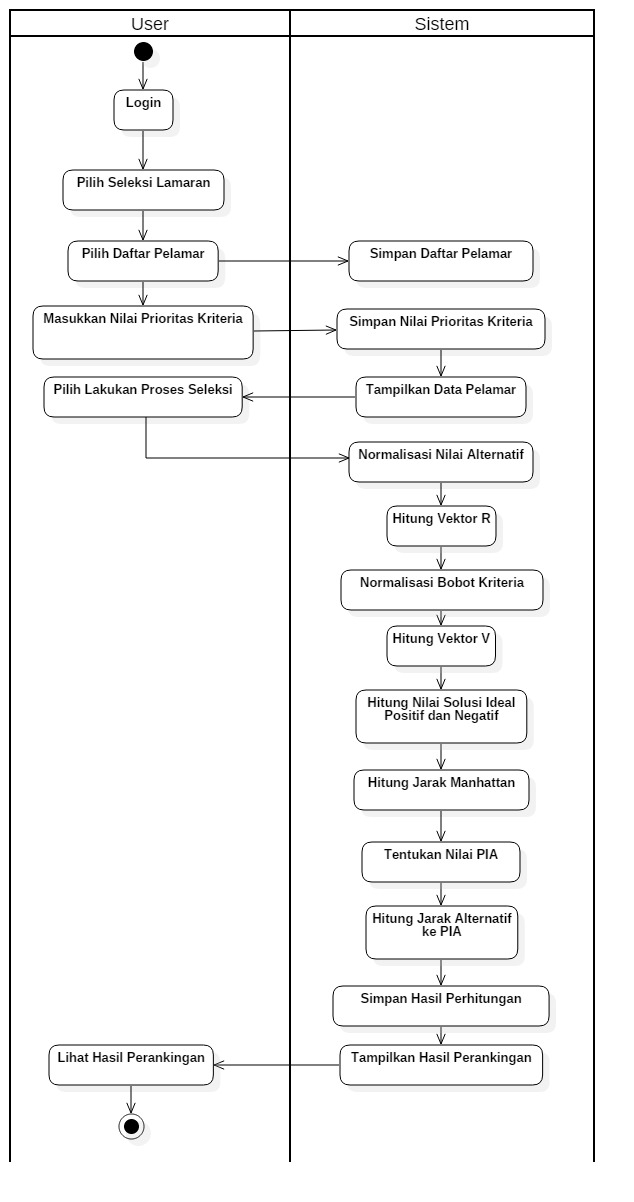
Gambar 4.1 Diagram *Use Case*

Di atas terdapat Gambar 4.1 yang merupakan diagram *Use Case* dari sistem. Terdapat dua jenis pengguna, yaitu *privileged user* dan *standard user. Privileged user* merupakan seorang admin sistem yang dapat mengelola master data seperti *jobtitle*, posisi, *requirement* dari posisi, dan kantor. Sedangkan *standard user* tidak dapat mengelola master data, namun hanya mengelola lowongan kerja, data pelamar, dan aplikasi lamaran serta melihat hasil pemeringkatan pelamar. Deskripsi dari *use case* dapat dilihat pada Tabel 4.1.

**Tabel 4.1 Deskripsi *Use Case* Modul Rekrutmen**

|  |  |
| --- | --- |
| Nama *Use Case* | Deskripsi |
| Kelola *Job Title* | *Use case* ini menggambarkan kegiatan pengelolaan data *Job Title*. |
| Kelola Posisi | *Use case* ini menggambarkan kegiatan pengelolaan data Posisi. |
| Kelola *Requirement* | *Use case* ini menggambarkan kegiatan pengelolaan data *requirement* dari posisi yang ada. Terdapat dua jenis *requirement* pada sistem, yaitu *requirement jobtitle* dan *requirement* posisi. |
| Kelola Kantor | *Use case* ini menggambarkan kegiatan pengelolaan data kantor, seperti nama, alamat, jenis kantor. |
| Kelola Lowongan Kerja | *Use case* ini menggambarkan kegiatan pengelolaan informasi lowongan kerja. |
| Kelola Data Pelamar | *Use case* ini menggambarkan kegiatan pengelolaan data kandidat yang melamar pekerjaan. |
| Kelola Aplikasi Lamaran | *Use case* ini menggambarkan kegiatan pengelolaan informasi aplikasi lamaran yang menghubungkan lowongan kerja dengan pelamar. |
| *Logout* | *Use case* ini menggambarkan kegiatan *logout* keluar dari aplikasi. |
| *Login* | *Use case* ini menggambarkan kegiatan *login* atau masuk ke dalam aplikasi. |
| Pemeringkatan Kualifikasi Pelamar | *Use case* ini menggambarkan kegiatan pemeringkatan kualifikasi dari para pelamar dengan menggunakan metode DiA. |
| Masukkan Bobot Kriteria | *Use case* ini menggambarkan kegiatan *input* bobot kriteria penilaian. |
| Pilih Daftar Pelamar | *Use case* ini menggambarkan kegiatan pemilihan daftar pelamar untuk diseleksi. |
| Lihat Hasil Pemeringkatan | *Use case* ini menggambarkan kegiatan melihat hasil pemeringkatan. |

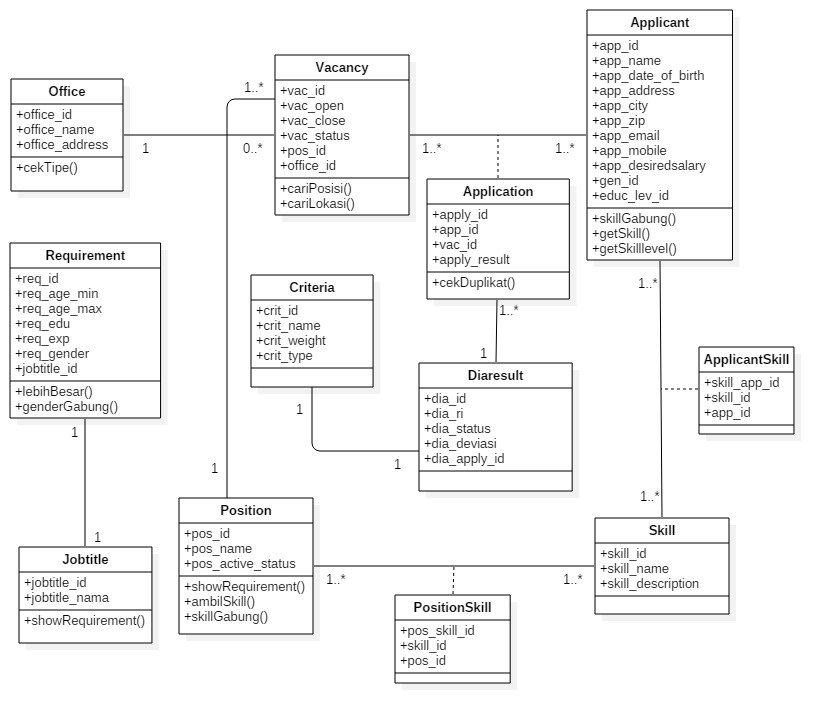
Pada Lampiran 2 terdapat penjelasan yang lebih detail terkait diagram *Use Case* sistem.

* + 1. ***Activity Diagram***

**Gambar 4.2 Diagram Aktivitas Proses Pemeringkatan Pelamar**

Pada Gambar 4.2 di atas dapat dilihat terdapat diagram aktivitas merepresentasikan alur pemeringkatan yang dilakukan sistem. Proses perhitungan dengan metode DiA dimulai dari tahap normalisasi nilai alternatif hingga penghitungan jarak alternatif ke PIA.

Beberapa kriteria akan digunakan sebagai dasar penilaian pelamar yang akan direkomendasikan sistem. Kriteria yang digunakan ialah umur, tingkat pendidikan, jenis kelamin, pengalaman kerja, kemampuan, dan gaji yang diharapkan. Sedangkan alternatif merupakan daftar pelamar yang melamar pada posisi tertentu pada lowongan pekerjaan.

* + 1. ***Class Diagram***

Gambar 4.3 *Class Diagram* Sistem

*Class diagram* dari modul rekrutmen sistem informasi perencanaan SDM dapat dilihat pada Gambar 4.4. Berikut penjelasan ringkas dari tiap kelas :

1. *Applicant* : Kelas ini mengelola data pelamar, yaitu informasi diri.
2. *Jobtitle* : Kelas ini mengelola data jabatan.
3. *Position* : Kelas ini mengelola data posisi. Kelas posisi memiliki kelas asosiatif dengan kelas *Skill* yang dinamakan *PositionSkill.* Kelas tersebut mengelola data *requirement* kemampuan dari masing-masing posisi.
4. *Office* : Kelas ini mengelola data kantor.
5. *Skill* : Kelas ini mengelola data kemampuan.
6. *Requirement* : Kelas ini mengelola data *requirement* dari *jobtitle*.
7. *Criteria* : Kelas ini mengelola data kriteria yang akan dinilai ketika melakukan pemeringkatan.
8. *Diaresult* : Kelas ini mengelola data hasil pemeringkatan.
9. *Vacancy* : Kelas ini mengelola data lowongan pekerjaan.
   * 1. ***Data Model***

Identifikasirancangan basis data dari sistem dilakukan melalui tahapan, yaitu *conceptual design, logical design,* dan *physical design*. Rancangan dibuat dengan menggunakan notasi *Crow Foot*. Rancangan *conceptual design, logical design, dan physical design* dapat ditemukan pada Lampiran 2.

* 1. **Implementasi**

Pada tahap implementasi, rancangan sistem yang telah dibuat pada tahap Perancangan akan diwujudkan ke dalam bentuk sistem yang sesungguhnya. Dengan berpatokan pada hasil perancangan, dilakukan pengembangan sistem untuk membangun fitur-fitur sistem agar dapat digunakan oleh pengguna. Bagian ini akan memaparkan kegiatan yang dilakukan ketika memasuki tahap Implementasi dari proses pengembangan. Pemaparan tahap implementasi akan dibagi menjadi dua bagian, yaitu implementasi metode DiA dan implementasi sistem.

* + 1. **Implementasi Metode DiA**

Seperti yang sudah disebutkan sebelumnya, metode DiA diterapkan pada subkategori bernama Seleksi Resume. Pada subkategori Seleksi Resume terdiri dari beberapa langkah yang harus dilakukan pengguna secara berurutan. Gambar 4.4 merepresentasikan urutan langkah yang perlu dilakukan oleh pengguna. Masing-masing langkah tersebut menuntun pengguna dalam menyediakan informasi yang dibutuhkan sistem untuk melakukan seleksi. Pada langkah pertama, sistem akan menyimpan data daftar pelamar yang akan diseleksi. Pada langkah kedua, sistem akan menyimpan bobot kriteria yang dimasukkan pengguna untuk selanjutnya digunakan dalam penilaian. Langkah ketiga hanya sekedar memberikan tampilan daftar pelamar yang akan diseleksi. Setelah selesai melihat data pelamar, pengguna akan menekan tombol “Lakukan Seleksi”. Aksi ini akan memulai proses perhitungan dengan metode DiA. Setelah proses perhitungan dan pemeringkatan selesai , maka hasilnya akan ditampilkan kepada pengguna.

**Gambar 4.4 Urutan Langkah Pengguna Untuk Melakukan Seleksi**

Untuk mendapatkan gambaran yang lebih baik mengenai penerapan metode DiA pada sistem, berikut penjelasan implementasi metode DiA langkah demi langkah dengan menggunakan sampel data pada Tabel 4.2. Data pada Tabel 4.2 merupakan daftar orang yang melamar pada posisi *Corporate Legal Officer* dengan kebutuhan kualifikasi sebagai berikut :

* Jenis Kelamin : Laki-laki / Perempuan
* Usia : 23 s/d 35 tahun
* Pendidikan : minimal S1
* Pengalaman : 2 tahun
* Kemampuan : MS Office
* Gaji : Rp3.000.000 – Rp4.000.000

**Tabel 4.2 Data Kualifikasi Alternatif**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama | C1  (2) | C2  (5) | C3  (4) | C4  (3) | C5  (5) | C6  (3) |
| Pelamar 1 | P | 23 | S1 | 1 | MS Office | 4.000 |
| Pelamar 2 | L | 30 | S1 | 5 | MS Office | 4.000 |
| Pelamar 3 | L | 27 | S1 | 4 | MS Office | 4.000 |
| Pelamar 4 | L | 29 | S2 | 4 | MS Office | 7.000 |
| Pelamar 5 | L | 24 | S1 | 2 | MS Office | 4.000 |
| Pelamar 6 | L | 35 | S2 | 6 | MS Office | 5.500 |
| Pelamar 7 | P | 36 | S2 | 7 | MS Office | 5.000 |
| Pelamar 8 | L | 22 | S1 | 1 | MS Office | 4.000 |
| Pelamar 9 | L | 32 | S1 | 3 | MS Office | 5.000 |
| Pelamar 10 | L | 33 | S1 | 5 | MS Office | 4.000 |

Keterangan :

C1 = Jenis Kelamin

C2 = Usia

C3 = Pendidikan

C4 = Pengalaman (tahun)

C5 = Kemampuan

C6 = Gaji yang Diharapkan (ribuan)

1. **Matriks Keputusan X**

Sesuai dengan karakteristik permasalahan MADM yang tersusun dari sekumpulan data alternatif beserta kriteria, maka dalam implementasi metode DiA ke sistem banyak mengandalkan tipe data *array.* Untuk membuat matriks keputusan X, data masing-masing pelamar pada Tabel 4.2 dibandingkan dengan kebutuhan kualifikasi posisi yang dilamar dan dilakukan konversi berdasarkan tabel konversi tiap kriteria di Bab 3. Berikut merupakan salah satu contoh perbandingan dan konversi dari data pelamar 1.

**Tabel 4.3 Tabel Perbandingan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Jenis Kelamin | Usia | Pendidikan | Pengalaman | Kemampuan | Gaji |
| *Requirement*  Posisi | L/P | 23 s/d 35 | S1 | 2 | MS Office | 3.000 s/d 4.000 |
| Kualifikasi Pelamar 1 | P | 23 | S1 | 1 | MS Office | 4.000 |

Dengan membandingkan jenis kelamin pelamar dengan *requirement*, diketahui bahwa hasilnya sesuai. Pada Tabel 3.1, hasil perbandingan yang sesuai maka bernilai 1. Hasil perbandingan usia juga sesuai, maka bernilai 1. Hasil perbandingan pendidikan merujuk pada Tabel 3.3 menjadi bernilai 80. Untuk data pengalaman kerja disajikan apa adanya, nilai tidak berubah. Hasil perbandingan kemampuan merujuk pada Tabel 3.4 menjadi bernilai 1, karena kemampuan pelamar memenuhi seluruh kemampuan yang disyaratkan*.* Sedangkan untuk gaji, data disajikan apa adanya, tidak ada perubahan. Maka hasil konversi data pelamar 1 hingga 10 dapat dilihat pada Tabel 4.4.

**­Tabel 4.4 Data Kualifikasi Alternatif Setelah Perbandingan dan Konversi**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama | C1  (2) | C2  (5) | C3  (4) | C4  (3) | C5  (5) | C6  (3) |
| Pelamar 1 | 1 | 1 | 80 | 1 | 1 | 4000 |
| Pelamar 2 | 1 | 1 | 80 | 5 | 1 | 4000 |
| Pelamar 3 | 1 | 1 | 80 | 4 | 1 | 4000 |
| Pelamar 4 | 1 | 1 | 90 | 4 | 1 | 7000 |
| Pelamar 5 | 1 | 1 | 80 | 2 | 1 | 4000 |
| Pelamar 6 | 1 | 1 | 90 | 6 | 1 | 5500 |
| Pelamar 7 | 1 | 0,75 | 90 | 7 | 1 | 5000 |
| Pelamar 8 | 1 | 0,75 | 80 | 1 | 1 | 4000 |
| Pelamar 9 | 1 | 1 | 80 | 3 | 1 | 5000 |
| Pelamar 10 | 1 | 1 | 80 | 5 | 1 | 4000 |



**Gambar 4.5 Kode Program Matriks Keputusan X**

Gambar 4.5 merupakan kode program untuk menyusun matriks keputusan X. Data para pelamar disimpan dalam sebuah *array* dua dimensi bernama “alternatif”. Fungsi nilaiEdu(), nilaijk(), nilaiusia(), nilaikemampuan(), dan experienceLong() adalah untuk melakukan perbandingan dan konversi nilai. Isi *array* “alternatif” ialah seperti pada Tabel 4.4. Jika direpresentasikan dengan matriks ialah sebagai berikut.

1. **Matriks R Ternormalisasi**

Nilai tiap-tiap elemen matriks X kemudian dinormalisasi dengan menggunakan rumus (3). Hasil normalisasi tiap elemen akan membentuk matriks yang dinamakan matriks R. Salah satu contoh perhitungan normalisasi elemen dengan i = 1 dan j =1 dari matriks X ialah sebagai berikut.

= =

= 0,3162

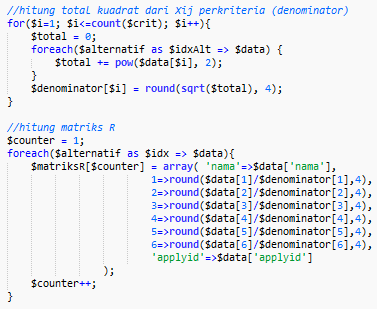
Tabel 4.5 menyajikan data hasil perhitungan normalisasi matriks X.

**­Tabel 4.5 Hasil Perhitungan Normalisasi Matriks X**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama | C1  (2) | C2  (5) | C3  (4) | C4  (3) | C5  (5) | C6  (3) |
| Pelamar 1 | 0,3162 | 0,331 | 0,3043 | 0,0741 | 0,3162 | 0,2665 |
| Pelamar 2 | 0,3162 | 0,331 | 0,3043 | 0,3706 | 0,3162 | 0,2665 |
| Pelamar 3 | 0,3162 | 0,331 | 0,3043 | 0,2965 | 0,3162 | 0,2665 |
| Pelamar 4 | 0,3162 | 0,331 | 0,3424 | 0,2965 | 0,3162 | 0,4664 |
| Pelamar 5 | 0,3162 | 0,331 | 0,3043 | 0,1483 | 0,3162 | 0,2665 |
| Pelamar 6 | 0,3162 | 0,331 | 0,3424 | 0,4448 | 0,3162 | 0,3665 |
| Pelamar 7 | 0,3162 | 0,2483 | 0,3424 | 0,5189 | 0,3162 | 0,3331 |
| Pelamar 8 | 0,3162 | 0,2483 | 0,3043 | 0,0741 | 0,3162 | 0,2665 |
| Pelamar 9 | 0,3162 | 0,331 | 0,3043 | 0,2224 | 0,3162 | 0,3331 |
| Pelamar 10 | 0,3162 | 0,331 | 0,3043 | 0,3706 | 0,3162 | 0,2665 |

Data dari Tabel 4.5 akan membentuk matriks R seperti berikut ini:

Gambar 4.6 menunjukkan kode untuk membuat matriks R. Dapat dilihat bahwa dilakukan pembulatan hingga 4 bilangan di belakang koma. Variabel denominator menyimpan data pembagi untuk masing-masing kriteria.



**Gambar 4.6 Kode Program Matriks R**

1. **Matriks V Terbobot**

Setelah mendapatkan nilai matriks R, berikutnya akan dihitung nilai matriks V terbobot. Perhitungan menggunakan rumus (5). Untuk mendapatkan nilai matriks V, tiap elemen matriks R akan dikalikan dengan bobot ternormalisasi dari masing-masing kriteria. Bobot masing-masing kriteria yang telah ditentukan pada Bab 3 harus dinormalisasi terlebih dahulu untuk memenuhi syarat dimana ∑w = 1. Normalisasi dilakukan dengan cara membagi nilai bobot sebuah kriteria dengan jumlah bobot seluruh kriteria. Berikut merupakan penyelesaian normalisasi bobot kriteria jenis kelamin.

Maka didapatkan bobot ternormalisasi masing-masing kriteria sebagai berikut.

Setelah itu perhitungan matriks V dapat dilakukan. Berikut salah satu penyelesaian perhitungan elemen matriks V untuk i = 1 dan j = 1.

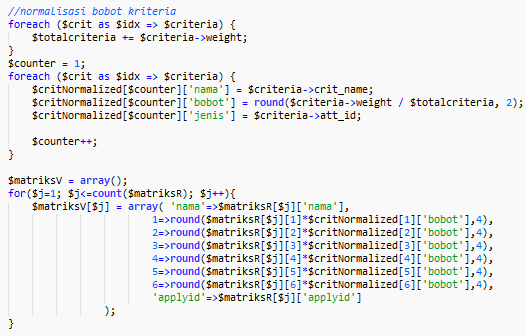
Dengan begitu didapatkan nilai matriks V yang disajikan pada Tabel 4.6.

**Tabel 4.6 Hasil Perhitungan Matriks V Terbobot**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama | C1  (0,09) | C2  (0,23) | C3  (0,18) | C4  (0,14) | C5  (0,23) | C6  (0,14) |
| Pelamar 1 | 0,0285 | 0,0761 | 0,0548 | 0,0104 | 0,0727 | 0,0373 |
| Pelamar 2 | 0,0285 | 0,0761 | 0,0548 | 0,0519 | 0,0727 | 0,0373 |
| Pelamar 3 | 0,0285 | 0,0761 | 0,0548 | 0,0415 | 0,0727 | 0,0373 |
| Pelamar 4 | 0,0285 | 0,0761 | 0,0616 | 0,0415 | 0,0727 | 0,0653 |
| Pelamar 5 | 0,0285 | 0,0761 | 0,0548 | 0,0208 | 0,0727 | 0,0373 |
| Pelamar 6 | 0,0285 | 0,0761 | 0,0616 | 0,0623 | 0,0727 | 0,0513 |
| Pelamar 7 | 0,0285 | 0,0571 | 0,0616 | 0,0726 | 0,0727 | 0,0466 |
| Pelamar 8 | 0,0285 | 0,0571 | 0,0548 | 0,0104 | 0,0727 | 0,0373 |
| Pelamar 9 | 0,0285 | 0,0761 | 0,0548 | 0,0311 | 0,0727 | 0,0466 |
| Pelamar 10 | 0,0285 | 0,0761 | 0,0548 | 0,0519 | 0,0727 | 0,0373 |

Penyajian Tabel 4.6 dalam bentuk matriks ialah sebagai berikut.

Gambar 4.7 menunjukkan kode program perhitungan matriks V. Dapat dilihat pada Gambar 4.10, terdapat tiga buah *looping.* Dua buah *foreach loop* yang di awal untuk melakukan normalisasi bobot kriteria. Sedangkan *for loop* yang terakhir untuk menghitung matriks V dengan cara mengalikan setiap elemen matriks R dengan bobot kriteria yang sudah dinormalisasi. Kemudian dilakukan pembulatan terhadap hasil hingga 4 angka di belakang koma.



**Gambar 4.7 Kode Program Matriks V**

1. **Nilai Solusi Ideal Positif dan Negatif**

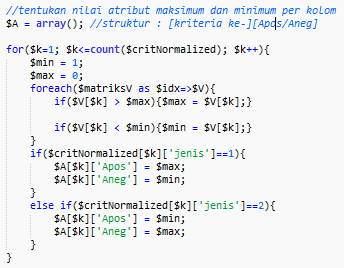
Penentuan nilai solusi ideal positif dan negatif menggunakan masing-masing satu rumus, namun rumusnya dibedakan untuk kriteria jenis *benefit* dan *cost.* Untuk kriteria yang berjenis *benefit*, dimana nilai lebih tinggi yang diprioritaskan, dapat menggunakan rumus (6) dan (7). Sedangkan, untuk kriteria yang berjenis *cost,* dimana nilai lebih rendah yang diprioritaskan, dapat menggunakan rumus (8) dan (9). Pada penelitian ini, kriteria yang berjenis *cost* ialah gaji yang diharapkan. Kriteria selain dari gaji yang diharapkan merupakan kriteria *benefit.* Sebagai contoh, nilai solusi ideal positif kriteria Jenis Kelamin yang berjenis *benefit* didapatkan dengan mencari nilai maksimum dari kolom Jenis Kelamin pada Matriks V. Sedangkan untuk nilai solusi ideal negatif kriteria Jenis Kelamin yang berjenis *benefit* didapatkan dengan mencari nilai minimum dari kolom Jenis Kelamin pada Matriks V.Tabel 4.7 menyajikan nilai solusi ideal positif dan solusi ideal negatif dari masing-masing kriteria.

**Tabel 4.7 Solusi Ideal Positif dan Negatif**

|  |  |  |  |
| --- | --- | --- | --- |
| Kriteria | Jenis | Solusi Ideal Positif  () | Solusi Ideal Negatif  () |
| Jenis Kelamin | *Benefit* | 0,0285 | 0,0285 |
| Usia | *Benefit* | 0,0761 | 0,0571 |
| Pendidikan | *Benefit* | 0,0616 | 0,0548 |
| Pengalaman | *Benefit* | 0,0726 | 0,0104 |
| Kemampuan | *Benefit* | 0,0727 | 0,0727 |
| Gaji yang Diharapkan | *Cost* | 0,0373 | 0,0653 |

Dapat dilihat pada Tabel 4.7, kriteria jenis kelamin dan kemampuan masing-masing memiliki nilai solusi ideal positif dan solusi ideal negatif yang sama. Hal ini disebabkan karena dua kriteria tersebut memiliki nilai yang seragam pada matriks V. Untuk mendapatkan nilai solusi ideal positif dari kriteria berjenis *benefit* dicari nilai maksimum dari seluruh alternatif terkait kriteria tersebut. Untuk mendapatkan nilai solusi ideal negatif dari kriteria berjenis *benefit* dicari nilai minimum dari seluruh alternatif terkait kriteria tersebut. Dikarenakan nilai seluruh alternatif dalam kriteria jenis kelamin ialah seragam, yaitu 0.0285, maka nilai maksimum dan minimumnya pun 0,0285. Sama halnya dengan kriteria kemampuan.

Kriteria gaji yang diharapkan memiliki nilai solusi ideal positif yang lebih rendah daripada nilai solusi ideal negatif. Hal ini disebabkan oleh jenis kriteria *cost.* Untuk jenis kriteria ini, solusi ideal yang paling baik ialah yang memiliki nilai paling rendah. Sedangkan untuk solusi ideal yang paling buruk ialah yang memiliki nilai tertinggi.



**Gambar 4.8 Kode Program Solusi Ideal Positif dan Negatif**

Gambar 4.8 menunjukkan kode program untuk menentukan solusi ideal positif dan negatif. Program melakukan iterasi pada setiap kriteria. Kemudian menentukan nilai maksimum dan minimum dari seluruh alternatif pada kriteria tersebut. Setelah itu melakukan pengecekan jenis kriteria. Apabila kriteria berjenis *benefit,* maka simpan nilai maksimum sebagai nilai solusi ideal positif dan nilai minimum sebagai nilai solusi ideal negatif. Jika kriteria *cost,* maka lakukan sebaliknya.

1. **Jarak Manhattan dan PIA**

Jarak Manhattan dihitung untuk mengetahui total jarak masing-masing pelamar terhadap solusi ideal positif dan solusi ideal negatif dari seluruh kriteria. Masing-masing pelamar memiliki jarak Manhattan positif dan jarak Manhattan negatif. Jarak Manhattan positif merupakan nilai total jarak pelamar dari solusi ideal positif seluruh kriteria. Sedangkan, jarak Manhattan negatif merupakan nilai total jarak pelamar dari solusi ideal negatif seluruh kriteria. Untuk menghitung jarak Manhattan dapat menggunakan rumus (10) dan (11). Penghitungan jarak dilakukan dengan mencari selisih antara tiap-tiap nilai pelamar di matriks V dengan solusi ideal positif dan negatif masing-masing kriteria. Kemudian, jarak Manhattan positif akan didapatkan dengan menjumlahkan seluruh hasil selisih perhitungan dengan solusi ideal positif. Jarak Manhattan negatif akan didapatkan dengan menjumlahkan seluruh hasil selisih perhitungan dengan solusi ideal negatif. Berikut salah satu penyelesaian untuk menghitung jarak Manhattan positif (D+) dan negatif (D-) dari pelamar 1 dimana i = 1.

Tabel 4.8 menyajikan nilai jarak Manhattan positif dan negatif dari seluruh pelamar.

**Tabel 4.8 Jarak Manhattan**

|  |  |  |
| --- | --- | --- |
| Nama | D+ | D- |
| Pelamar 1 | 0,069 | 0,047 |
| Pelamar 2 | 0,0275 | 0,0885 |
| Pelamar 3 | 0,0379 | 0,0781 |
| Pelamar 4 | 0,0591 | 0,0569 |
| Pelamar 5 | 0,0586 | 0,0574 |
| Pelamar 6 | 0,0243 | 0,0917 |
| Pelamar 7 | 0,0283 | 0,0877 |
| Pelamar 8 | 0,088 | 0,028 |
| Pelamar 9 | 0,0576 | 0,0584 |
| Pelamar 10 | 0,0275 | 0,0885 |

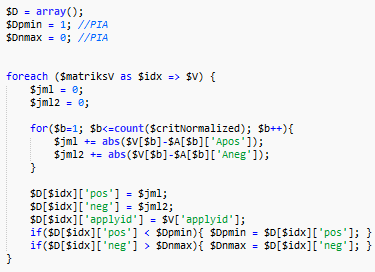
Dari Tabel 4.8, nilai D+ maksimum dimiliki oleh Pelamar 8 sebesar 0,088 dan D+ minimum dimiliki oleh Pelamar 6 sebesar 0,0243. Hal itu menyatakan bahwa Pelamar 8 memiliki jarak paling jauh ke Solusi Ideal Positif dan Pelamar 2 memiliki jarak terdekat ke Solusi Ideal Positif.

Nilai D- maksimum dimiliki oleh Pelamar 6 sebesar 0,0917 dan D- minimum dimiliki oleh Pelamar 8 sebesar 0,028. Hal itu menyatakan bahwa Pelamar 6 memiliki jarak terjauh dengan Solusi Ideal Negatif dan Pelamar 8 memiliki jarak terdekat dengan Solusi Ideal Negatif.

Nilai PIA didapatkan dengan menggunakan rumus (12). Nilai PIA adalah nilai minimum D+ dan nilai maksimum D-. Maka dari itu nilai PIA ialah sebagai berikut.

PIA = {0,0243, 0,0917}

Gambar 4.9 merupakan kode program untuk melakukan perhitungan jarak Manhattan dan PIA. Variabel D menyimpan data *array* dari jarak Manhattan positif dan negatif tiap pelamar. Variabel Dpmin menyimpan salah satu nilai PIA, yaitu nilai minimum dari jarak Manhattan positif (D+). Variabel Dnmax menyimpan salah satu nilai PIA, yaitu nilai maksimum dari jarak Manhattan negatif (D-).



**Gambar 4.9 Kode Program Jarak Manhattan dan PIA**

1. **Jarak Alternatif ke PIA**

Setelah mendapatkan nilai PIA, jarak alternatif ke PIA dapat dihitung menggunakan rumus (13). Salah satu penyelesaian perhitungan Ri dengan i = 1 adalah sebagai berikut.

Tabel 4.9 menyajikan nilai jarak alternatif ke PIA dari masing-masing pelamar beserta peringkat mereka. Semakin kecil nilai Ri maka alternatif tersebut semakin direkomendasikan.

**Tabel 4.9 Jarak Alternatif ke PIA**

|  |  |  |
| --- | --- | --- |
| Nama | Ri | Peringkat |
| Pelamar 1 | 0,0632 | 9 |
| Pelamar 2 | 0,0045 | 2 |
| Pelamar 3 | 0,0192 | 5 |
| Pelamar 4 | 0,0492 | 8 |
| Pelamar 5 | 0,0485 | 7 |
| Pelamar 6 | 0 | 1 |
| Pelamar 7 | 0,0057 | 4 |
| Pelamar 8 | 0,0901 | 10 |
| Pelamar 9 | 0,0471 | 6 |
| Pelamar 10 | 0,0045 | 3 |

Dari tabel di atas, dapat dilihat 5 orang pelamar dengan peringkat teratas ialah Pelamar 6, Pelamar 2, Pelamar 10, Pelamar 7, dan Pelamar 3. Tabel 4.10 menyajikan data kualifikasi alternatif setelah diurutkan sesuai peringkat. Jika diperhatikan, pelamar yang menduduki peringkat 1 hingga 3 seluruhnya memenuhi kriteria. Dari peringkat 1 hingga 3, masing-masing pelamar mendapatkan nilai maksimal pada kriteria jenis kelamin, usia, serta kemampuan. Sehingga yang menjadi penentu ialah dari kriteria pendidikan yang memiliki tingkat prioritas Penting dan juga kriteria pengalaman. Dalam hal ini, pelamar 6 memiliki pendidikan dan pengalaman yang lebih tinggi dari dua lainnya maka dari itu pelamar 6 lebih unggul dari yang lain. Pelamar 6 dapat menduduki peringkat 1 meskipun gaji yang diharapkannya melebihi batas. Hal ini dapat terjadi karena tingkat prioritas kriteria gaji yang diharapkan adalah normal, sehingga kekurangan nilainya dapat ditutupi oleh nilai dari kriteria pendidikan dan pengalaman.

Pada peringkat ke-empat, terdapat Pelamar 7. Pelamar 7 memiliki ketidaksesuaian pada kriteria usia, dimana usia dari pelamar 7 lebih besar setahun daripada yang disyaratkan. Dari kriteria usia, pelamar 7 tidak mendapatkan nilai maksimal. Namun kekurangan nilainya dapat terbantu oleh nilai kriteria berprioritas tinggi, yaitu pendidikan serta pengalaman, yang tergolong cukup besar. Pelamar 7 adalah pelamar yang memiliki pengalaman kerja terpanjang di antara yang lainnya. Terlebih lagi, pelamar 7 memiliki pendidikan yang lebih unggul daripada yang lain. Maka dari itu, meskipun memiliki ketidaksesuaian pada kriteria usia, pelamar 7 layak direkomendasikan.

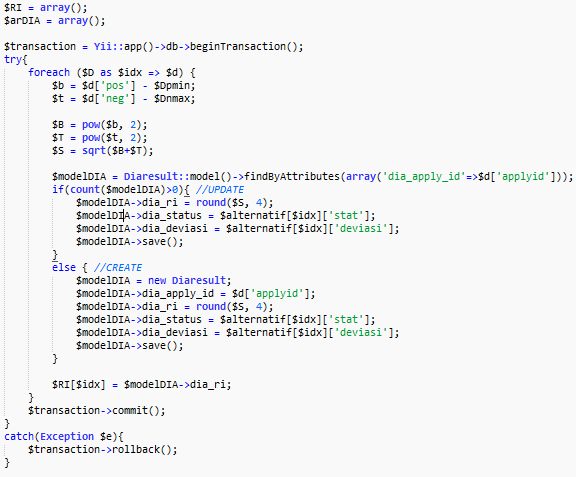
Pelamar 4, terlepas dari kenyataan bahwa ia mendapatkan nilai maksimal pada kriteria jenis kelamin, usia, dan kemampuan, berada pada posisi tiga terbawah. Kriteria yang menghalangi pelamar 4 untuk naik ke atas ialah kriteria gaji yang diharapkan. Pelamar 4 merupakan pemilik nilai kriteria C6 terbesar. Kriteria C6 berjenis *cost* dimana pelamar dengan nilai tertinggi maka kurang diprioritaskan. Meski tingkat prioritas kriteria C6 hanyalah normal, namun nilai gaji yang diharapkan pelamar 4 memiliki rentang yang terlalu besar dibanding dengan pelamar lainnya, maka ia tidak diprioritaskan.

Pelamar 1 dan 8 memiliki kualifikasi yang tidak jauh berbeda. Pelamar 1 mendapatkan nilai maksimal pada kriteria jenis kelamin, usia, pendidikan, serta kemampuan. Namun, pelamar 1 masih tertinggal jauh dengan yang lain pada kriteria pengalaman kerja. Nilai kriteria gaji yang diharapkan tidak mampu mendongkrak peringkat pelamar 1, karena mayoritas pelamar yang lain juga memiliki nilai rentang gaji yang tak jauh berbeda. Hal yang sama terjadi pada pelamar 8. Namun, pelamar 8 lebih tidak direkomendasikan daripada pelamar 1 karena nilai kriteria usianya yang tidak maksimal, layaknya yang dimiliki pelamar 8.

**Tabel 4.10 Data Kualifikasi Pelamar Setelah Pemeringkatan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nama  Kualifikasi | C1  (2)  L/P | C2  (5)  23 – 35 th | C3  (4)  S1 | C4  (3)  2 th | C5  (5)  Ms Office | C6  (3)  3jt – 4jt |
| Pelamar 6 | L | 35 | S2 | 6 | MS Office | 5.500 |
| Pelamar 2 | L | 30 | S1 | 5 | MS Office | 4.000 |
| Pelamar 10 | L | 33 | S1 | 5 | MS Office | 4.000 |
| Pelamar 7 | P | 36 | S2 | 7 | MS Office | 5.000 |
| Pelamar 3 | L | 27 | S1 | 4 | MS Office | 4.000 |
| Pelamar 9 | L | 32 | S1 | 3 | MS Office | 5.000 |
| Pelamar 5 | L | 24 | S1 | 2 | MS Office | 4.000 |
| Pelamar 4 | L | 29 | S2 | 4 | MS Office | 7.000 |
| Pelamar 1 | P | 23 | S1 | 1 | MS Office | 4.000 |
| Pelamar 8 | L | 22 | S1 | 1 | MS Office | 4.000 |

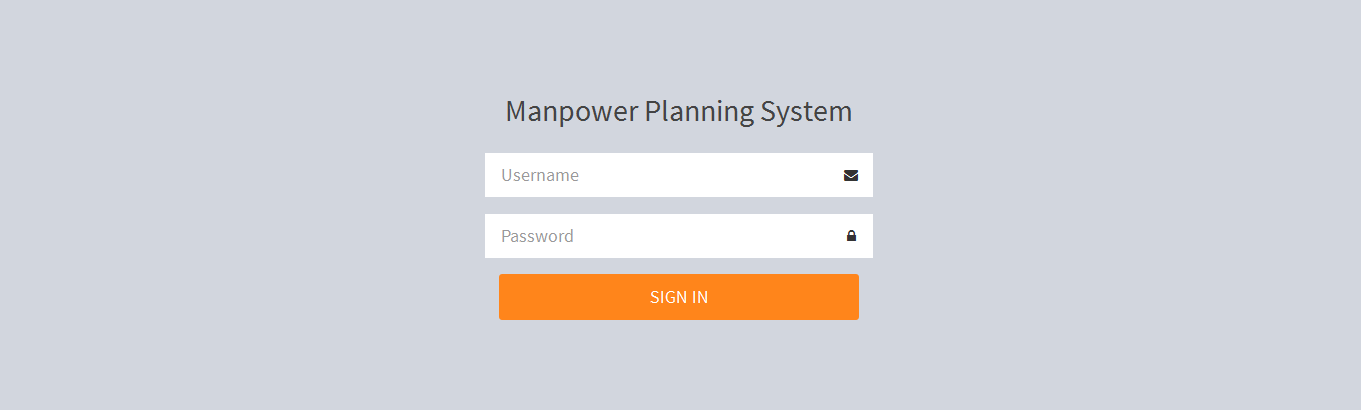
Gambar 4.10 menunjukkan kode program untuk menghitung nilai Ri. Nilai Ri atau nilai akhir masing-masing pelamar disimpan dalam variabel bernama RI. Dalam kode program, khususnya bagian yang menggunakan kondisi “If.. Else” berguna untuk menyimpan nilai terkait perhitungan DiA dari masing-masing pelamar ke basis data. Apabila terdapat kesalahan dalam penyimpanan, maka penyimpanan akan dibatalkan.



**Gambar 4.10 Kode Program Perhitungan RI**

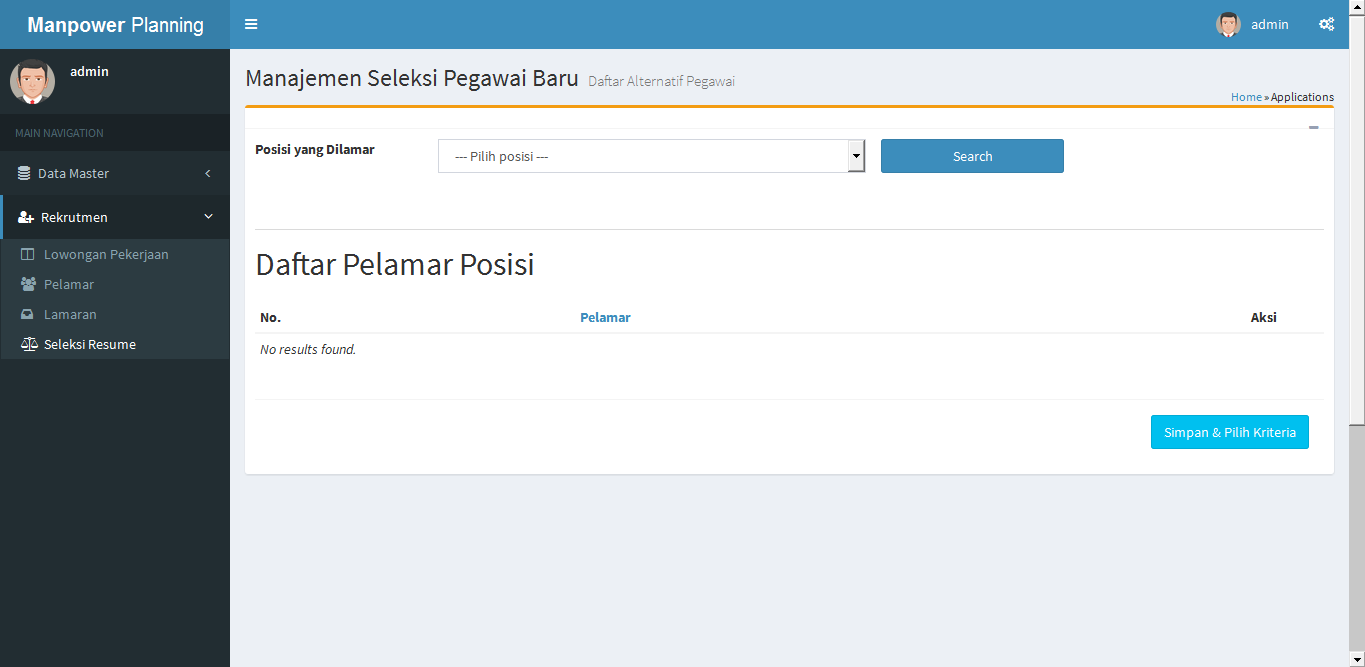
* + 1. **Implementasi Antarmuka**

Implementasi antarmuka sistem dilakukan berdasarkan pada perencanaan antarmuka yang telah dilakukan. Implementasi antar muka sistem menggunakan sebuah *template* HTML responsif *open source* yang didapatkan secara bebas melalui internet. Hal ini dilakukan bertujuan untuk memangkas waktu yang diperlukan untuk implementasi antarmuka sistem. *Template* yang disebut AdminLTE 2 memiliki ketergantungan pada Bootstrap 3, JQuery 1.11+, serta *plugin* lainnya agar bisa menampilkan elemen antarmuka dengan baik. Berikut hasil implementasi antarmuka sistem.

****

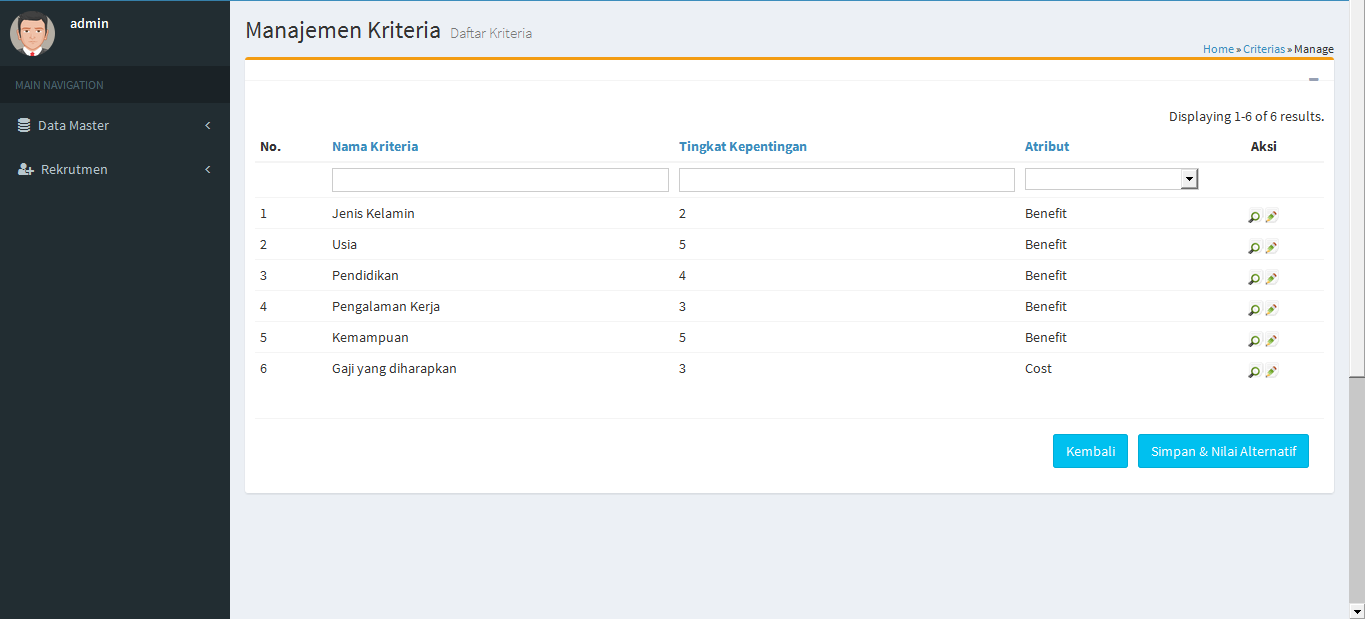
**Gambar 4.11 Halaman Login**

Gambar 4.11 menunjukkan halaman awal sistem, yaitu halaman *login*. Pengguna diharuskan memasukkan *username* dan *password* yang sesuai untuk bisa masuk ke dalam sistem.



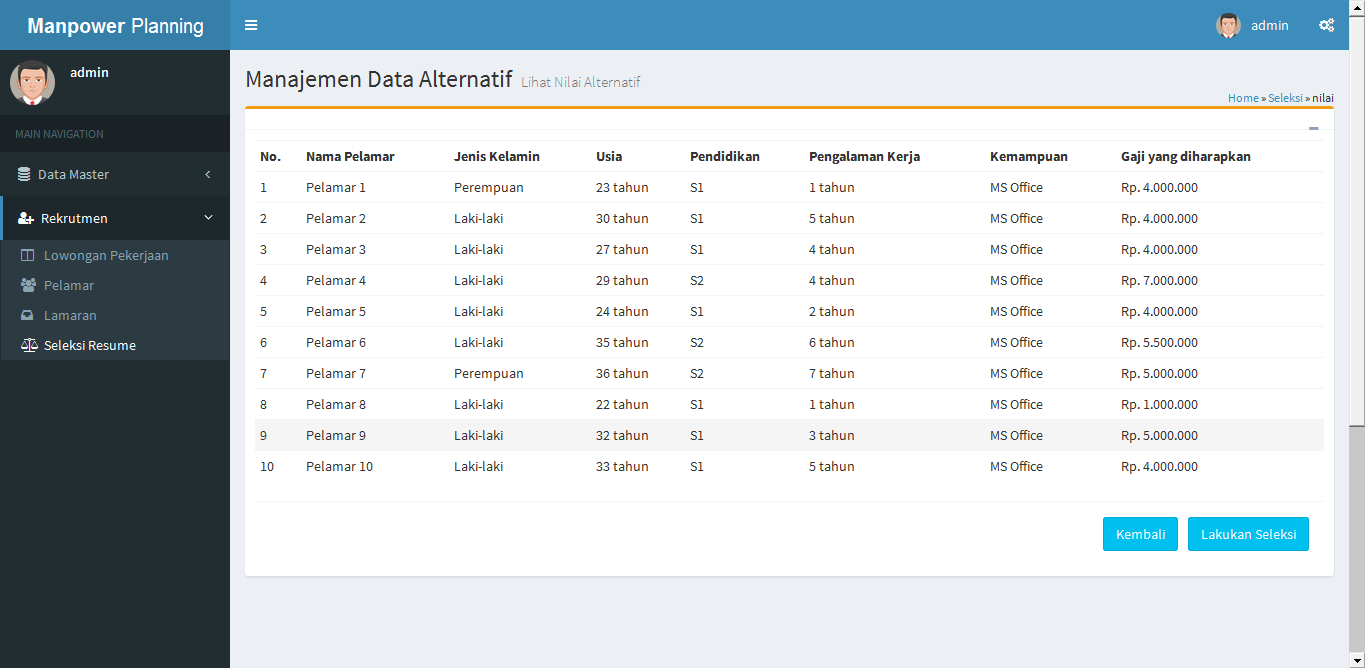
**Gambar 4.12 Halaman Awal Seleksi Resume**

Gambar 4.12 menunjukkan halaman awal subkategori seleksi resume. Pengguna diharuskan untuk memilih kolom posisi yang dilamar sebelum melakukan langkah selanjutnya. Selain itu, daftar pelamar posisi juga harus berisi minimal tiga orang pelamar untuk bisa melanjutkan ke tahap selanjutnya.

****

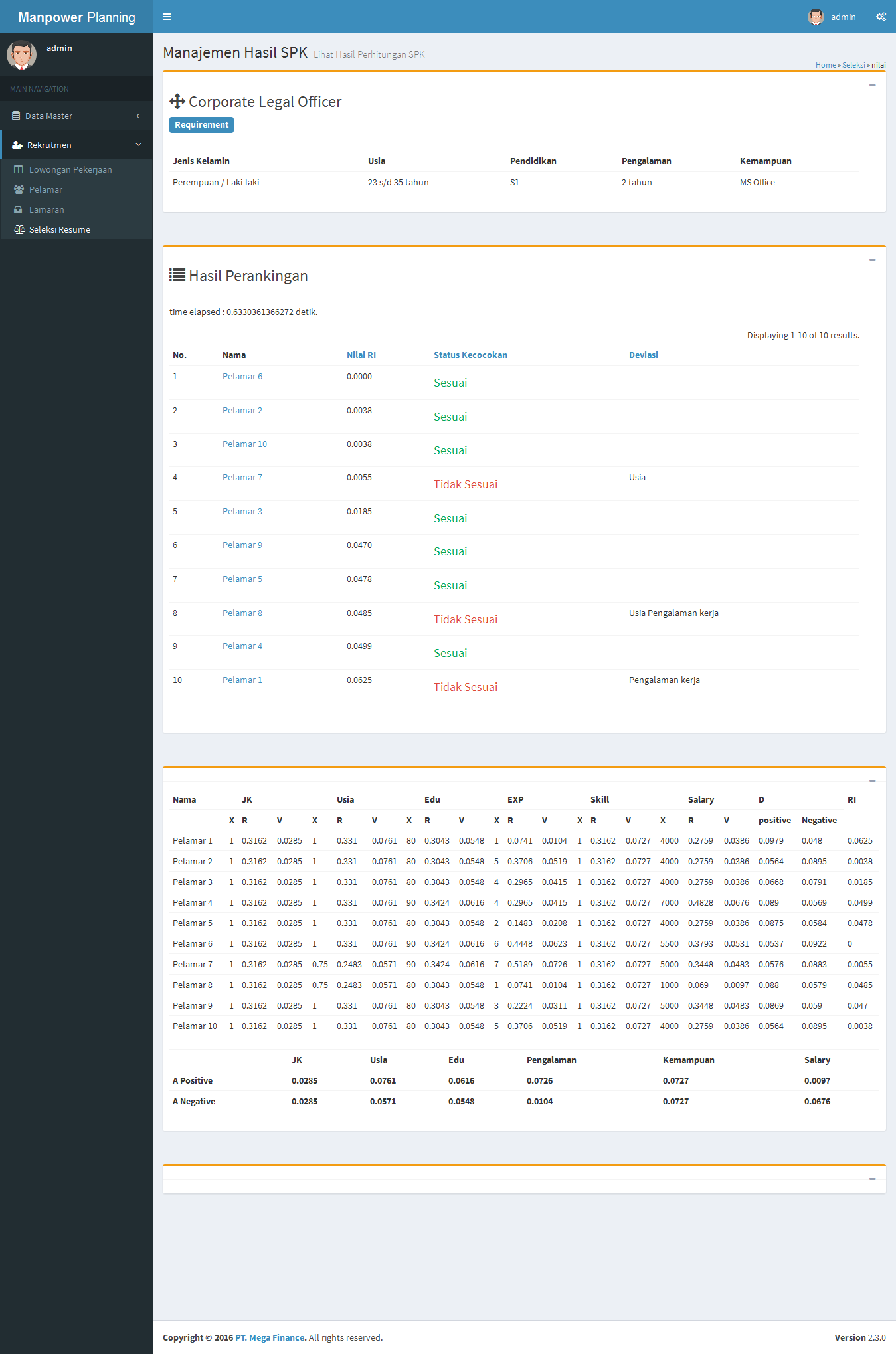
**Gambar 4.13 Halaman Pemilihan Tingkat Prioritas Kriteria**

Gambar 4.13 menunjukkan halaman pemilihan tingkat prioritas kriteria. Pengguna dapat menekan tombol bergambar pensil untuk mengubah tingkat kepentingan dan jenis kriteria baris tertentu.

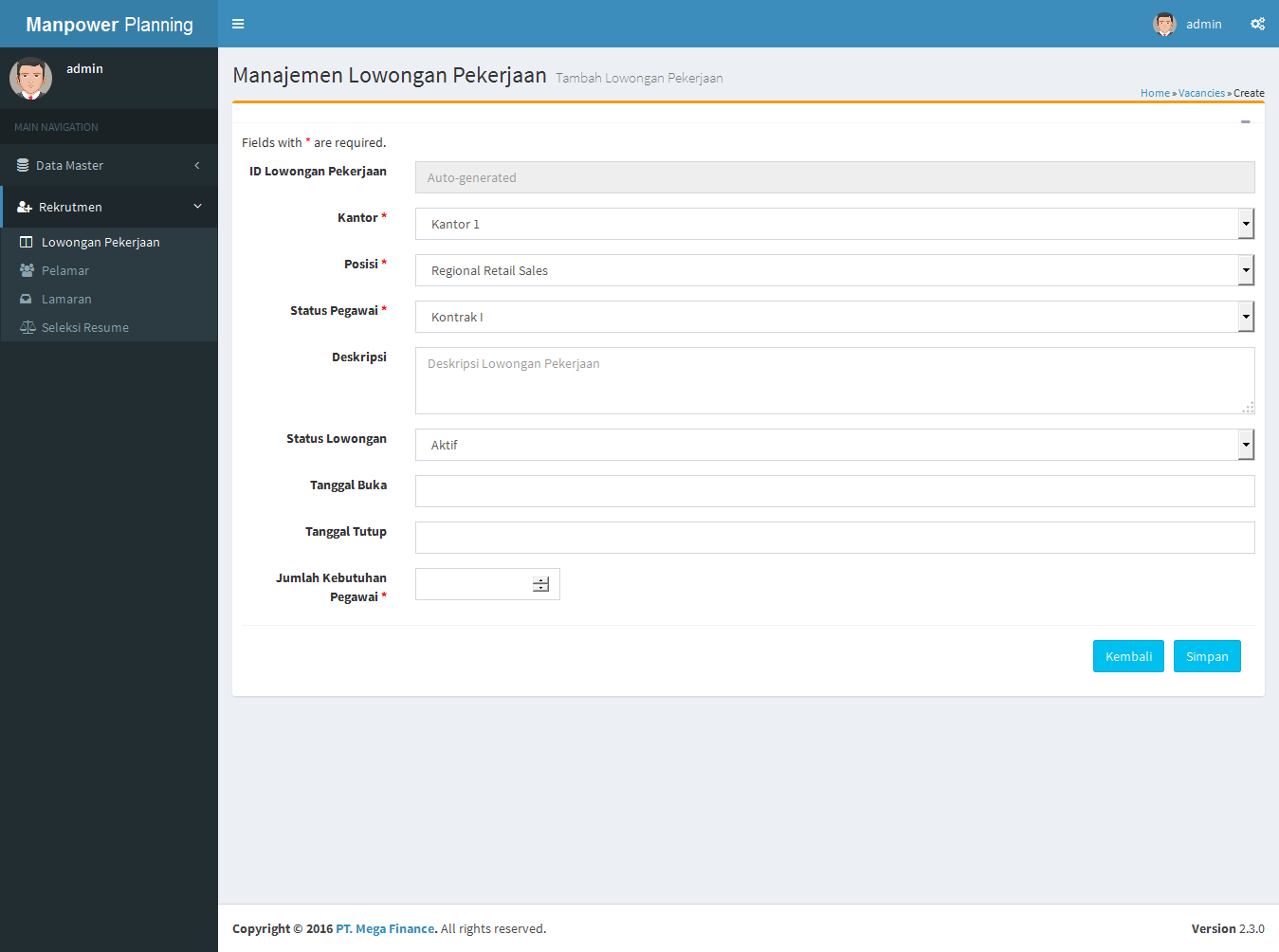
****

**Gambar 4.14 Halaman Melihat Daftar Data Pelamar Seleksi**

Gambar 4.14 menunjukkan halaman melihat daftar data kualifikasi pelamar sebelum diseleksi .

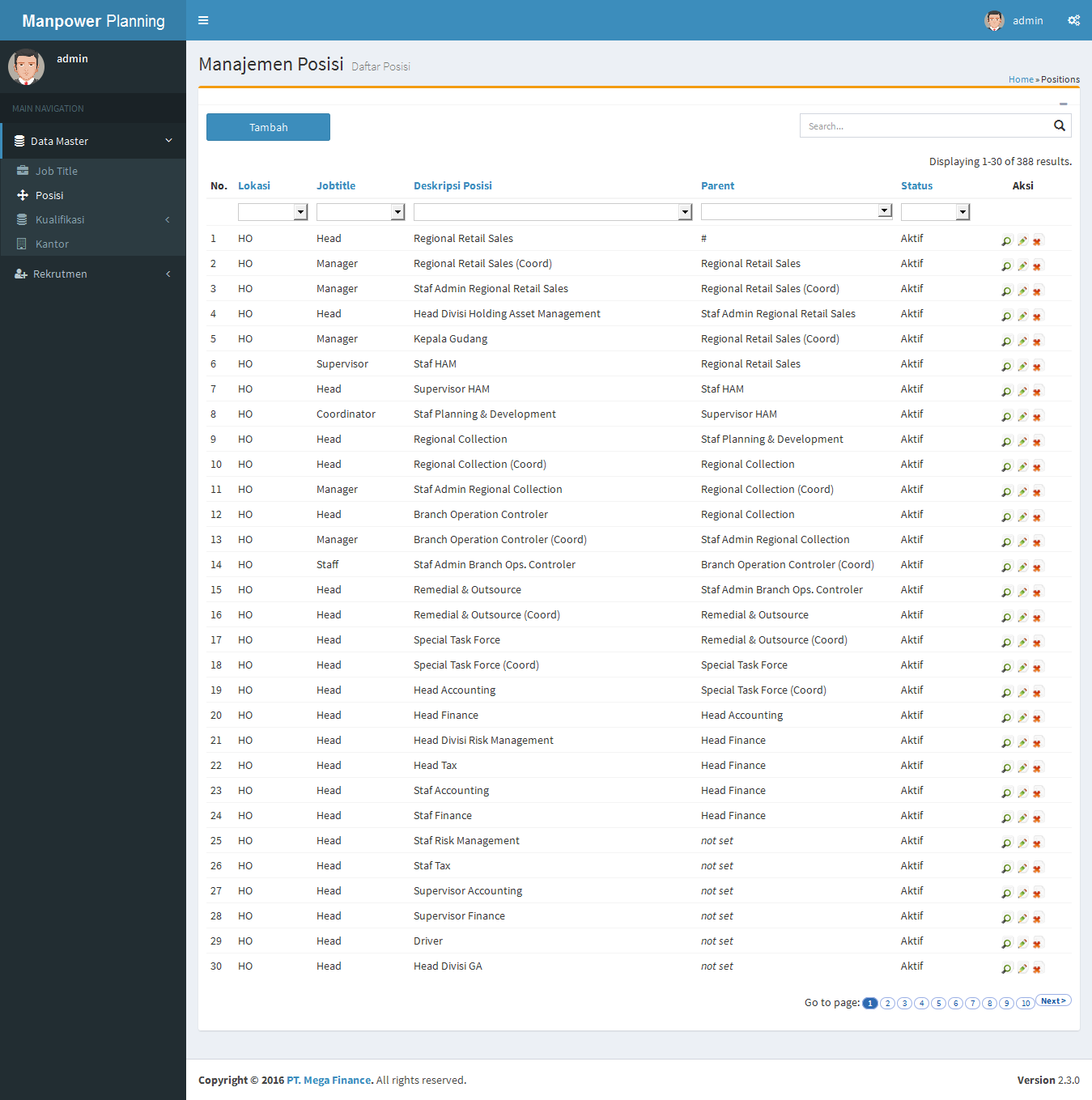
**Gambar 4.15 Halaman Lihat Hasil Seleksi**

Gambar 4.15 merupakan halaman yang muncul setelah seleksi selesai dilakukan. Terdapat sebuah tabel hasil pemeringkatan pelamar. Tabel terdiri dari kolom nama, nilai Ri, status kecocokan, dan deviasi. Kolom deviasi menunjukkan nama kualifikasi yang tidak dipenuhi oleh pelamar dengan status kecocokan tidak sesuai.



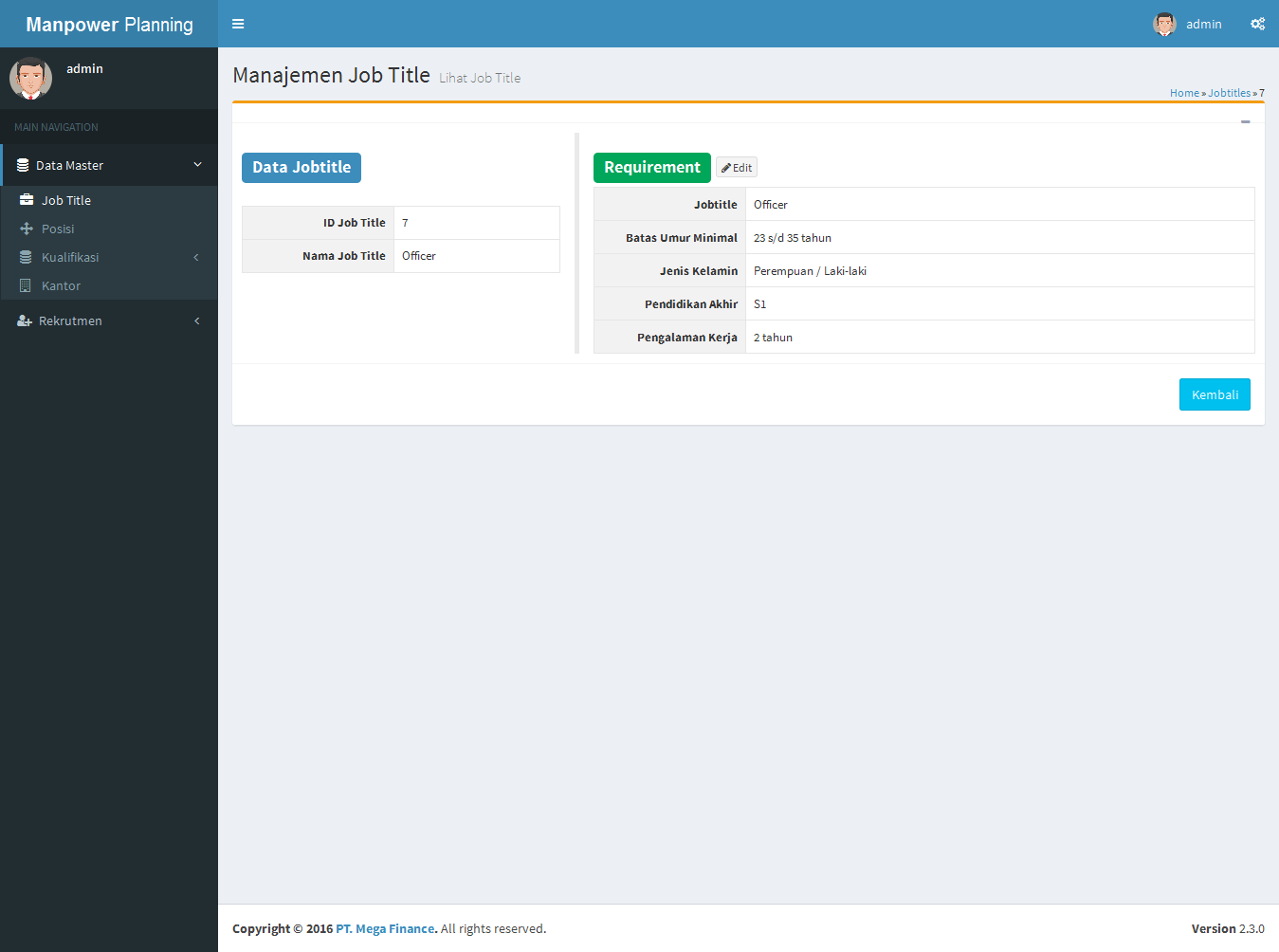
**Gambar 4.16 Halaman Tambah Lowongan Pekerjaan**

Gambar 4.16 merupakan halaman untuk membuat lowongan pekerjaan. Label dengan *asterisk* (\*) berwarna merah merupakan kolom yang wajib diisi. Tombol simpan akan menyimpan data yang telah dimasukkan ke dalam basis data.

****

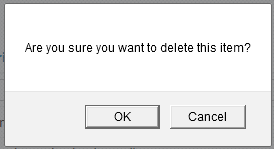
**Gambar 4.17 Halaman Lihat Daftar Posisi**

Gambar 4.17 merupakan halaman untuk melihat daftar posisi yang tersimpan dalam sistem. Kolom Aksi terdiri dari tiga buah tombol. Tombol kaca pembesar berguna untuk melihat seluruh informasi terkait posisi pada baris tersebut. Tombol pensil berguna untuk mengubah informasi terkait posisi pada baris tersebut. Tombol silang berwarna merah berguna untuk menghapus informasi posisi pada baris tersebut.

****

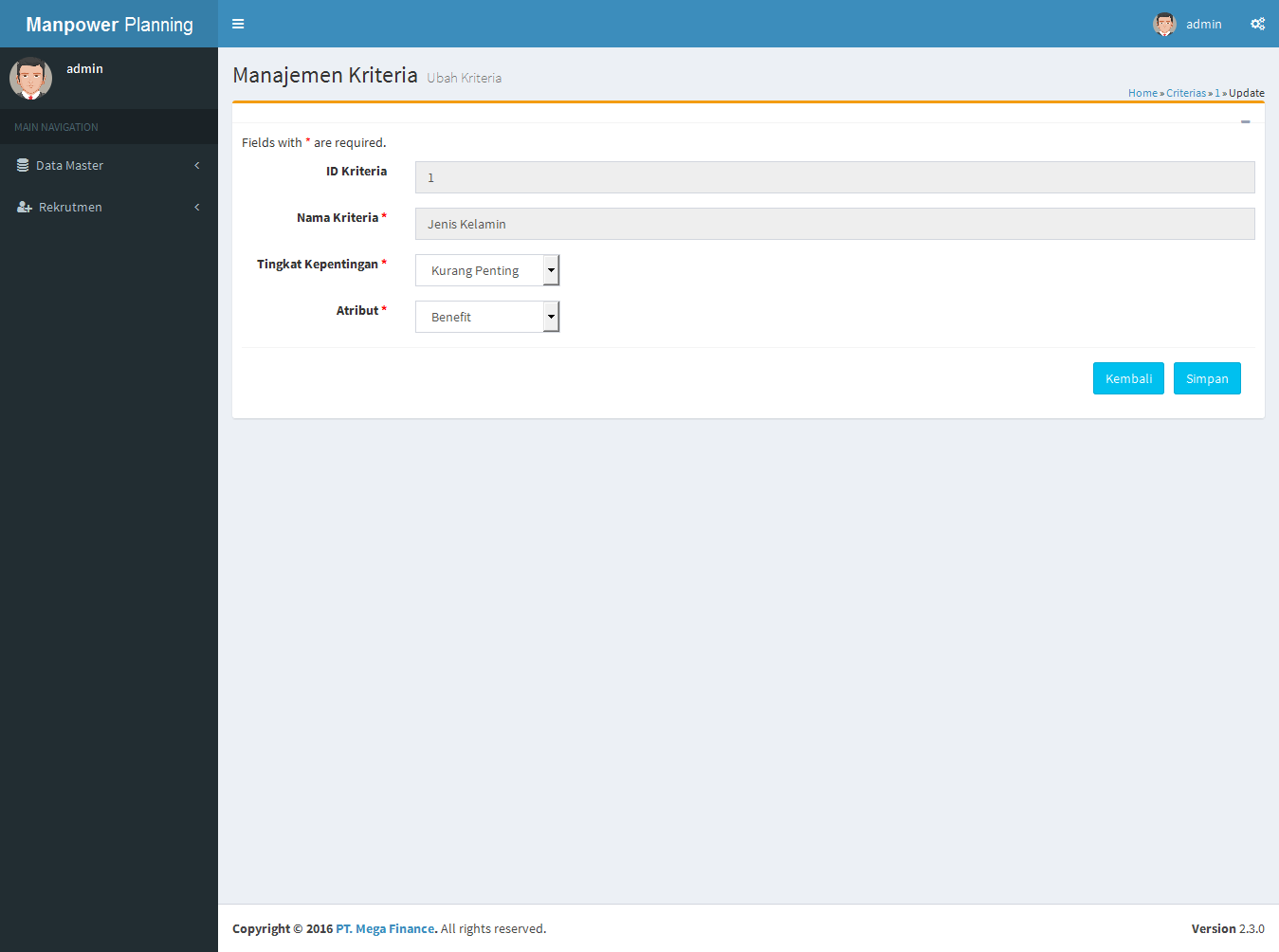
**Gambar 4.18 Halaman Lihat *Job Title***

Gambar 4.18 merupakan halaman melihat sebuah *Job Title*. Pada halaman ini terdapat informasi kualifikasi untuk *Job Title* tersebut.

****

**Gambar 4.19 *Pop-Up* Hapus Posisi**

Gambar 4.19 merupakan tampilan *pop*-*up* ketika tombol hapus ditekan.

****

**Gambar 4.20 Halaman Ubah Kriteria**

Gambar 4.20 merupakan tampilan untukmengubah nilai prioritas kriteria.

* 1. **Pengujian Algoritma**

Pengujian merupakan salah satu tahap penting dalam pengembangan sistem. Pada tahap ini dilakukan serangkaian pengujian terhadap sistem yang dibangun, khususnya pada implementasi algoritma di sistem. Hal ini dilakukan untuk memastikan bahwa metode DiA yang diimplementasikan ke dalam sistem telah berjalan sebagaimana mestinya. Pengujian sistem dilakukan dengan dua metode, yaitu *white box* dan *black box.* Pelaksanaanmasing-masing metode yang digunakan akan dijabarkan pada subbab setelah ini.

* + 1. **Pengujian *White Box***

Pengujian *White box* diterapkan pada beberapa fungsi yang melakukan perhitungan metode DiA. Hal ini dilakukan untuk mencari tahu keberadaan *error* dalam aplikasi dengan mempertimbangkan mekanisme internal sistem (Khan, 2011). Pengujian *White box* ini dilakukan oleh penulis. Beberapa *test case* akan digunakan sebagai acuan dalam pengujian. Hasil dari pengujian *White box* dapat dilihat pada Lampiran 4.

* + 1. **Pengujian *Black Box***

Pengujian *Black box* yang dilakukan dalam penelitian ini diterapkan dengan mengacu pada kebutuhan fungsional sistem yang ada di Lampiran 2. Pengujian *black box* ini dilakukan oleh *user* dari Departemen Rekrutmen dan Asesmen. Hasil pengujian *black box* disajikan pada Tabel 4.11.

**Tabel 4.11 *Test Case* dan Hasil Pengujian *Black Box***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Skenario Pengujian** | ***Test Case*** | **Status** |
| 1. | Mengizinkan *user* untuk melakukan *login* ke sistem. | *User* memasukkan informasi *username* dan *password* pada halaman *login* dan menekan tombol *Login* | Berhasil |
| 2. | Mengizinkan *user* untuk melakukan kelola *job title.* | *User* menekan tombol tambah, kemudian memasukkan data *job title* baru dan menyimpannya ke basis data | Berhasil |
| *User* menekan tombol *view* pada baris *job title* tertentu dan melihat informasi sebuah *job title* secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan perubahan data *job title* dan menyimpannya ke basis data | Berhasil |
| *User* menekan tombol *delete*, pada baris *job title* tertentu dan melakukan konfirmasi penghapusan *job title*. | Berhasil |
| 3. | Mengizinkan *User* untuk melakukan kelola posisi. | *User* menekan tombol tambah, kemudian memasukkan data *posisi* baru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris posisi tertentu dan melihat informasi sebuah posisi secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data *posisi* baru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris posisi tertentu dan melihat informasi sebuah posisi secara lengkap | Berhasil |
| 4. | Mengizinkan *User* untuk melakukan kelola *requirement*. | *User* menekan tombol tambah, kemudian memasukkan data *requirement* umumbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris *requirement* umum tertentu dan melihat informasi sebuah *requirement* umum secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data *requirement* umumbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris *requirement* umum tertentu dan melihat informasi sebuah *requirement* umum secara lengkap | Berhasil |
| *User* menekan tombol tambah, kemudian memasukkan data *requirement* khususbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris requirement khusus tertentu dan melihat informasi sebuah requirement umum secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data *requirement* khususbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris *requirement* khusus tertentu dan melihat informasi sebuah *requirement* khusus secara lengkap | Berhasil |
| 5. | Mengizinkan *User* untuk melakukan kelola kantor. | *User* menekan tombol tambah, kemudian memasukkan data kantorbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris kantor tertentu dan melihat informasi sebuah kantor secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data kantorbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris kantor tertentu dan melihat informasi sebuah kantor secara lengkap | Berhasil |
| 6. | Mengizinkan *User* untuk melakukan kelola lowongan kerja. | *User* menekan tombol tambah, kemudian memasukkan data lowongan kerjabaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris lowongan kerja tertentu dan melihat informasi sebuah lowongan kerja secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data lowongan kerjabaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris lowongan kerja tertentu dan melihat informasi sebuah lowongan kerja secara lengkap | Berhasil |
| 7. | Mengizinkan *User* untuk melakukan kelola data pelamar. | *User* menekan tombol tambah, kemudian memasukkan data pelamarbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris pelamar tertentu dan melihat informasi sebuah pelamar secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data pelamarbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris pelamar tertentu dan melihat informasi sebuah pelamar secara lengkap | Berhasil |
| 8. | Mengizinkan *User* untuk melakukan kelola aplikasi lamaran. | *User* menekan tombol tambah, kemudian memasukkan data aplikasi lamaranbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *view* pada baris aplikasi lamaran tertentu dan melihat informasi sebuah aplikasi lamaran secara lengkap | Berhasil |
| *User* menekan tombol *update*, kemudian memasukkan data aplikasi lamaranbaru dan klik tombol simpan | Berhasil |
| *User* menekan tombol *delete* pada baris aplikasi lamaran tertentu dan melihat informasi sebuah aplikasi lamaran secara lengkap | Berhasil |
| 9. | Mengizinkan *User* untuk memasukkan bobot kriteria. | *User* menekan tombol *update*, kemudian memasukkan data bobot kriteriadan klik tombol simpan | Berhasil |
| 10. | Mengizinkan *User* untuk memilih daftar pelamar untuk diseleksi. | *User* memilih posisi lowongan yang akan diseleksi pelamarnya, kemudian menekan tombol *search.* | Berhasil |
| 11. | Mengizinkan *User* untuk melihat hasil pemeringkatan. | *User* menekan tombol Lakukan seleksi | Berhasil |
| 12. | Mengizinkan *User* untuk melakukan *logout* dari sistem. | *User* menekan menu bertuliskan nama *user* pada bagian *header*. Kemudian klik tombol *sign out.* | Berhasil |

# BAB 5 SIMPULAN DAN SARAN

**5.1 Simpulan**

**5.2 Saran**

**DAFTAR PUSTAKA**

Armstrong, M., & Taylor, S. (2014). *Armstrong's Handbook Of Human Resource Management Practice* (13th ed.). London: Kogan Page.

Bulmash, J., Chhinzer, N., & Speers, E. (2010). *Strategic Planning for Human Resources* (1st ed.). Toronto: McGraw-Hill Ryerson.

Career Builder. (2014, Agustus 7). *Fifty-eight Percent of Employers Have Caught a Lie on a Resume, According to a New CareerBuilder Survey*. Dipetik Mei 31, 2016, dari Career Builder: http://www.careerbuilder.com/share/aboutus/pressreleasesdetail.aspx?sd=8%2F7%2F2014&id=pr837&ed=12%2F31%2F2014

Chack, D. K. (2013, July). Decision Support System For Human Resource Management Of The Organization. *International Journal of Management Research and Business Strategy (IJMRBS), 2*.

Diah R., A. K. (2013). Sistem Pendukung Keputusan Rekrutmen Karyawan Produksi Menggunakan Metode Wighted Product Pada PT.Ploss Asia Semarang. *Skripsi, Fakultas Ilmu Komputer*.

Effendy, D. A., & Irawan, R. H. (2015). Uji Sensitivitas WP, SAW, dan TOPSIS Dalam Menentukan Titik Lokasi Repeater Internet Wireless. *Seminar Nasional Teknologi Informasi dan Multimedia*, 85-90.

Kendall, K. E., & Kendall, J. E. (2011). *Systems Analysis and Design* (8th ed.). (E. Svendsen, Penyunt.) Upper Saddle River: Prentice Hall.

Mathis, R. L., & Jack, J. H. (2010). *Human Resource Management* (9th ed.). Amerika Serikat: Cengage Learning.

Purnamasari, R. A. (2015, Juni). Penentuan Penerima Beasiswa Dengan Metode Simple Additive Weighting Dan Metode The Distance To The Ideal Alternative. *Universitas Jember*.

Tran, P. N., & Boukhatem, N. (2008, September 17). The Distance To The Ideal Alternative (DiA) Algorithm for Interface Selection in Heterogenous Wireless Network. *ACM Mobility Management and Wireless Access*.

**Lampiran** **1 – Wawancara**

Di bawah ini merupakan rangkuman wawancara yang penulis (selanjutnya disebut dengan P) lakukan dengan Bapak Vidhestira Dwimadia (selanjutnya disebut dengan VD) selaku *Human Capital Division Head* di perusahaan XYZ pada hari Jumat, 20 Mei 2016 dan Rabu, 27 Juli 2016.

P : Berapa jumlah karyawan yang dimiliki oleh perusahaan?

VD : Terdapat 3 jenis karyawan, yaitu karyawan kontrak, permanen, dan mitra. Total karyawan kontrak dan permanen hingga saat ini kurang lebih sekitar 3800 orang. Jumlah karyawan mitra kurang lebih sekitar 900 orang. Jadi, total karyawan keseluruhan kurang lebih 4700 orang.

P : Berapa banyak cabang yang dimiliki oleh perusahaan?

VD : Terdapat kurang lebih 160 cabang yang tersebar di seluruh Indonesia.

P : Bagaimana cara untuk mengetahui apakah kebutuhan di cabang dengan *Branch model* sudah sesuai atau belum?

VD : Caranya mengetahui sesuai atau tidaknya jumlah *Branch model* dapat dilihat dari dua sisi, yaitu kualifikasi dan pengontrolan jumlahnya. Dari segi kualifikasi, perusahaan telah memiliki standar kualifikasi minimal yang harus dimiliki oleh orang-orang di cabang. Bicara soal kualifikasi terdapat beberapa kriteria, sebagai contoh usia dan fasilitas yang dimiliki. Proses seleksi dimulai dari hal itu. Setelah seleksi dilakukan, akan dilakukan *desk call* dari kantor pusat untuk verifikasi kesesuaian kualifikasi.

Kalau untuk pengontrolan jumlah, terdapat sistem yang membantu pelaksanaannya. Sistem memberikan informasi kebutuhan *manpower* di cabang dan juga informasi perencanaan SDM yang telah dibuat sebelumnya. Jadi, misalnya ketika cabang mengajukan seorang karyawan, sistem memberikan informasi apakah jumlahnya sudah sesuai atau belum berdasarkan perencanaan SDM yang sudah dibuat.

P : Apa kendala yang ditemukan saat melakukan proses ini dengan bantuan sistem?

VD : Secara sistemnya sendiri, menurut saya sudah cukup baik. Tetapi bicara lebih lanjut lagi tentang sistem juga membahas masalah kualifikasi yang tadi itu. Jadi isunya kalau buat saya ke depan adalah soal produktivitas. Jadi, bagaimana caranya dari awal bibitnya kita rekrut orang-orang yang bagus. Kemudian kita didik dia sampai nanti benar-benar orang yang kita dapat adalah orang-orang yang sesuai kualifikasi. Sehingga nanti terbentuk sistem yang memang sudah sesuai dengan *Branch model* yang sudah kita persyaratkan. Kalau sekarang masih banyak berapa kejadian, kami akui ada beberapa deviasi. Jadi artinya mungkin secara kualifikasi tidak masuk, karena cabang butuh orang jadi kita rekrut.

P : Bagaimana alur perusahaan dalam melakukan rekrutmen?

VD : Pelaksanaan rekrutmen dibedakan menjadi tiga, yaitu untuk cabang, kantor pusat, dan *management trainee*. Saat ini, untuk di cabang lebih simpel alurnya. Jadi hanya proses verifikasi data, *interview,* kemudian finalisasi. Jadi penandatanganan PKWT, *offering*, dan lainnya. Itu untuk yang di cabang.

Kalau untuk kantor pusat ialah seleksi administrasi, proses psikotes, kemudian *interview* berjenjang. Ada *interview* dengan *user* dan *interview* dengan HRD. Baru tahap terakhir finalisasi.

Kalau proyek *management trainee* lebih kompleks lagi. Ada proses *interview* 1, *interview* 2, psikotes, kemudian *leaderless group discussion*, proses *offering*, *medical check-up*, baru *onboarding*.

P : Rata-rata berapa banyak orang yang mendaftar pada lowongan pekerjaan saat rekrutmen dilaksanakan?

VD : Kalau pastinya berapa banyak kita berbeda-beda setiap bulan. Karena setiap bulan pasti akan ikut *jobfair*. Itu *range*-nya kalau *jobfair*, misal di Jogja ada hingga 300 orang, begitu juga di Sumatera. Tetapi ada juga yang hanya 70 orang. Untuk ODP khususnya. Kalau untuk yang lain-lain pastinya di kantor pusat, rata-rata yang kita panggil kalau di kantor pusat itu 1-15 orang per satu jabatan atau pekerjaan. Kalau sekarang misal ada 5 lowongan yang dibuka berarti ada 75 orang yang akan dipanggil. Belum ditambah yang ODP. Di daerah beda lagi ceritanya. Kalau di daerah belum ada pemantauan khusus berapa yang dipanggil, berapa yang datang.

Rata-rata penerimaan kita untuk *field* *staff* itu sekitar 150 untuk sebulan, tambah yang lain-lain hampir 250 lah ya dalam satu bulan, nasional.

P : Bagaimana alur pembukaan lowongan?

VD : Terbagi menjadi dua, yaitu internal dan eksternal. Untuk internal, setiap awal tahun-kan punya *manpower plan*. Setiap divisi itu punya MPP. MPP itu jumlah-jumlah orang yang dibutuhkan perusahaan sampai 1 tahun ke depan. Dari MPP itu nanti kalau misalnya ada proses perekrutan, mereka akan mengajukan ada *form* namanya *form* pengajuan karyawan (FPK). FPK akan berisi kualifikasi akan kebutuhan orang itu apa saja, contohnya usia, pendidikan, *skill*-nya yang dibutuhkan apa. Itulah yang digunakan menjadi dasar mencari orang saat rekrutmen. Jadi kualifikasi-kualifikasinya seperti itu.

Setelah menerima *form* yang ditandatangani oleh kepala divisi sampai direksi, setelah itu baru dilakukan proses *sourcing* atau publikasi*.* Publikasi bisa dua cara, internal atau eksternal. Kalau internal, iklan dipasang di dalam kantor sendiri. Jadi misalkan kita ada lowongan A B C D, ada yang berminat tidak saudara atau keluarga memenuhi kualifikasi yang dimaksud.

Kalau eksternal, bisa pakai cara yang berbayar atau tidak berbayar. Kalau yang berbayar kita pakai *jobstreet*, *jobsdb*, *careerbuilder*, koran kalau diperlukan. Kalau tidak berbayar, gampangnya sekarang adalah *social media*, *facebook*, *twitter*, *instagram*, kemudian juga *website*-*website* yang gratis atau *free*, kita juga kerja sama dengan kampus atau SMK dan SMA. Dari situ baru ketika ada CV dari sumber tadi, baru dilakukan seleksi administrasi. Apakah kualifikasi cocok dengan yang disyaratkan. Setelah itu kita akan panggil *interview*. Pertama kali oleh HRD, lalu *user*, dan adakan psikotes. Atau *interview* dulu baru psikotes.

P : Bagaimana alur melakukan penyaringan lamaran yang masuk? Kriteria apa saja yang dipertimbangkan dalam penilaian penyaringan lamaran?

VD : Kriteria yang utama ada tiga, usia, pendidikan dan *skill*. Jadi yang dari kualifikasi biasa kita persyaratkan tiga tadi. Dari usia, pengalaman kalau memang perlu ada, lalu ada *skill*. Itu aja sih, ga ada yang lain. Kalau memang ada yang lain mungkin *skill*-*skill* khusus. Contohnya kalau IT, *skill*-nya kan butuh PHP, JAVA. Kalau yang spesifik misalnya suka ada yang minta dari kampus tertentu. Kalau itu saya akan tetap bandingkan dengan kandidat lainnya. Bukan berarti harga mati harus mengambil dari satu kampus tertentu sehingga menghilangkan peluang untuk yang lain.

Jenis kelamin dan gaji yang diharapkan tidak termasuk kriteria. Sebenarnya kalau gaji sudah ada *grading* sebelumnya. Jadi sebenarnya tidak masuk tahapan ini. Sudah ditentukan sebelumnya. Untuk tiap-tiap golongan sudah ada *range* gaji tertentu.

Kalau bicara soal prioritas kriteria, tergantung kebutuhan yang diminta oleh *user*. Misalnya kaya tadi IT, mencari *Programmer*, berarti mereka ingin *programming skill*-nya yang oke. Kalau akunting, ada spesifik *excel*-nya. Kalau prioritas menurut saya, kualifikasi semua saya akan usaha untuk penuhi. Kalau masalah mana yang penting dari lainnya, kalau buat saya adalah kompetensi balik lagi. Jadi kalau orang yang pada dasarnya kompeten (pernah menduduki jabatan, atau kompetensi dari atribut perilaku misal komunikatif).

Dasarnya dari kualifikasi, awalnya pasti mulai di awal dengan usia. Kita akan *screening* di usia dulu kalau terlalu jauh di luar *range* kita langsung tidak terima. Kalau di *jobstreet* lebih gampang, ada dari usia dan *salary expectation* kalau di *jobstreet*. Kalau di CV kan tidak menuliskan gaji. Biasanya kita prioritas di usia dan *skill*-nya cocok apa tidak.

P : Siapa yang bertanggung jawab melaksanakan tugas ini (seleksi administratif)?

VD : Di tim saya ada bagian rekrutmen dan *assessment*. Jadi bagian rekrutmen adalah dia yang menyeleksi orang-orang berdasarkan seleksi administratif atau psikologis. Anggota rekrutmen dan *assessment* di pusat ada dua, di cabang ada 4, jadi total 6. Di cabang regional yang menangani cabang kecil di Jawa, Sumatera, dll.

P : Berapa lama waktu yang dibutuhkan untuk melakukan proses penyaringan lamaran?

VD : Hingga saat ini penyaringan lamaran seleksi administratif dilakukan secara manual. Kita pasang iklan hari ini, kemudian terima CV. Kalau satu CV sekitar 5 menit selesai paling lama.

P : Apakah terdapat kendala ketika melakukan proses penyaringan lamaran melalui situs lowongan kerja yang tadi disebutkan?

VD : Sistem padasitus lowongan kerja itu bagus bisa memberikan *rating* kecocokan. Namun, kadang saya suka liat ya yang *rejected* di sistem. Memang yang namanya sistem itu kaku. Ada yang *rejected* itu gini, kita cari umur 30, dia umur 31, tapi secara kompetensinya dia memenuhi ya saya panggil, orang bedanya setahun doang.

Misal gaji, kalau misal *expected salary*-nya ga *matching* udah langsung *reject* aja. Masuk ke *rejected*. Yang *must have* bisa dikunci di sistem, usia dan gaji.

Mengetahui,

( Vidhestira Dwimadia )

Human Capital Div. Head

**Lampiran** **2 – *Software Requirement Specification* (SRS)**

**Implementasi Metode *The Distance To The Ideal Alternative* Pada Sistem Informasi Perencanaan SDM di PT. XYZ**

**Software Requirement Specification**

**Versi 1.0**

**9 Agustus 2016**

**Addina Nuriyanti Rahmi**

**Software Engineer**

Dipersiapkan untuk

Kelengkapan Tugas Akhir Informatika Universitas Bakrie

Pembimbing : Yusuf Lestanto

1. **PENDAHULUAN**

Dokumen SRS berikut ini berisi penjelasan mengenai spesifikasi kebutuhan yang diperlukan dalam pengembangan modul rekrutmen dari Sistem Informasi Perencanaan SDM berbasis *web* untuk Departemen *Recruitment* dan *Assessment* PT. XYZ. Isi dokumen ini dibagi menjadi tiga bagian besar, yaitu Pendahuluan, Gambaran Umum, dan Spesifikasi Kebutuhan. Dokumen ini dibuat berdasarkan standar penulisan SRS IEEE – 830.

* 1. **Tujuan**

Penulisan dokumen yang berisi penjelasan spesifikasi kebutuhan sistem bertujuan untuk mempermudah pembaca dalam memahami serta memberikan pengetahuan terkait sistem yang akan dibangun.

* 1. **Ruang Lingkup**

Dokumen SRS ini akan menjelaskan spesifikasi kebutuhan dari Modul Rekrutmen pada Sistem Informasi Perencanaan SDM*.* Modul Rekrutmen dibuat untuk membantu Departemen *Recruitment and Assessment* dalam memanajemen informasi Lowongan Kerja, Pelamar, serta Aplikasi Lamaran yang masuk ke perusahaan. Modul Rekrutmen juga dilengkapi fitur Seleksi Resume yang mampu membandingkan dan menilai kecocokan kualifikasi pelamar dengan spesifikasi posisi yang dilamar.

Metode DiAakan diterapkan ke dalam fitur Seleksi Resume. Metode ini akan memberikan rekomendasi pelamar yang lebih unggul dari pelamar lainnya dinilai dari beberapa kriteria. Kriteria yang digunakan ialah usia, pendidikan, jenis kelamin, kemampuan, pengalaman, dan gaji yang diharapkan.

Berdasarkan rekomendasi pelamar tersebut, sistem dapat membantu pengambil keputusan untuk menentukan pelamar yang cocok untuk dipanggil ke tahap selanjutnya dari tahap seleksi administratif.

* 1. **Glosarium**

**Tabel 3.1 Daftar istilah beserta definisinya**

|  |  |
| --- | --- |
| **Istilah** | **Definisi** |
| Spesifikasi Posisi |  |
| Kualifikasi Pelamar |  |
| Seleksi Administratif |  |
| DiA |  |

* 1. **Referensi**
  2. ***Overview***

Seperti yang telah disebutkan sebelumnya, dokumen SRS ini tersusun dari tiga bab. Pada Bab 2 akan menjelaskan informasi mengenai gambaran umum sistem yang akan dibangun. Umumnya informasi pada Bab 2 diperuntukkan bagi pengguna sistem. Pada Bab 3 akan menjelaskan secara detail spesifikasi kebutuhan dari sistem yang akan dibangun, di antaranya mencakup antarmuka pengguna dan kebutuhan fungsional sistem. Informasi pada Bab 3 ini akan lebih bermanfaat untuk pengembang sistem.

1. **GAMBARAN UMUM**
   1. **Perspektif Produk**

Sistem Informasi Perencanaan SDM Modul Rekrutmen merupakan sistem berbasis *web* untuk membantu memanajemen data terkait lowongan kerja, pelamar, aplikasi lamaran, serta seleksi resume di PT. XYZ. Sistem ini menerapkan metode DiA untuk melakukan penilaian dengan membandingkan kualifikasi pelamar terhadap kualifikasi yang dibutuhkan sebuah posisi. Hasil penilaian masing-masing pelamar akan diurutkan untuk mendapatkan rekomendasi pelamar yang lebih unggul dari yang lain. Pengguna dari sistem ini adalah pegawai Departemen *Recruitment* dan *Assessment* pada PT. XYZ.

* 1. **Fungsi Produk**

Fungsi utama dari sistem berdasarkan kebutuhan pengguna ialah sebagai berikut :

* Menyediakan fungsi untuk melakukan manajemen data terkait rekrutmen, seperti Lowongan Kerja, Pelamar, Aplikasi Lamaran, dan Spesifikasi Posisi.
* Memberikan rekomendasi pelamar yang lebih unggul dari segi kualifikasi pada tahap seleksi administratif untuk dipanggil mengikuti proses seleksi selanjutnya.
  1. **Karakteristik Pengguna**

Sistem ini diperuntukkan bagi staf Departemen *Recruitment and Assessment* pada PT XYZ. Agar dapat mengoperasikan sistem dengan baik, pengguna harus memahami alur umum proses rekrutmen. Pengguna juga diharuskan memiliki pengalaman menggunakan *web browser*. *Web browser* digunakan untuk mengakses sistem melalui jaringan *intranet* perusahaan*.*

* 1. **Batasan Umum**

Berikut ini merupakan batasan umum dari sistem :

* Terlepas dari proses rekrutmen secara keseluruhan, sistem hanya mengakomodasi manajemen data lowongan kerja, pelamar, aplikasi lamaran, dan seleksi resume.
* Sistem yang dibangun berbentuk aplikasi *web* menggunakan bahasa pemrograman PHP, HTML, CSS, dan Javascript. Menggunakan XAMPP sebagai *web server* dan *database server*.
* Data yang diperlukan untuk melakukan seleksi resume akan dimasukkan secara manual oleh pengguna. Data yang diperlukan termasuk *jobtitle*, posisi, kemampuan, kualifikasi, lokasi, lowongan kerja, pelamar, lamaran, serta bobot kriteria penilaian untuk seleksi.
* Metode DiA diterapkan pada bagian Seleksi Resume untuk memberikan pemeringkatan serta rekomendasi pelamar yang lebih unggul dari segi kualifikasi dibandingkan pelamar yang lainnya.
* Kriteria yang digunakan untuk penilaian ialah usia, pendidikan, jenis kelamin, kemampuan, pengalaman, serta gaji yang diharapkan. Kriteria tidak dapat ditambahkan atau dikurangkan.
  1. **Asumsi dan Ketergantungan**
* Pengguna harus memiliki kemampuan untuk mengoperasikan *web browser.*
* Data posisi, lowongan kerja, pelamar, aplikasi lamaran, serta bobot kriteria tidak boleh kosong dan harus diisi dengan lengkap agar proses seleksi resume dapat dilakukan dengan baik.
* Pengalaman pelamar yang dimasukkan ke sistem hanyalah pengalaman yang sesuai dengan posisi dilamar.

1. **SPESIFIKASI KEBUTUHAN**
   1. **Antarmuka Pengguna**

Modul Rekrutmen terdiri dari dua menu utama, yaitu Data Master dan Rekrutmen. Menu tersebut masing-masing memiliki beberapa sub-menu di bawah mereka dimana tiap sub-menu memiliki beberapa halaman antarmuka. Pada Tabel 3.1 terdapat daftar sub-menu dari masing-masing menu. Kecuali untuk sub-menu Seleksi Resume, sub-menu pada Tabel 3.1 memiliki fungsi yang sama, yaitu untuk melakukan manajemen data dari nama sub-menu tersebut. Manajemen data terdiri dari operasi menampilkan daftar seluruh data *(index)*, membuat data *(create)*, mengubah data *(update)*, menampilkan data spesifik *(view)*, serta menghapus data *(delete)*. Maka dari itu, sub-menu dengan fungsi manajemen data memiliki rancangan antarmuka dengan format yang sama untuk halaman *index, create, view,* dan *update*. Sedangkan, untuk operasi *delete* tidak memiliki halaman khusus, melainkan hanya memberikan *pop-up*.

**Tabel 3.1 Daftar Menu dan Sub-Menu yang dimilikinya.**

|  |  |
| --- | --- |
| **Menu** | **Sub-Menu yang Dimiliki** |
| Data Master | * Jobtitle * Posisi * Requirement * Lokasi |
| Rekrutmen | * Lowongan Pekerjaan * Pelamar * Lamaran * Seleksi Resume |

Antarmuka ditampilkan melalui *web browser* seperti *Mozilla Firefox* atau *Google Chrome* namun tidak terbatas pada yang telah disebutkan. Persyaratan antarmuka yang diharapkan pengguna ialah tampilan yang *user-friendly.*

**Gambar 3.1 Rancangan Antarmuka Halaman Login**

Seperti yang dapat dilihat pada Gambar 3.1, terdapat dua buah *field* untuk memasukkan data yaitu *username* dan *password* pengguna pada halaman *login*. Tombol *Login* berguna untuk mengirim data hasil *input* untuk kemudian divalidasi oleh sistem. Apabila data sesuai, maka pengguna akan diarahkan ke halaman utama sistem.

**Gambar 3.2 Rancangan Antarmuka Halaman Melihat Daftar Lowongan Kerja**

Gambar 3.2 merupakan rancangan antarmuka halaman *index* dari sub-menu Lowongan Kerja. Pada halaman ini terdapat sebuah tombol Tambah untuk membuat lowongan kerja baru. Kemudian terdapat sebuah tabel untuk menampilkan daftar lowongan kerja beserta informasi mengenai lowongan tersebut. Tabel tersebut memiliki kolom Aksi yang berisi tiga buah tombol untuk melakukan operasi *view, update*, dan *delete* terhadap data pada baris tabel tertentu.

Rancangan halaman *index* ini juga berlaku untuk tampilan halaman *index* dari sub-menu lainnya yang berfungsi untuk manajemen data. Perbedaannya hanyalah pada data yang akan ditampilkan.

**Gambar 3.3 Rancangan Antarmuka Halaman Membuat / Mengubah Pelamar**

Gambar 3.3 merupakan rancangan antarmuka halaman *create* / *update* dari sub-menu Pelamar. Pada halaman ini terdapat sebuah *form* yang terdiri dari kumpulan label beserta elemen *input* untuk diisi oleh pengguna. Kemudian terdapat dua buah tombol, yaitu kembali dan simpan.

Rancangan halaman *create/update* ini juga berlaku untuk tampilan halaman *create/update* dari sub-menu lainnya yang berfungsi untuk manajemen data. Perbedaannya hanyalah pada daftar data yang perlu dilengkapi.

**Gambar 3.4 Rancangan Antarmuka Melihat Sebuah Aplikasi Lamaran**

Gambar 3.4 merupakan rancangan antarmuka halaman *view* dari sub-menu Aplikasi Lamaran. Pada halaman ini terdapat sebuah tabel yang berisi data-data terkait nama sub-menu tersebut. Kemudian terdapat sebuah tombol, yaitu kembali.

Rancangan halaman *view* ini juga berlaku untuk tampilan halaman *view* dari sub-menu lainnya yang berfungsi untuk manajemen data. Perbedaannya hanyalah pada daftar data yang akan ditampilkan.

**Gambar 3.5 Rancangan Antarmuka Halaman Awal Seleksi Resume**

Gambar 3.5 merupakan rancangan antarmuka halaman awaldari sub-menu Seleksi Resume. Pada halaman ini terdapat sebuah *drop down list* dari daftar lowongan kerja. Sebuah tombol *Search* berfungsi untuk melakukan proses pencarian pelamar dari lowongan kerja tersebut. Di bawahnya terdapat sebuah tabel yang akan menampilkan daftar pelamar yang melamar pada lowongan kerja tersebut. Daftar nama para pelamar yang ditampilkan tersebut akan menjadi daftar alternatif yang diseleksi nantinya. Tombol Simpan & Pilih Kriteria akan membawa pengguna ke tahap selanjutnya dalam proses seleksi resume.

**Gambar 3.6 Rancangan Antarmuka Halaman Lihat Nilai Alternatif**

Gambar 3.6 merupakan rancangan antarmuka halaman lihat nilai alternatif dari sub-menu Seleksi Resume. Pada halaman ini terdapat sebuah tabelyang merangkum data-data para alternatif atau pelamar. Tabel akan menampilkan data nama, jenis kelamin, usia, pendidikan, pengalaman kerja, kemampuan, dan gaji yang diharapkan dari masing-masing alternatif. Terdapat tombol kembali untuk kembali ke halaman awal seleksi resume. Sedangkan tombol lakukan seleksi akan membawa pengguna ke halaman selanjutnya pada tahap seleksi resume.

**Gambar 3.7 Rancangan Antarmuka Halaman Hasil Seleksi Resume**

Gambar 3.7 merupakan rancangan antarmuka halaman hasil seleksi resume. Pada halaman ini terdapat informasi mengenai posisi yang dilamar oleh daftar alternatif, di antaranya ialah nama posisi dan informasi kualifikasi yang dibutuhkan oleh posisi tersebut.

**Gambar 3.8 Rancangan Pop-Up Untuk Operasi *Delete***

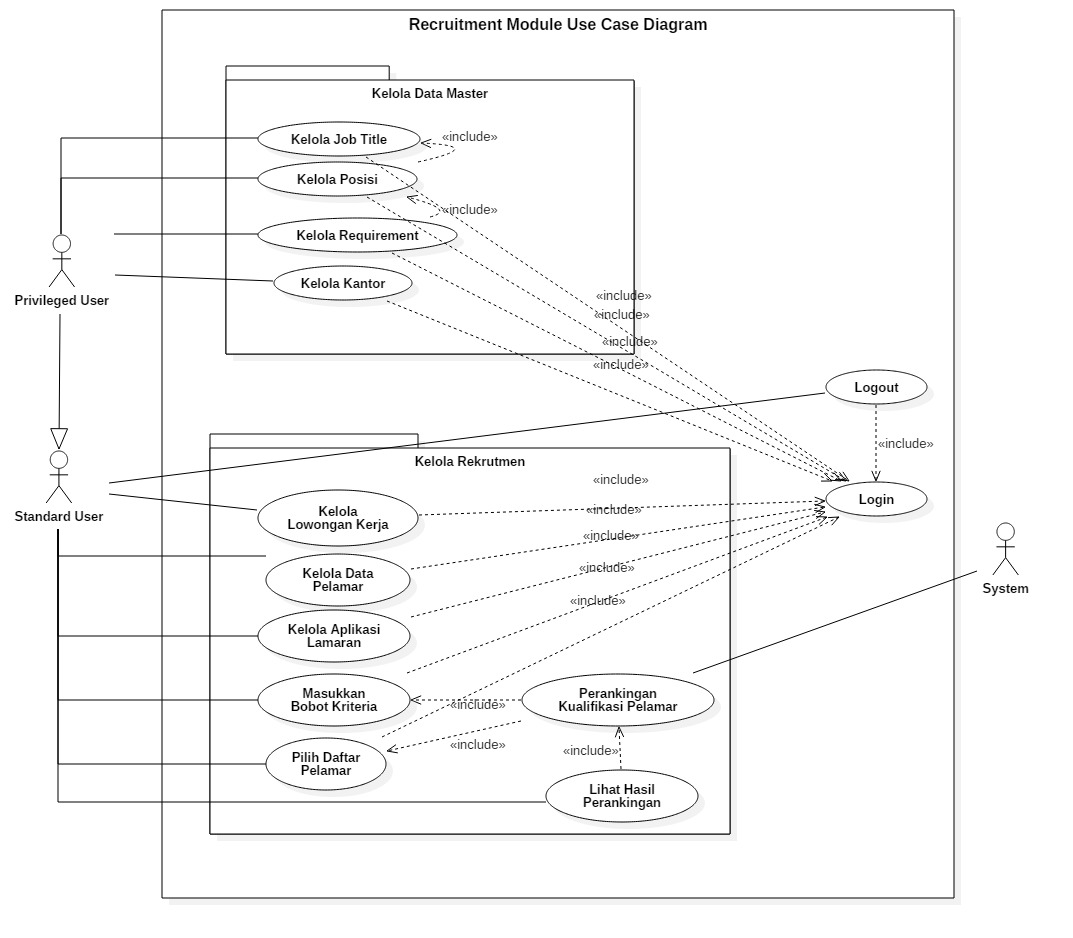
Gambar 3.8 merupakan rancangan tampilan pop-up yang akan muncul ketika tombol *delete* ditekan.

* 1. **Kebutuhan *Hardware***

Berikut merupakan infromasi *hardware* yang dibutuhkan untuk membangun sistem :

1. Sebuah komputer *laptop* atau PC digunakan untuk proses merancang, membangun, serta menjalankan sistem dengan spesifikasi sebagai berikut :
   * Prosesor Intel Pentium @2.10Ghz
   * Memori RAM 2 GB
   * 64-bit *Windows Operating System*
   * *Hard Disk* 320 GB
   1. **Kebutuhan *Software***

Berikut merupakan daftar *software* yangdibutuhkan untuk menjalankan sistem :

* XAMPP untuk menjalankan *web server* Apache dan RDBMSMySQL*.*
* *Web browser* *Mozilla Firefox.*
* Sistem Operasi *Windows* 7.
* Bahasa pemrograman PHP, HTML, CSS, Javascript.
* Yii dan JQuery sebagai framework PHP dan Javascript secara berurutan.
* Admin LTE 2 sebagai template tampilan sistem.
  1. **Kebutuhan Fungsional**

**Gambar 3.9 *Use case diagram* sistem**

Pada bagian ini akan dijelaskan mengenai fungsi-fungsi yang dapat dilakukan oleh sistem. Diagram *use case* pada Gambar 3.9 menunjukkan fungsi-fungsi dari sistem yang dapat dilakukan oleh pengguna.

Nama *Use Case* : Login

Aktor : *Standard* *User*

*Pre-condition* : Aktor membuka sistem melalui *web browser*

*Post-condition* : Aktor berhasil *login* dan masuk ke dalam aplikasi

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan *login*.

**Tabel 3.2 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case Login***

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Membuka sistem melalui *web browser* | 2. Menampilkan halaman *login* serta meminta *username* dan *password* |
| 3. Memasukkan *username* dan *password* | 4. Melakukan validasi data yang dimasukkan oleh aktor. |
|  | 5. Jika validasi benar, maka Aktor dapat masuk ke sistem. Jika salah, menampilkan pesan eror. |

Nama *Use Case* : *Logout*

Aktor : *Standard User*

*Pre-condition* : Aktor menekan tombol *logout* pada sistem

*Post-condition* : Aktor telah keluar dari sistem

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan *logout*.

**Tabel 3.3 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case Logout***

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Menekan tombol *logout* pada sistem | 2. Mengakhiri sesi penggunaan sistem milik aktor |
|  | 3. Menampilkan halaman *login* |

Nama *Use Case* : Kelola *Jobtitle*

Aktor : *Privileged User*

*Pre-condition* : Aktor memilih sub-menu *jobtitle*

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola *Jobtitle*. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data *jobtitle*.

**Tabel 3.4 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Kelola *Job Title***

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu *jobtitle.* | 2. Menampilkan daftar seluruh j*obtitle.* |
| 3. Menekan tombol tambah jobtitle. | 4. Menampilkan formulir isian data *jobtitle.* |
| 5. Mengisi data *jobtitle* baru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data *jobtitle* pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar *jobtitle* di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data *jobtitle* yang akan diubah |
| 12. Memasukkan perubahan data *jobtitle* dan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data *jobtitle* pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index jobtitle* |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Kelola Posisi

Aktor : *Privileged User*

*Pre-condition* : Aktor memilih sub-menu posisi

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola Posisi. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data Posisi.

**Tabel 3.5 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Kelola Posisi**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu posisi*.* | 2. Menampilkan daftar seluruh posisi*.* |
| 3. Menekan tombol tambah posisi. | 4. Menampilkan formulir isian data posisi*.* |
| 5. Mengisi data posisibaru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data posisi pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar posisi di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data posisi yang akan diubah |
| 12. Memasukkan perubahan data posisidan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data posisi pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index* posisi |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Kelola *Requirement*

Aktor : *Privileged User*

*Pre-condition* : Aktor memilih sub-menu kualifikasi

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola Kualifikasi. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data Kualifikasi.

**Tabel 3.6 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Kelola *Requirement***

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu kualifikasi*.* | 2. Menampilkan daftar seluruh kualifikasi*.* |
| 3. Menekan tombol tambah kualifikasi. | 4. Menampilkan formulir isian data kualifikasi*.* |
| 5. Mengisi data kualifikasibaru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data kualifikasi pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar kualifikasi di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data kualifikasi yang akan diubah |
| 12. Memasukkan perubahan data kualifikasidan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data kualifikasi pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index* kualifikasi |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Kelola Kantor

Aktor : *Privileged User*

*Pre-condition* : Aktor memilih sub-menu Kantor

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola Kantor. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data Kantor.

**Tabel 3.7 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Kelola Kantor**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu kantor*.* | 2. Menampilkan daftar seluruh kantor*.* |
| 3. Menekan tombol tambah kantor. | 4. Menampilkan formulir isian data kantor*.* |
| 5. Mengisi data kantorbaru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data kantor pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar kantor di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data kantor yang akan diubah |
| 12. Memasukkan perubahan data kantordan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data kantor pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index* kantor |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Kelola Lowongan Kerja

Aktor : *Standard User*

*Pre-condition* : Aktor memilih sub-menu Lowongan Kerja

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola Lowongan Kerja. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data Lowongan Kerja.

**Tabel 3.8 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Lowongan Kerja**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu lowongan kerja*.* | 2. Menampilkan daftar seluruh lowongan kerja*.* |
| 3. Menekan tombol tambah lowongan kerja. | 4. Menampilkan formulir isian data lowongan kerja*.* |
| 5. Mengisi data lowongan kerjabaru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data lowongan kerja pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar lowongan kerja di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data lowongan kerja yang akan diubah |
| 12. Memasukkan perubahan data lowongan kerjadan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data lowongan kerja pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index* lowongan kerja |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Kelola Pelamar

Aktor : *Standard User*

*Pre-condition* : Aktor memilih sub-menu Pelamar

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola Pelamar. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data Pelamar.

**Tabel 3.9 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Kelola Pelamar**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu pelamar*.* | 2. Menampilkan daftar seluruh pelamar*.* |
| 3. Menekan tombol tambah pelamar. | 4. Menampilkan formulir isian data pelamar*.* |
| 5. Mengisi data pelamarbaru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data pelamar pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar pelamar di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data pelamar yang akan diubah |
| 12. Memasukkan perubahan data pelamardan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data pelamar pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index* pelamar |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Kelola Aplikasi Lamaran

Aktor : *Standard User*

*Pre-condition* : Aktor memilih sub-menu Aplikasi Lamaran

*Post-condition* : Sistem menampilkan data yang telah dikelola Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melakukan Kelola Aplikasi Lamaran. Aktivitas ini terdiri dari membuat, melihat, serta menghapus data Aplikasi Lamaran.

**Tabel 3.10 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Kelola Aplikasi Lamaran**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih sub-menu aplikasi lamaran*.* | 2. Menampilkan daftar seluruh aplikasi lamaran*.* |
| 3. Menekan tombol tambah aplikasi lamaran. | 4. Menampilkan formulir isian data aplikasi lamaran*.* |
| 5. Mengisi data aplikasi lamaranbaru dan menekan tombol simpan | 6. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 7. Menampilkan data aplikasi lamaran pada halaman *view*. |
| 8. Menekan tombol kembali | 9. Menampilkan seluruh daftar aplikasi lamaran di halaman *index* |
| 10. Menekan tombol *update* | 11. Menampilkan data aplikasi lamaran yang akan diubah |
| 12. Memasukkan perubahan data aplikasi lamarandan menekan tombol simpan | 13. Melakukan validasi data masukan. Jika tervalidasi, maka akan menyimpan data ke *database.* |
|  | 14.Menampilkan data aplikasi lamaran pada halaman *view*. |
| 15. Menekan tombol kembali | 16. Menampilkan halaman *index* aplikasi lamaran |
| 17. Menekan tombol hapus | 18. Memberikan *pop-up* konfirmasi untuk menghapus data. |
| 19. Menekan tombol yes | 20. Menghapus data dari *database* |

Nama *Use Case* : Masukkan Bobot Kriteria

Aktor : *Standard User*

*Pre-condition* : Aktor telah *login* ke sistem

*Post-condition* : Sistem menampilkan data yang telah masukkan Aktor

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor memasukkan bobot kriteria.

**Tabel 3.11 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Masukkan Bobot Kriteria**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Menekan tombol *update* pada kolom aksi dari kriteria yang ingin diubah | 2. Menampilkan formulir perubahan kriteria |
| 3. Memasukkan bobot kriteria yang baru dan tekan tombol simpan | 3. Menyimpan data yang baru dan menampilkan hasil perubahan. |

Nama *Use Case* : Pilih Daftar Pelamar

Aktor : *Standard User*

*Pre-condition* : Aktor berada pada halaman awal seleksi resume

*Post-condition* : Daftar pelamar berhasil disimpan sistem

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor memilih daftar pelamar untuk diseleksi.

**Tabel 3.12 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Pilih Daftar Pelamar**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Memilih posisi yang mana pelamarnya akan diseleksi, kemudian menekan tombol *search* | 2. Menampilkan daftar pelamar dari posisi tersebut |
|  | 3. Menyimpan daftar pelamar yang dipilih |

Nama *Use Case* : Perankingan Kualifikasi Pelamar

Aktor : Sistem

*Pre-condition* : Daftar pelamar dan tingkat prioritas kriteria sudah ditentukan

*Post-condition* : Hasil pemeringkatan tersimpan ke *database*

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika sistem melakukan pemeringkatan kualifikasi pelamar.

**Tabel 3.13 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Pemeringkatan Kualifikasi Pelamar**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
|  | * + - 1. Mengambil data pelamar yang akan diseleksi |
|  | 2. Melakukan proses perhitungan metode DiA |
|  | 3. Menyimpan hasil perhitungan ke *database* |
|  | 4. Menampilkan hasil pemeringkatan ke pengguna. |

Nama *Use Case* : Lihat Hasil Pemeringkatan

Aktor : *Standard User*

*Pre-condition* : Pemeringkatan kualifikasi pelamar telah dilakukan

*Post-condition* : -

Deskripsi : *Use Case* ini menggambarkan aktivitas ketika aktor melihat hasil pemeringkatan kualifikasi pelamar.

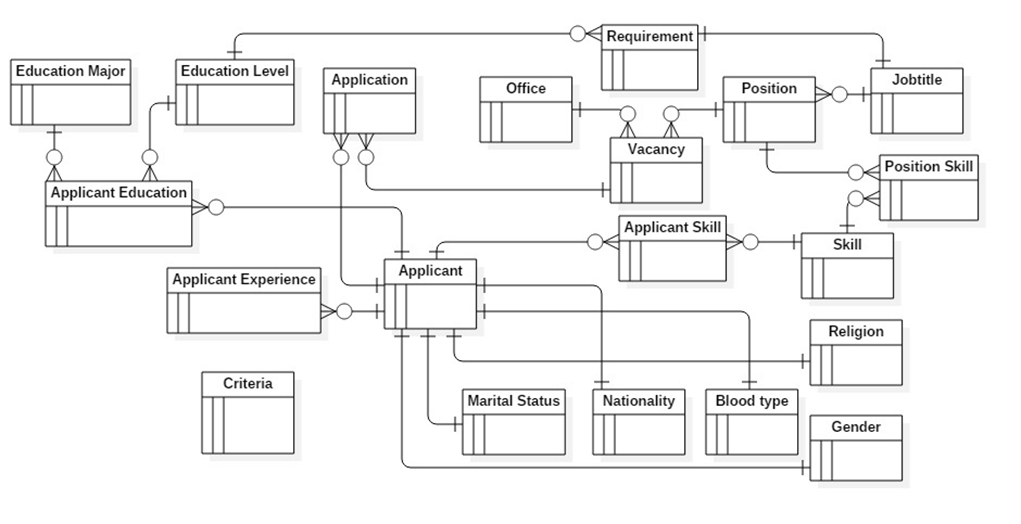
**Tabel 3.14 Deskripsi Aksi Aktor dan Respon Sistem Untuk *Use Case* Lihat Hasil Pemeringkatan**

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
|  | 1. Menampilkan hasil pemeringkatan kualifikasi pelamar |
| 2. Melihat hasil pemeringkatan |  |

* 1. **Kebutuhan Non-Fungsional**

Kebutuhan non-fungsional dari sistem adalah sebagai berikut :

1. Sistem dapat digunakan oleh pengguna yang sudah *login.*
2. Sistem dapat digunakan pada *cross-browser*.
   * 1. ***Conceptual Database Design***

****

Gambar 3.10 *Conceptual Data Model*

Pada Gambar 3.10 dapat dilihat bahwa pada tahap *conceptual design*, rancangan basis data dari sistem terdiri dari 20 tabel. Nama yang melekat pada masing-masing tabel mencirikan data yang disimpan pada tabel tersebut. Terdapat beberapa tabel yang merupakan master, seperti *marital status, nationality, blood type, gender, religion, education major,* dan *education level.* Tabel master tidak dapat dikelola melalui sistem.

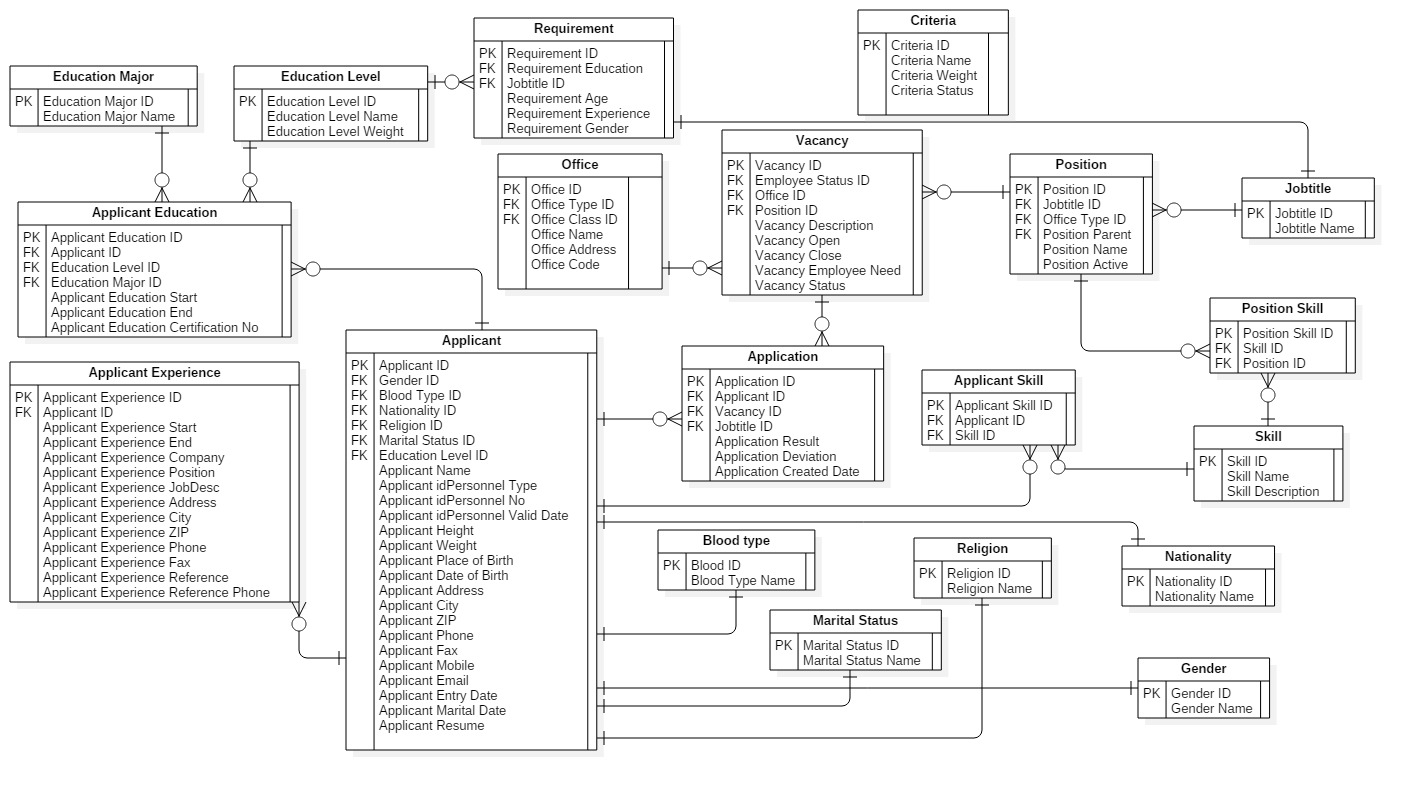
Pada rancangan basis data ini, terdapat beberapa tabel yang bersifat *many-to-many,* yaitu *applicant* *education* (menghubungkan tabel *applicant* dengan *education* *major* dan *education level*)*, application* (menghubungkan tabel *applicant* dengan *vacancy*)*, vacancy* (menghubungkan tabel *office dan position*), *position skill* (menghubungkan tabel *position* dan *skill*)*, applicant skill* (menghubungkan tabel *applicant* dengan *skill*).

* + 1. ***Logical Database Design***

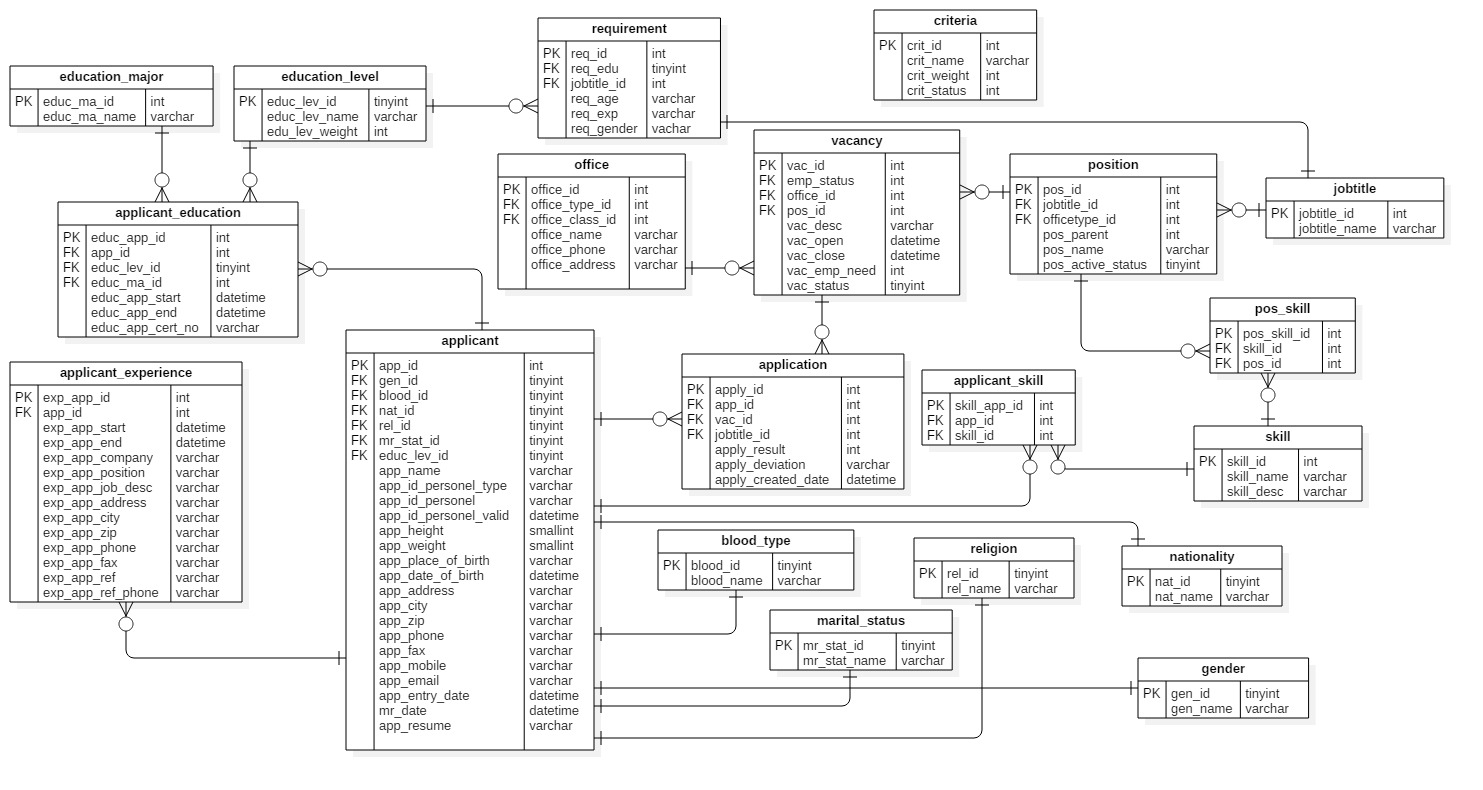
Gambar 3.11 merupakan rancangan *logical design* dari basis data modul rekrutmen pada sistem. Perbedaan *logical design* dengan *conceptual design* ialah keberadaan daftar atribut pada tiap-tiap tabel. Selain adanya daftar atribut, *primary key* serta *foreign key* pada tiap tabel juga telah diidentifikasi. Sedangkan pada rancangan *conceptual design* yang telah teridentifikasi hanya daftar tabel serta hubungan antar tabel satu dengan yang lainnya saja.

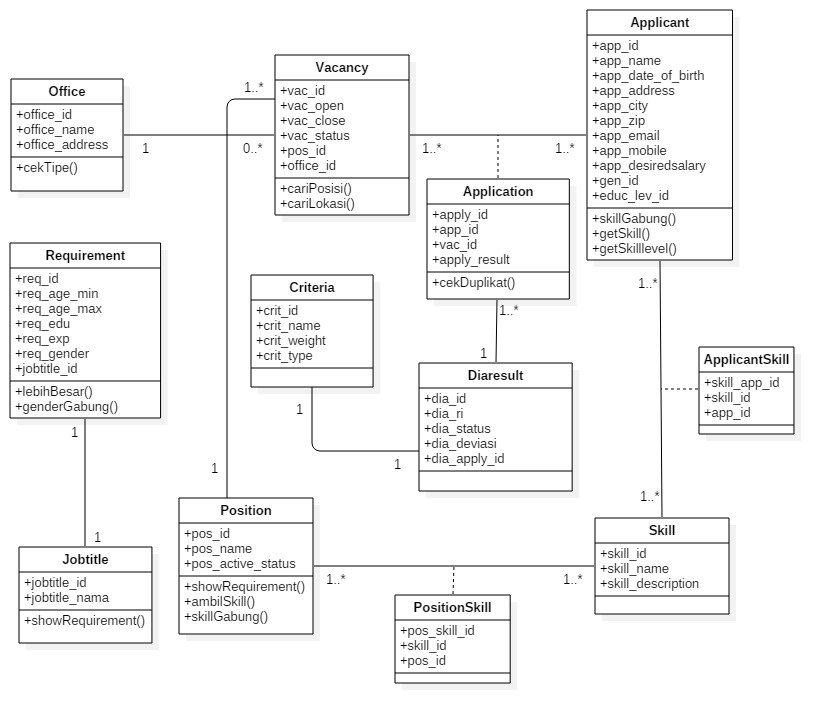
* + 1. ***Physical Database Design***

Gambar 3.12 merupakan rancangan *physical design* dari basis data modul rekrutmen pada sistem. Tidak banyak perbedaan antara *physical design* dengan *logical design,* hanya pada *physical design* masing-masing atribut pada tiap tabel telah memiliki tipe data. Dengan diidentifikasinya daftar tabel, atribut, serta tipe data atribut pada tiap tabel, pada tahap ini, maka rancangan basis data sudah selesai dilakukan. Rancangan ini kemudian akan diimplementasikan untuk mengorganisir penyimpanan data yang dibutuhkan oleh sistem nantinya.

****

Gambar 3.11 *Logical Data Model*

**Gambar 3.12 *Physical Data Model***

* + 1. ***Class Diagram***

**Gambar 3.13 *Physical* Data Model**

*Class diagram* dari modul rekrutmen sistem informasi perencanaan SDM dapat dilihat pada Gambar 4.4. Berikut penjelasan ringkas dari tiap kelas :

1. *Applicant* : Kelas ini mengelola data pelamar, yaitu informasi diri.
2. *Jobtitle* : Kelas ini mengelola data jabatan.
3. *Position* : Kelas ini mengelola data posisi. Kelas posisi memiliki kelas asosiatif dengan kelas *Skill* yang dinamakan *PositionSkill.* Kelas tersebut mengelola data *requirement* kemampuan dari masing-masing posisi.
4. *Office* : Kelas ini mengelola data kantor.
5. *Skill* : Kelas ini mengelola data kemampuan.
6. *Requirement* : Kelas ini mengelola data *requirement* dari *jobtitle*.
7. *Criteria* : Kelas ini mengelola data kriteria yang akan dinilai ketika melakukan pemeringkatan.
8. *Diaresult* : Kelas ini mengelola data hasil pemeringkatan.
9. *Vacancy* : Kelas ini mengelola data lowongan pekerjaan.

**Lampiran** **3 – Rencana Kegiatan**

| No | Jenis Kegiatan | JAN | | | | FEB | | | | MAR | | | | APR | | | | MEI | | | | JUN  2016 | | | | JUL  2016 | | | | AGS  2016 | | | |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2016 | | | | 2016 | | | | 2016 | | | | 2016 | | | | 2016 | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1 | Penentuan Topik |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Perencanaan Kebutuhan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Pengumpulan Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Pengujian dan Evaluasi Error |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 | Penyusunan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |