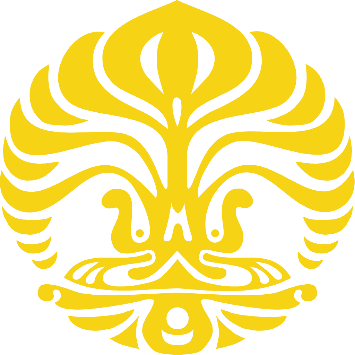
**DESIGN REPORT**

Iterasi 2

**HRRIS**

**Human Resource Recruitment Information System**

****

**ASDOS :** Haidar Aji K.

**KELOMPOK :** PROPENSI B09

**ANGGOTA KELOMPOK :**

1306383092 Dini Seprilia

1306383073 Ferri Saputra

1306382846 Khalila Hunafa Ansor Arifai

1306464133 Nabila Clydea Harahap

1306412092 Yohanes

**FAKULTAS ILMU KOMPUTER**

**PROGRAM STUDI SISTEM INFORMASI**

**DEPOK**

**2016**

**DAFTAR ISI**

[DAFTAR GAMBAR 3](#_Toc451455086)

[DAFTAR TABEL 4](#_Toc451455087)

[A. Structural Model Design 5](#_Toc451455088)

[1. Class Diagram 5](#_Toc451455089)

[B. Behavioral Model 20](#_Toc451455090)

[1. Use Case Realization Specification 20](#_Toc451455091)

[1.1 Use Case: View List of Available Position 20](#_Toc451455092)

[1.2 Use Case: View Available Position Detail Information 21](#_Toc451455093)

[C. Database Design 23](#_Toc451455094)

[1. Database Schema 23](#_Toc451455095)

[2. Table Structures and Attributes 24](#_Toc451455096)

[D. User Interface 29](#_Toc451455097)

[1. Site Map 29](#_Toc451455098)

[2. User Interface Design 31](#_Toc451455099)

[E. Architecture 33](#_Toc451455100)

[1. Physical Architecture Design 33](#_Toc451455101)

[2. Hardware and Software Specification 34](#_Toc451455102)

[F. Test Plan 36](#_Toc451455103)

[1. Unit Testing 36](#_Toc451455104)

[2. User Acceptance test (UAT) 36](#_Toc451455105)

[References 38](#_Toc451455106)

# DAFTAR GAMBAR

[Figure 1. Design Class Diagram 5](#_Toc451455107)

[Figure 2. Skema Relational 23](#_Toc451455108)

[Figure 3. Site Map Applicant 29](#_Toc451455109)

[Figure 4. Site Map Admin 29](#_Toc451455110)

[Figure 5. Site Map Recruiter 30](#_Toc451455111)

[Figure 6. Site Map HR 30](#_Toc451455112)

[Figure 7. Wireframe List of Available Position 31](#_Toc451455113)

[Figure 8. Wireframe Available Position Detail 32](#_Toc451455114)

[Figure 9. Three Tier Client Structure 33](#_Toc451455115)

# DAFTAR TABEL

[Table 1. Detailed Class Applicant 7](#_Toc451455116)

[Table 2. Detailed Class Applicant Manager 7](#_Toc451455117)

[Table 3. Detailed Class User 8](#_Toc451455118)

[Table 4. Detailed Class User Manager 9](#_Toc451455119)

[Table 5. Detailed Class Divisi 9](#_Toc451455120)

[Table 6. Detailed Class Company 10](#_Toc451455121)

[Table 7. Detailed Class Work Experience 10](#_Toc451455122)

[Table 8. Detailed Class Status Applicant 11](#_Toc451455123)

[Table 9. Detailed Class Status Applicant Manager 11](#_Toc451455124)

[Table 10. Detailed Class Status. 12](#_Toc451455125)

[Table 11. Detailed Class Job Vacant 13](#_Toc451455126)

[Table 12. Detailed Class Job Vacant Manager 13](#_Toc451455127)

[Table 13. Detailed Class Available Schedule 14](#_Toc451455128)

[Table 14. Detailed Class Available Schedule Manager 14](#_Toc451455129)

[Table 15. Detailed Class Report 15](#_Toc451455130)

[Table 16. Detailed Class Report Manager 15](#_Toc451455131)

[Table 17. Detailed Class Interview 16](#_Toc451455132)

[Table 18. Detailed Class Interview Manager 16](#_Toc451455133)

[Table 19. Detailed Class SLA 17](#_Toc451455134)

[Table 20. Detailed Class SLA Manager 17](#_Toc451455135)

[Table 21. Detailed Class Assessment Form 18](#_Toc451455136)

[Table 22. Detailed Class Assessment Form Manager 18](#_Toc451455137)

[Table 23. Detailed Class Competency 19](#_Toc451455138)

[Table 24. Tipe data dan attribut 28](#_Toc451455139)

[Table 26. Hardware and Software Specification 35](#_Toc451455140)

# Structural Model Design

## Class Diagram

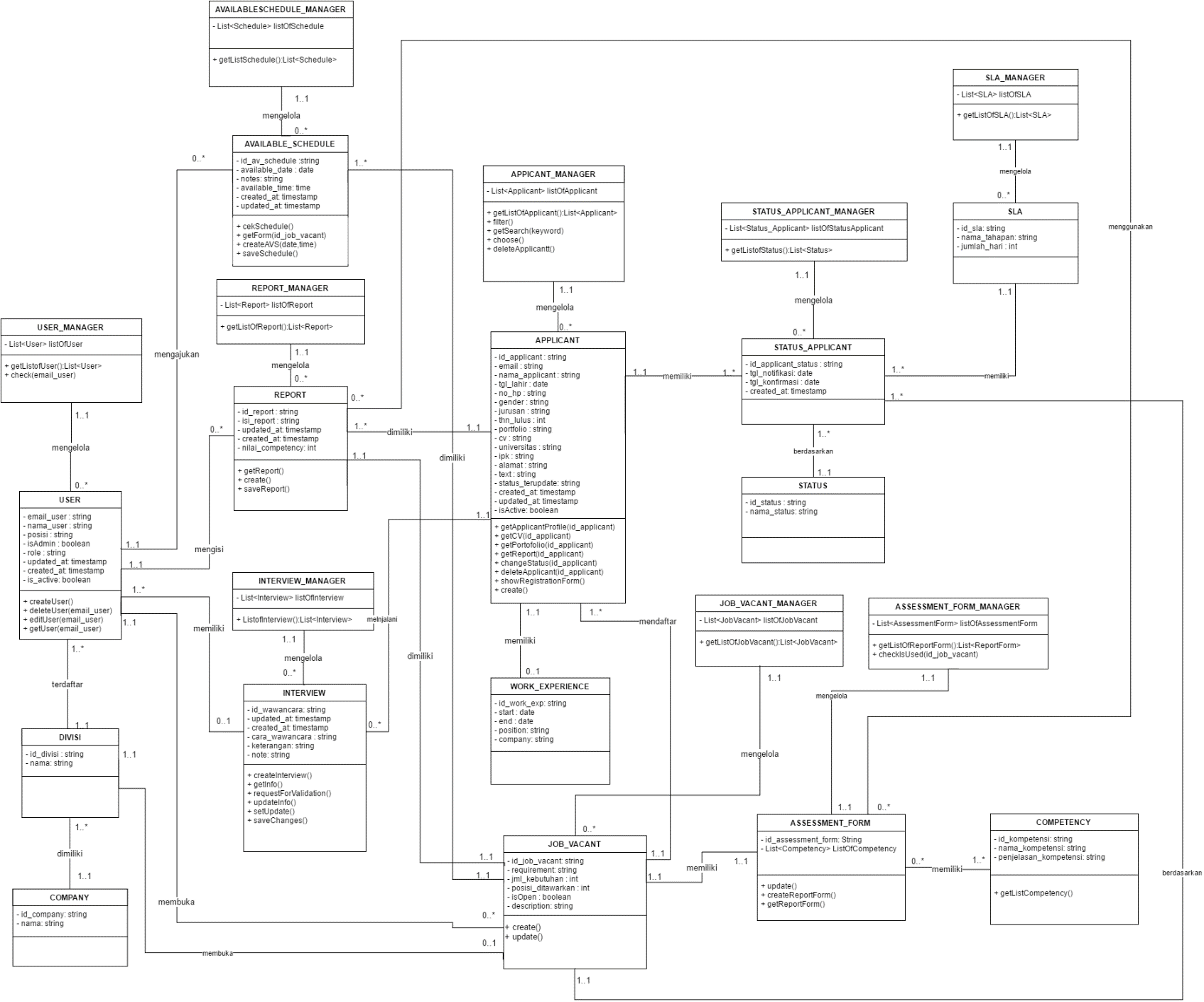


Figure 1. Design Class Diagram

Penjelasan Class Diagram:

* **APPLICANT**

Kelas ini merepresentasikan *applicant* yang mendaftarkan diri pada lowongan pekerjaan. *Method-method* yang terdapat pada kelas *APPLICANT* antara lain: *getApplicantProfile(id\_applicant), getReport(id\_applicant), getCV(id\_applicant), getPortofolio(id\_applicant), changeStatus(id\_applicant), deleteApplicant(id\_applicant), showRegistrationForm(),* dan *create()*. Kelas ini memiliki relasi dengan kelas *APPLICANT\_MANAGER, REPORT, STATUS\_APPLICANT, INTERVIEW, WORK\_EXPERIENCE,* dan *JOB\_VACANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *Id\_Applicant* | *String* | *Primary key* dari kelas *APPLICANT* yang otomatis diberikan oleh sistem. *Not null*. |
| *Email* | *String* | *Not null* dan *unique.* Alamat *email applicant* |
| Nama\_applicant | *String* | *Not null*. Nama lengkap *applicant* |
| Tgl\_Lahir | *date* | *Not null*. Tanggal lahir *applicant* |
| *Gender* | *String* | *Not null*. Jenis kelamin *applicant* |
| Jurusan | *String* | *Not null*. Jurusan yang diambil *applicant* saat kuliah |
| Thn\_lulus | int | *Not null*. Tahun lulus *applicant* dari perkuliahan |
| *Portfolio* | *String* | *Not null*. *Path directory* dari *Portfolio applicant* |
| CV | *String* | *Not null*. *Path directory* dari *CV applicant* |
| Universitas | *String* | Nama universitas tempat *applicant* kuliah |
| IPK | *String* | *Not null*. Indeks prestasi kumulatif *applicant* |
| *No\_hp* | *String* | *Not null*. Nomor *handphone applicant* |
| Alamat | *String* | Alamat rumah *applicant* |
| *Text* | *String* | Keterangan tambahan *applicant* |
| *Status\_terupdate* | *String* | Keterangan status *applicant* terbaru |
| *Created\_at* | *Timestamp* | *Not null.* Tanggal dan waktu data dibuat/*create* |
| *Updated\_at* | *Timestamp* | *Not null.* Tanggal dan waktu data diubah/*update* |
| *isActive* | *boolean* | *Not null.* Status *applicant* apakah aktif atau tidak |

Table 1. Detailed Class Applicant

* ***APPLICANT\_MANAGER***

Kelas ini merepresentasikan kumpulan *applicant* yang mendaftarkan diri pada lowongan pekerjaan. Adapun *method-method* yang dimiliki oleh kelas ini antara lain: *getListOfApplicant(), filter(), getSearch(keyword), choose(),* dan *deleteApplicant().* Kelas *APPLICANT\_MANAGER* hanya memiliki relasi dengan kelas *APPLICANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfApplicant* | *List*<*Applicant*> | Kumpulan dari *applicant* |

Table 2. Detailed Class Applicant Manager

* ***USER***

Kelas ini merepresentasikan pegawai yang akan menggunakan sistem iniyang berperan sebagai *HR, User HR, COO*, atau *Admin*. Method-method yang dimiliki oleh kelas ini antara lain: *createUser(), deleteUser(email\_user), editUser(email\_user),* dan *getUser(email\_user).* Kelas ini memiliki relasi dengan kelas-kelas lainnya, seperti: *USER\_MANAGER, AVAILABLE\_SCHEDULE, REPORT, INTERVIEW, JOB\_VACANT* dan *DIVISI*. Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *Email\_user* | *String* | *Primary key* dari kelas USER yang otomatis diberikan oleh sistem. *Not null*. |
| Nama\_user | *String* | *Not null*. Nama lengkap pegawai/user |
| Posisi | *String* | *Not null*. Jabatan pegawai dalam perusahaan |
| isAdmin | *boolean* | *Not null. Role* user apakah admin atau bukan |
| role | *String* | *Not null. Role user* |
| *Created\_at* | *Timestamp* | *Not null.* Tanggal dan waktu data dibuat/*create* |
| *Updated\_at* | *Timestamp* | *Not null.* Tanggal dan waktu data diubah/*update* |
| *Is\_active* | *boolean* | *Not null.* Status user apakah masih aktif atau tidak |

Table 3. Detailed Class User

* ***USER\_MANAGER***

Kelas ini merepresentasikan kumpulan *user* yang menjadi yang akan menggunakan sistem ini (*HR, User HR, COO* dan *Admin*). Kelas ini mempunyai method yaitu *getListofUser():List<User>* dan *check(email\_user).* Kelas *USER\_MANAGER* hanya memiliki relasi dengan kelas *USER.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| l*istOfUser* | *List*<*User*> | Kumpulan dari *user* |

Table 4. Detailed Class User Manager

* ***DIVISI***

Kelas ini merepresentasikan divisi apa saja yang terdapat pada suatu perusahaan. Kelas ini tidak memiliki method. Kelas ini mempunyai relasi dengan kelas *USER, JOB\_VACANT* dan *COMPANY.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_Divisi | *String* | *Primary key* dari entitas DIVISI yang otomatis diberikan oleh sistem. *not null*. |
| Nama | *String* | *Not null*. Nama divisi |

Table 5. Detailed Class Divisi

* ***COMPANY***

Kelas ini merepresentasikan perusahaan yang akan menggunakan sistem ini, baik itu berupa induk perusahaan atau anak perusahaan. Kelas ini tidak memiliki method. Kelas ini hanya mempunyai relasi dengan kelas *DIVISI.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_*Company* | *String* | *Primary key* dari entitas COMPANY yang otomatis diberikan oleh sistem. *not null*. |
| Nama | *String* | *Not null*. Nama perusahaan |

Table 6. Detailed Class Company

* ***WORK\_EXPERIENCE***

Kelas ini merepresentasikan pengalaman kerja apa saja yang dimiliki *applicant.* Kelas ini tidak memiliki method. Kelas ini hanya mempunyai relasi dengan kelas *APPLICANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_*Work\_Exp* | *String* | *Primary key* dari kelas WORK\_EXPERIENCE yang otomatis diberikan oleh sistem. *Not null*. |
| Position | *String* | *Not null*. Jabatan/Posisi pekerjaan yang pernah diambil *applicant* |
| *Company* | *String* | *Not null*. Perusahaan yang pernah menjadi tempat kerja *applicant* |
| *Period\_of\_Time* | *String* | *Not null*. Jenjang waktu kerja yang sudah dilakukan *applicant* |

Table 7. Detailed Class Work Experience

* ***STATUS\_APPLICANT***

Kelas ini merepresentasikan status *applicant* dari proses *recruitment,* dimulai dati *applicant* tersebut mendaftar sampai *applicant* diterima atau ditolak. Kelas ini tidak memiliki method. Kelas ini mempunyai relasi ke beberapa kelas, diantranya dengan kelas *STATUS\_APPLICANT\_MANAGER, APPLICANT, SLA, STATUS, JOB\_VACANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *Id\_applicant\_status* | *String* | *Primary key* dari kelas STATUS\_APPLICANT yang diberikan oleh sistem. *Not null*. |
| Tgl\_notifikasi | *Date* | Boleh *null.* Tanggal pemberian status |
| Tgl\_konfirmasi | *Date* | Boleh *null.* Tanggal pemberian konfirmasi |
| *Created\_at* | *Timestamp* | *Not null.* Tanggal dan waktu data dibuat/*create* |

Table 8. Detailed Class Status Applicant

* ***STATUS\_APPLICANT\_MANAGER***

Kelas ini merepresentasikan kumpulan status dari *applicant* yang mendaftar sesuai dengan proses *recruitment* yang sedang dijalani. Kelas ini mempunyai method yaitu *getListofUser():List<User>.* Kelas *STATUS\_APPLICANT\_MANAGER* hanya memiliki relasi dengan kelas *STATUS\_APPLICANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfStatusApplicant* | List<Status\_Applicant> | kumpulan status |

Table 9. Detailed Class Status Applicant Manager

* ***STATUS***
* Kelas ini merepresentasikan kumpulan dari status *applicant*. Kelas ini mempunyai relasi ke *STATUS\_APPLICANT.* Kelas ini tidak memiliki method.Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *Id\_status* | *String* | *Primary key* dari kelas STATUS\_APPLICANT yang diberikan oleh sistem. *Not null*. |
| Nama\_status | *Date* | Boleh *null.* Tanggal pemberian status |

Table 10. Detailed Class Status.

* ***JOB\_VACANT***

Kelas ini merepresentasikan lowongan pekerjaan yang dibuka oleh perusahaan. Kelas ini memiliki *method-method* seperti: *ceate (),* dan *update()*. Kelas *JOB\_VACANT* memiliki relasi dengan beberapa kelas seperti: *JOB\_VACANT\_MANAGER, APPLICANT, REPORT, AVAILABLE SCHEDULE, USER, DIVISI* dan *ASSESSMENT\_FORM.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_*Job\_Vacant* | *String* | *Primary key* dari kelas JOB\_VACANT yang otomatis diberikan oleh sistem. *not null*. |
| *Requirement* | *String* | *Not null*. Deskripsi dan spesifikasi pekerjaan. |
| Jml\_Kebutuhan | Int | *Not null*. Jumlah orang yang dibutuhkan untuk mengisi pekerjaan tersebut. |
| Posisi\_ditawarkan | Int | *Not null*. Posisi pekerjaan |
| *isOpen* | *boolean* | *Not null*. Status *job vacant* |
| *description* | *String* | Boleh null. Deskripsi dari suatu *job vacant* |

Table 11. Detailed Class Job Vacant

* ***JOB\_VACANT\_MANAGER***

Kelas ini merepresentasikan kumpulan *job vacant* yang sudah/sedang dibuka oleh perusahaan. Kelas ini mempunyai *method* yaitu *getListofJobVacant():List<JobVacant>.* Kelas *JOB\_VACANT\_MANAGER* hanya memiliki relasi dengan kelas *JOB\_VACANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfJobVacant* | *List*<*JobVacan*t> | Kumpulan dari *job vacant* yang dibuka |

Table 12. Detailed Class Job Vacant Manager

* ***AVAILABLE\_SCHEDULE***

Kelas ini merepresentasikan jadwal yang diajukan *users* untuk melakukan wawancara dengan *applicant*. Kelas ini memiliki beberapa *method,* seperti *cekSchedule(), getForm(id\_job\_vacant), createAVS(date, time)* dan saveSchedule()*.* Kelas ini memiliki relasi dengan beberapa kelas yang lain, seperti: *AVAILABLE\_SCHEDULE\_MANAGER, USER* dan *JOB\_VACANT*. Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_*av\_schedule* | *String* | *Primary key* dari kelas *AVAILABLE\_POSITION* yang otomatis diberikan oleh sistem. *not null*. |
| *Available\_date* | *Date* | *Not null*. Tanggal yang disediakan *users* untuk melakukan wawancara |
| *Notes* | *String* | Boleh *null.* Keterangan tambahan mengenai jadwal wawancara |
| *Available\_time* | *Time* | *Not null.* Tanggal yang disediakan *users* untuk melakukan wawancara |
| *Created\_at* | *Timestamp* | *Not null* |
| *Updated\_at* | *Timestamp* | *Not null* |

Table 13. Detailed Class Available Schedule

* ***AVAILABLE\_SCHEDULE\_MANAGER***

Kelas ini merepresentasikan kumpulan *available schedule* yang direkomendasikan dari *users*. Kelas ini mempunyai *method* yaitu *getListofSchedule():List<Schedule>.* Kelas *AVAILABLE\_SCHEDULE\_MANAGER* hanya memiliki relasi dengan kelas *AVAILABLE\_SCHEDULE.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfSchedule* | *List<Schedule>* | kumpulan *available schedule* |

Table 14. Detailed Class Available Schedule Manager

* ***REPORT***

Kelas ini merepresentasikan laporan penilaian *applicant* yang ditulis oleh *recruiter* (*User HR, HR*, atau *COO*). *Method-method* yang terdapat pada kelas *REPORT* antara lain: *getReport(), create()* dan *saveReport().* Kelas ini memiliki relasi dengan beberapa kelas lain, yaitu dengan kelas *REPORT\_MANAGER, USER, ASSESSMENT\_FORM, APPLICANT* dan *JOB\_VACANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_*Report* | *String* | *Primary key* dari kelas REPORT yang otomatis diberikan oleh sistem. *not null*. |
| Isi\_report | *String* | *Not null*. isi penilaian *applicant* oleh *Recruiter* |
| *Created\_at* | *Timestamp* | *Not null* |
| *Updated\_at* | *Timestamp* | *Not null* |
| *Nilai\_competency* | Int | Boleh *null.* Nilai untuk setiap report |

Table 15. Detailed Class Report

* ***REPORT\_MANAGER***

Kelas ini merepresentasikan kumpulan laporan penilaian (*report.* Kelas ini mempunyai *method* yaitu *getListOfReport():List<Report>.* Kelas *REPORT\_MANAGER* hanya memiliki relasi dengan kelas *REPORT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfReport* | *List<Report>* | kumpulan *report* |

Table 16. Detailed Class Report Manager

* ***INTERVIEW***

Kelas ini merepresentasikan *interview* yang dilakukan antara *users* dan *applicant. Method-method* yang dimiliki oleh kelas ini yaitu *createInterview(), getInfo(), requestForValidation(), updateInfo(), setUpdate()* dan saveChanges()*.* Kelas *INTERVIEW* memiliki relasi dengan kelas *INTERVIEW\_MANAGER, USER* dan *APPLICANT*. Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_*wawancara* | *String* | *Primary key* dari kelas INTERVIEW yang otomatis diberikan oleh sistem. *not null*. |
| *Created\_at* | *Timestamp* | *Not null* |
| *Updated\_at* | *Timestamp* | *Not null* |
| Cara\_Wawancara | *String* | *Not null*. Cara atau media yang dipakai untuk melakukan wawancara |
| Keterangan | *String* | *Boleh null.* Keterangan wawancara |
| *Note* | *String* | Catatan penjelasan wawancara |

Table 17. Detailed Class Interview

* ***INTERVIEW\_MANAGER***

Kelas ini merepresentasikan kumpulan *interview* yang akan dilakukan oleh *user* dan *applicant*. Kelas ini mempunyai *method* yaitu *getListOfInterview():List<Interview>.* Kelas *INTERVIEW\_MANAGER* hanya memiliki relasi dengan kelas *INTERVIEW.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfInterview* | *List<Interview>* | kumpulan *interview* |

Table 18. Detailed Class Interview Manager

* ***SLA***

Kelas ini merepresentasikan status proses *recruitment* yang akan disimpan untuk menentukan *Service Level of Aggrement (SLA)*. Kelas ini tidak memiliki method. Kelas ini berelasi dengan kelas *SLA\_MANAGER* dan *STATUS\_APPLICANT.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_sla | *String* | *Primary key* dari kelas SLA yang otomatis diberikan oleh sistem. *not null*. |
| Nama\_tahapan | *String* | *Not null*. Nama proses SLA |
| Jumlah\_Hari | int | *Not null*. Total hari yang diperbolehkan oleh SLA |

Table 19. Detailed Class SLA

* ***SLA\_MANAGER***

Kelas ini merepresentasikan kumpulan SLA dari proses *recruitment*. Kelas ini mempunyai *method* yaitu *getListOfSLA():List<SLA>.* Kelas *SLA\_MANAGER* hanya memiliki relasi dengan kelas *SLA.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfSLA* | *List<SLA>* | kumpulan SLA |

Table 20. Detailed Class SLA Manager

* ***ASSESSMENT\_FORM***

Kelas ini merepresentasikan *form* yang digunakan untuk pengisian *report* atau penilaian terhadap *applicant*. Kelas ini memiliki beberapa *method,* antara lain *update(), createReportForm(), getReportForm().* Kelas *ASSESSMENT\_FORM* memiliki relasi dengan kelas *ASSESSMENT\_FORM\_MANAGER, JOB\_VACANT,* *REPORT* dan *COMPETENCY.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *Id\_assessment\_form* | *String* | *Primary key* dari kelas ASSESSMENT\_FORM yang otomatis diberikan oleh sistem. *not null*. |
| List<Competency> ListOfCompetency | Objek COMPETENCY | Kumpulan *list of competency* |

Table 21. Detailed Class Assessment Form

* ***ASSESSMENT\_FORM\_MANAGER***

Kelas ini merepresentasikan kumpulan *report form* dari proses *recruitment*. Kelas ini mempunyai *method* yaitu *getListOfReportForm():List<ReportForm>* dan chechIsUsed(id\_job\_vacant)*.* Kelas *ASSESSMENT\_FORM\_MANAGER* hanya memiliki relasi dengan kelas *ASSESSMENT\_FORM.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| *listOfAssessmentForm* | *List<Assessment\_Form>* | kumpulan *assessment form* |

Table 22. Detailed Class Assessment Form Manager

* ***COMPETENCY***

Kelas ini merepresentasikan kompentensi/uji kemampuan apa yang akan dilakukan pada *applicant.* Kelas ini hanya memiliki method getListCompetency()*.* Kelas ini hanya berelasi dengan kelas *ASSESSMENT\_FORM.* Spesifikasi atribut dalam kelas ini disajikan dalam bentuk tabel:

|  |  |  |
| --- | --- | --- |
| **Nama Atribut** | **Tipe Data** | **Keterangan** |
| Id\_kompetensi | *String* | *Primary key* dari kelas COMPETENCY yang otomatis diberikan oleh sistem. *not null*. |
| Nama\_kompetensi | *String* | *Not* *null.* Nama kompetensi dari suatu kompetensi |
| Penjelasan\_kompetensi | String | *Not null.* Penjelasan dari setiap kompetensi |

Table 23. Detailed Class Competency

# Behavioral Model

## Use Case Realization Specification

### 1.1 Use Case: View List of Available Position

|  |  |  |
| --- | --- | --- |
| ***Use Case Name:*** *View List of Available Position* | ***ID*:** R10 | ***Priority*:** High |
| ***Author*:** Khalila Hunafa Ansor Arifai | | |
| ***Primary Actor*:** *Recruiter* | | |
| ***Secondary Actor*:** N/A | | |
| ***Brief Description*:** *Use case* ini digunakan untuk melihat *list* *available* *position* yang telah dibuat. | | |
| ***Relationship*:**   * Included use case: N/A * Extention use case: View Available Position Detail Information * Base use case: N/A | | |
| ***Precondition*:**   1. *Recruiter* telah berada di halaman *home* | | |
| ***Normal Flow of Events*:**   |  |  | | --- | --- | | ***Recruiter*** | **Sistem** | | 1. Me-klik *button* *available* *position* |  | |  | 1. Me-*load* data dari *database* | |  | 1. Menampilkan *list* *available* *position* | | | |
| ***Subflows:*** N/A | | |
| ***Alternate Flows*:** N/A | | |

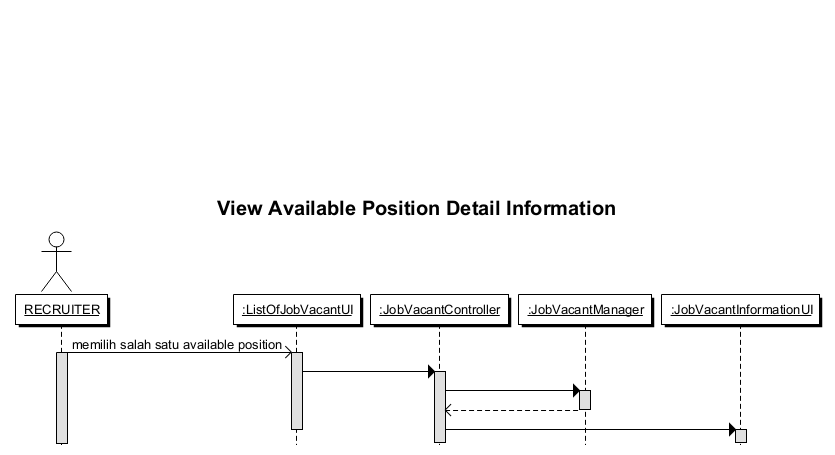
#### Sequence Diagram:

**usecase-khalila.png**

### 1.2 Use Case: View Available Position Detail Information

|  |  |  |
| --- | --- | --- |
| ***Use Case Name:*** *View available position detail information* | ***ID*:** R11 | ***Priority*:** Medium |
| ***Author*:** Khalila Hunafa Ansor Arifai | | |
| ***Primary Actor*:** *Recruiter* | | |
| ***Secondary Actor*:** N/A | | |
| ***Brief Description*:** *Use case* ini digunakan untuk melihat in*form*asi dari suatu *available* *position* | | |
| ***Relationship*:**   * Included use case: N/A * Extention use case: N/A * Base use case: *View* list of *available* *position* | | |
| ***Precondition*:**   1. *Recruiter* telah *login* kedalam sistem 2. *Recruiter* telah berada di halaman *list of available position* | | |
| ***Normal Flow of Events*:**   |  |  | | --- | --- | | ***Recruiter*** | **Sistem** | | 1. Memilih salah satu *available* *position* |  | |  | 1. Menampilkan halaman *available* *position* *detail* *information* | | | |
| ***Subflows*:** N/A | | |
| ***Alternate Flows*:** N/A | | |

#### Sequence Diagram:



# Database Design

## Database Schema

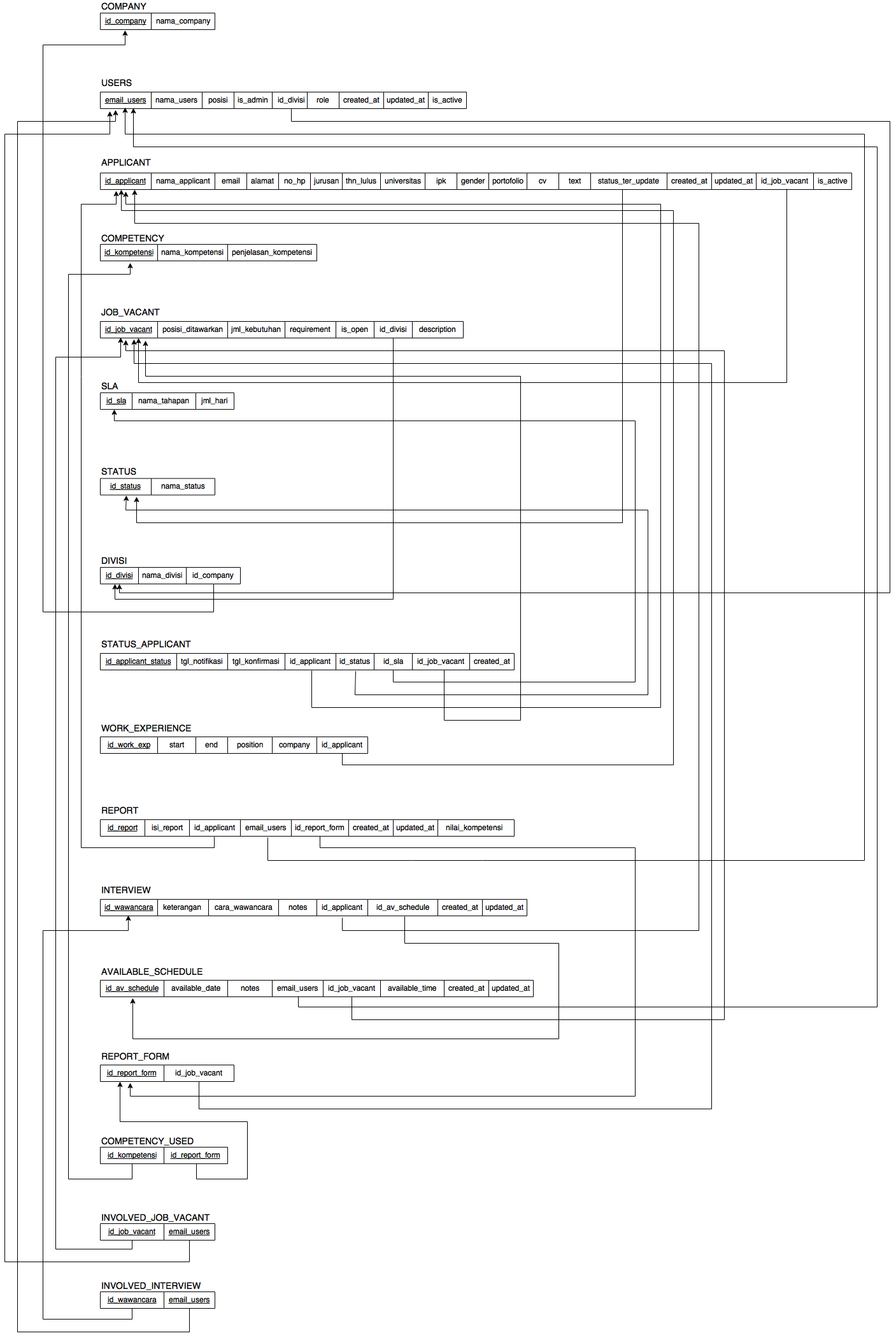


Figure 2. Skema Relational

## Table Structures and Attributes

|  |  |  |
| --- | --- | --- |
| **APPLICANT** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_applicant | Primary Key. Autoincrement | Integer |
| nama\_applicant | Not null | Varchar(70) |
| email | Not null | Varchar(70) |
| alamat | Not null | Text |
| no\_hp | Not null | Varchar(70) |
| jurusan | Not null | Varchar(70) |
| thn\_lulus | Not null | Smallint |
| universitas | Not null | Varchar(70) |
| ipk | Not null | Varchar(70) |
| gender | Not null | Varchar(2) |
| portofolio | Boleh null | Blob |
| cv | Not null | Blob |
| text | Boleh null | Text |
| status\_ter\_update | Not null. Foreign Key(STATUS).  On update/delete restrict | Integer |
| created\_at | Boleh null | Timestamp |
| updated\_at | Boleh null | Timestamp |
| id\_job\_vacant | Not null. Foreign Key(JOB\_VACANT).  On update/delete restrict | Integer |
| is\_active | Not null | Boolean |
| **USERS** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| email\_users | Primary Key | Varchar(70) |
| nama\_users | Not null | Varchar(70) |
| posisi | Not null | Varchar(70) |
| is\_admin | Not null | Boolean |
| id\_divisi | Not null. Foreign Key(DIVISI).  On update/delete restrict | Integer |
| role | Not null | Varchar(70) |
| created\_at | Boleh null | Timestamp |
| updated\_at | Boleh null | Timestamp |
| is\_active | Not null | Boolean |
| **DIVISI** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_divisi | Primary Key. Autoincrement | Integer |
| nama\_divisi | Alternate Key | Varchar(70) |
| id\_company | Not null. Foreign Key(COMPANY).  On update/delete restrict | Integer |
| **COMPANY** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_company | Primary Key. Autoincrement | Integer |
| nama\_company | Alternate Key | Varchar(70) |
| penjelasan\_kompetensi | Not null | Text |
| **JOB\_VACANT** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_job\_vacant | Primary Key. Autoincrement | Integer |
| posisi\_ditawarkan | Not null | Varchar(70) |
| jml\_kebutuhan | Not null | Smallint |
| requirement | Not null | Text |
| is\_open | Not null | Boolean |
| id\_divisi | Not null. Foreign Key(DIVISI).  On update/delete restrict | Integer |
| description | Not null | Text |
| **REPORT** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_report | Primary Key. Autoincrement | Integer |
| isi\_report | Boleh null | text |
| id\_applicant | Not null. Foreign Key(APPLICANT).  On update/delete restrict | Integer |
| email\_users | Not null. Foreign Key(USERS).  On update/delete cascade | Varchar(70) |
| id\_report\_form | Not null. Foreign Key(REPORT\_FORM).  On update/delete restrict | Integer |
| created\_at | Boleh null | Timestamp |
| updated\_at | Boleh null | Timestamp |
| nilai\_kompetensi | Boleh null | text |
| {id\_applicant, email\_users, id\_report\_form} Unik | | |
| **AVAILABLE\_SCHEDULE** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_av\_schedule | Primary Key. Autoincrement | Integer |
| available\_date | Not null | Date |
| notes | Boleh null | Text |
| email\_users | Not null. Foreign Key(USERS).  On update/delete cascade | Varchar(70) |
| id\_job\_vacant | Not null. Foreign Key(JOB\_VACANT).  On update/delete restrict | Integer |
| available\_time | Not null | Time |
| created\_at | Boleh null | Timestamp |
| updated\_at | Boleh null | Timestamp |
| **REPORT\_FORM** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_report\_form | Primary Key. Autoincrement | Integer |
| id\_job\_vacant | Alternate Key. Foreign Key(JOB\_VACANT).  On update/delete restrict | Integer |
| **COMPETENCY** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_kompetensi | Primary Key | Char(5) |
| nama\_kompetensi | Alternate Key | Varchar(70) |
| penjelasan\_kompetensi | Not null |  |
| **COMPETENCY\_USED** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_kompetensi | Not null. Foreign Key(COMPETENCY).  On update/delete restrict | Char(5) |
| id\_report\_form | Not null. Foreign Key(REPORT\_FORM).  On update/delete restrict | Integer |
| {id\_kompetensi, id\_report\_form} Primary Key | | |
| **STATUS\_APPLICANT** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_applicant\_status | Primary Key. Autoincrement | Integer |
| tgl\_notifikasi | Boleh null | Date |
| tgl\_konfirmasi | Boleh null | Date |
| id\_sla | Not null. Foreign Key(SLA).  On update/delete restrict | Integer |
| id\_status | Not null. Foreign Key(STATUS).  On update/delete restrict | Integer |
| id\_applicant | Not null. Foreign Key(APPLICANT).  On update/delete restrict | Integer |
| id\_job\_vacant | Not null. Foreign Key(JOB\_VACANT).  On update/delete restrict | Integer |
| created\_at | Boleh null | Timestamp |
| **WORK\_EXPERIENCE** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_work\_exp | Primary Key. Autoincrement | Integer |
| position | Boleh null | Varchar(70) |
| company | Boleh null | Varchar(70) |
| start | Boleh null | Date |
| end | Boleh null | Date |
| id\_applicant | Not null. Foreign Key(APPLICANT).  On update/delete restrict | Integer |
| **INTERVIEW** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_wawancara | Primary Key. Autoincrement | Integer |
| keterangan | Boleh null | Text |
| cara\_wawancara | Boleh null | Varchar(70) |
| notes | Boleh null | Text |
| id\_applicant | Not null. Foreign Key(INTERVIEW).  On update/delete restrict | Integer |
| id\_av\_schedule | Not null. Foreign Key(AVAILABLE\_SCHEDULE).  On update/delete restrict | Integer |
| created\_at | Boleh null | Timestamp |
| updated\_at | Boleh null | Timestamp |
| **SLA** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_sla | Primary Key. Autoincrement | Integer |
| nama\_tahapan | Not null | Varchar(70) |
| jml\_hari | Not null | Smallint |
| **STATUS** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_status | Primary Key. Autoincrement | Integer |
| nama\_status | Alternate Key | Varchar(70) |
| **INVOLVED\_JOB\_VACANT** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_job\_vacant | Not null. Foreign Key(JOB\_VACANT).  On update/delete restrict | Integer |
| email\_users | Not null. Foreign Key(USERS).  On update/delete cascade | Varchar(70) |
| {id\_job\_vacant, email\_users} Unik | | |
| **INVOLVED\_INTERVIEW** | | |
| **Atribut** | **Keterangan** | **Tipe data** |
| id\_wawancara | Not null. Foreign Key(INTERVIEW).  On update/delete restrict | Integer |
| email\_users | Not null. Foreign Key(USERS).  On update/delete cacade | Varchar(70) |
| {id\_wawancara, email\_users} Primary Key | | |

Table 24. Tipe data dan attribut

# User Interface

## Site Map

***Applicant***

D:\Kuliah\Semester 6\Propensi\SAD\Site Map\Applicant.png

Figure 3. Site Map Applicant

**Admin**



Figure 4. Site Map Admin

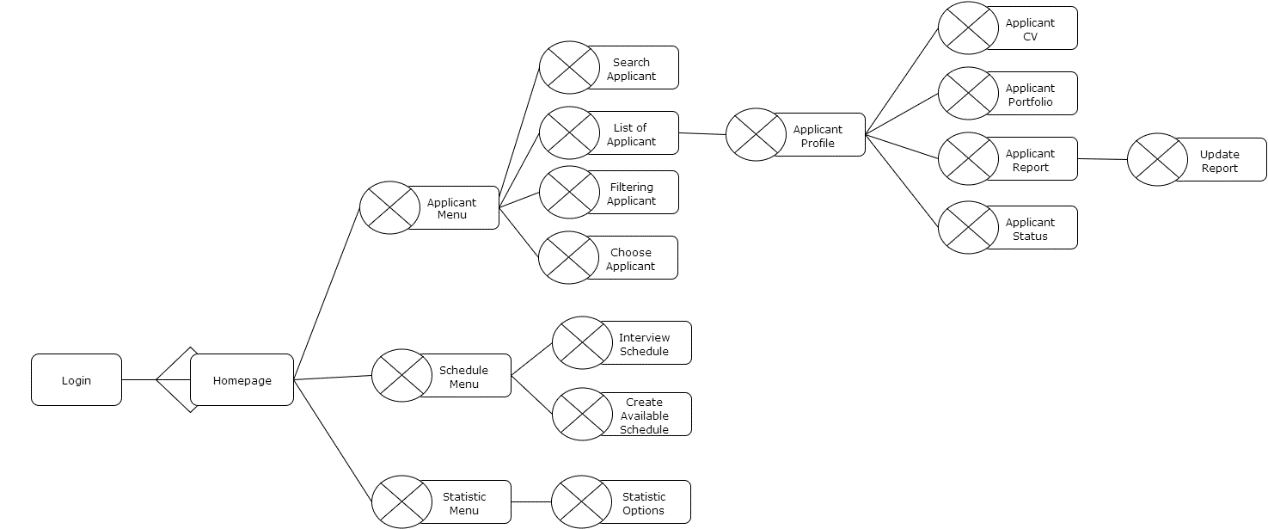
***Recruiter***

Figure 5. Site Map Recruiter

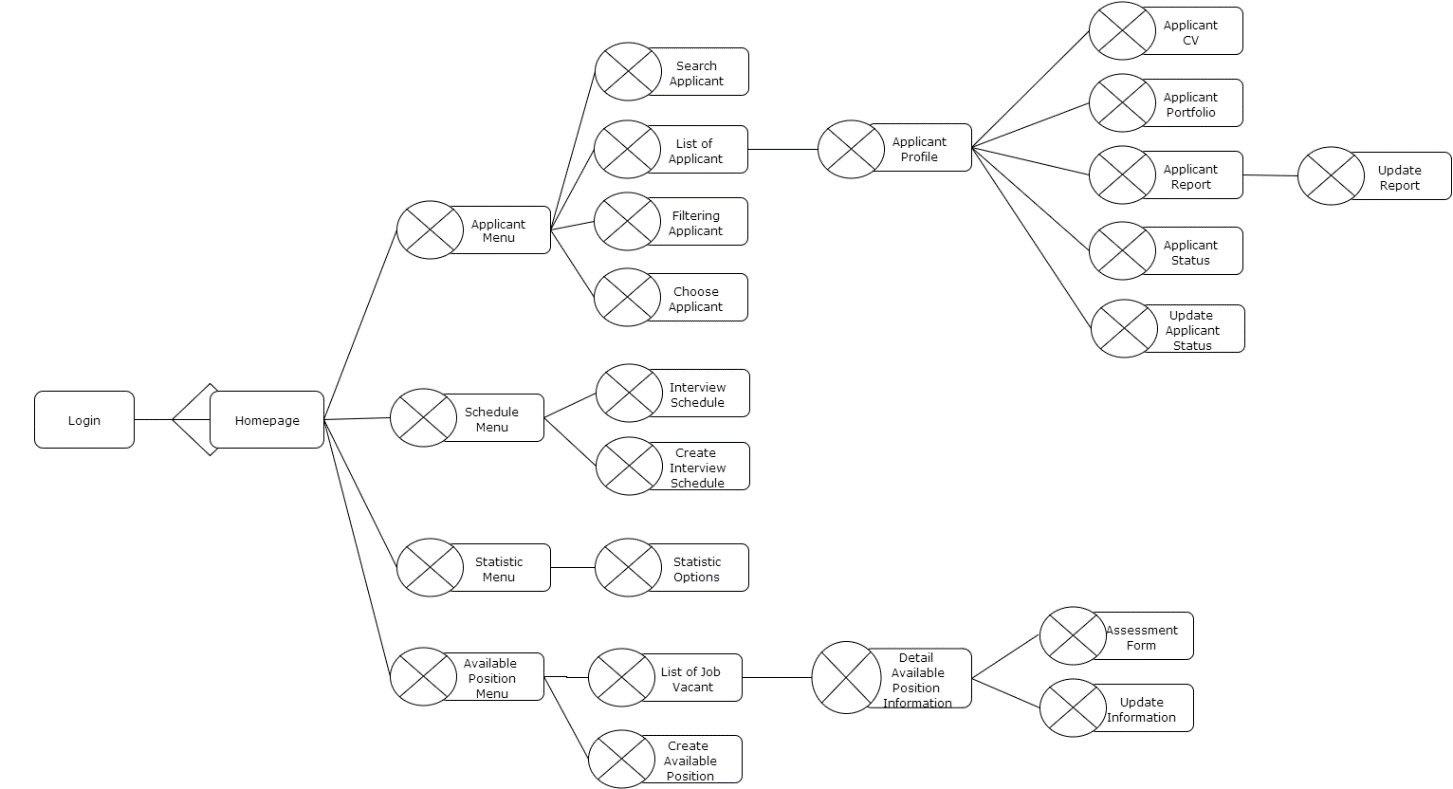
**HR**

Figure 6. Site Map HR

## User Interface Design

**Recruiter**

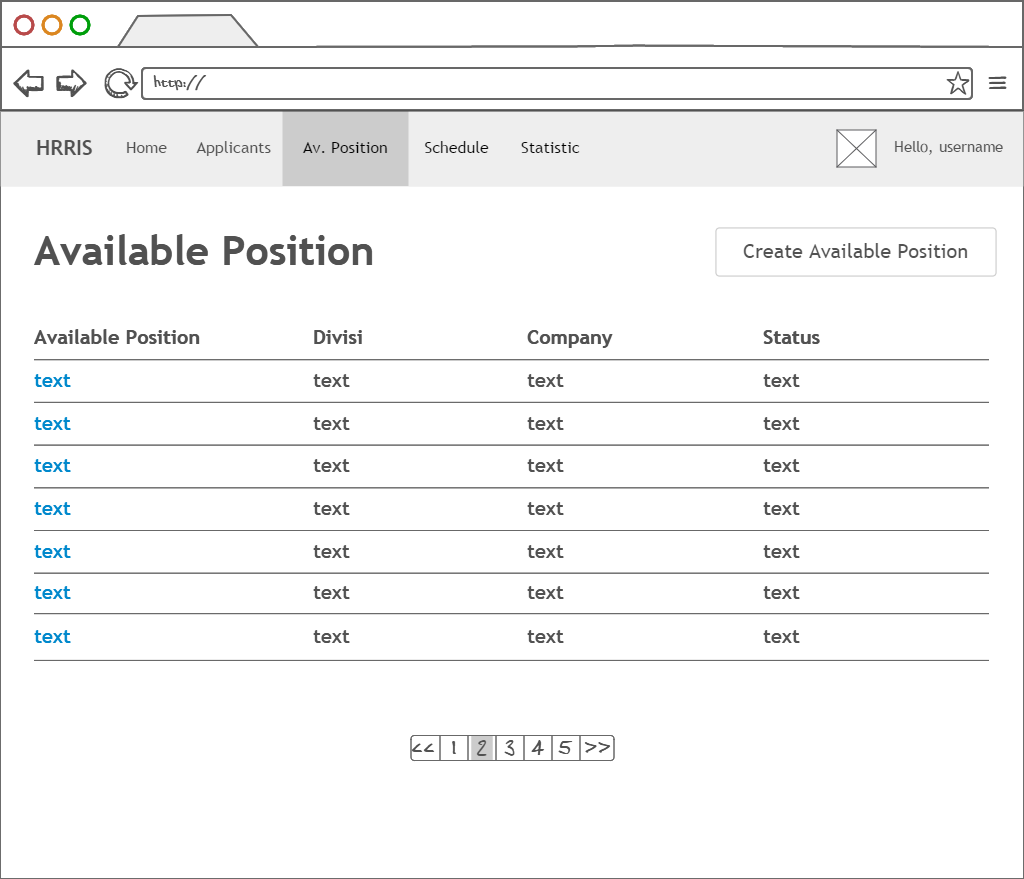


Figure 7. Wireframe List of Available Position

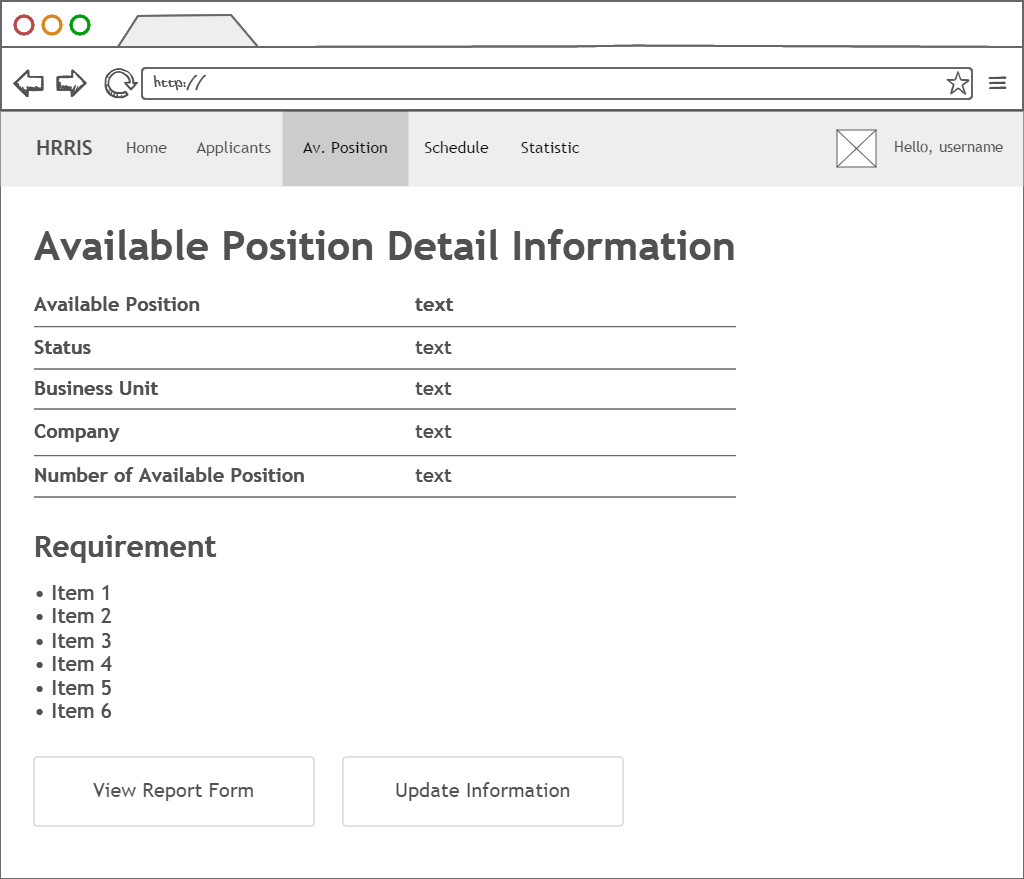
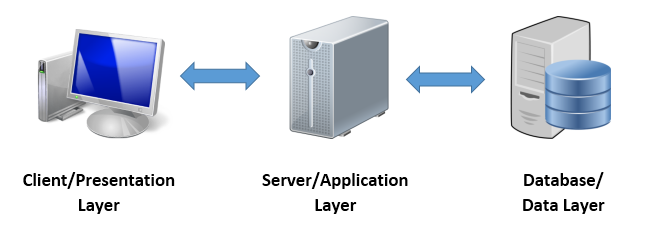


Figure 8. Wireframe Available Position Detail

# Architecture

## Physical Architecture Design



D:\Kuliah\Semester 6\Propensi\Revisi SAD 1\Architecture Component.png

Figure 9. Three Tier Client Structure

HRRIS akan menggunakan konsep desain three tier client-server architecture dikarenakan HRRIS menggunakan *protocol* HTTP. Berikut adalah penjelasan dari arsitektur yang akan digunakan beserta komponen yang ada :

* ***Presentation Layer*** :

Sebagai *top level* *layer* dan tampilan informasi dari *service* yang ada didalam aplikasi. *Layer* ini akan digunakan oleh user yang mengakses *web application* baik menggunakan *personal computer*.

* ***Application Layer*** :

Pengontrol fungsionalitas dari aplikasi dengan menjalankan *detailed processing*. *Layer* ini merupakan proses lanjutan dari *layer* sebelumnya, yaitu *presentation layer*. Layer ini berisi *application server* yang digunakan untuk melakukan komunikasi antara *layer* sebelum maupun sesudahnya dan *business logic* yang digunakan untuk melakukan *business service* yang disediakan aplikasi.

* ***Database Layer*** :

*Layer* terakhir dalam arsitektur ini. *Layer* ini digunakan untuk menyimpan dan mengolah data sesuai dengan permintaan layer sebelumnya. Komponen *data* *access logic* digunakan untuk mengakses data yang dibutuhkan dan *database server* digunakan untuk *create*, *read*, *update*, dan *delete*.

## Hardware and Software Specification

|  |  |
| --- | --- |
| ***System Resources*** | |
| ***Sofware*** | |
| ***Resource*** | ***Description*** |
| Laravel | *Framework* yang digunakan dalam *development.* |
| MySQL | *Database* yang digunakan dalam *development.* |
| *Internet Browser* | Penggunaan dalam *testing* dan pencarian referensi.  Browser yang digunakan minimal mampu menjalankan HTML5, JavaScript, CSS3, WOFF (Web Open Font Format). |
| Microsoft Office | Digunakan sebagi *text editor* dan *presentation medium* |
| Adobe Photoshop | Penggunaan dalam desain-desain yang dibutuhkan. |
| *Source Code Editor* | Penggunaan dalam meng-*edit source code.* |
| XAMPP | Penggunaan dalam masa *development* untuk melakukan *testing* dalam *localhost*. |
| Google Drive | Penggunaan dalam manajemen *file* dan *collaborative writing*. |
| ***Hardware*** | |
| ***Resource*** | ***Description*** |
| *Server* | Digunakan dalam tahap *deployment* terakhir. Server yang digunakan merupakan *staging server* dengan DigitalOcean*.*  Spesifikasi :   * RAM : 2GB * *Disk Space* : 40 GB SSD * No. CPU : 2 * OS : Ubuntu 14.04 LTS |
| *PC /* Laptop | Digunakan untuk *development* dan *testing.*  Min. *requirement* :   * Windows XP 32 bit * RAM 512MB * Intel Pentium 4 * Ethernet |

Table 26. Hardware and Software Specification

# Test Plan

Untuk memastikan sistem yang dikembangkan berjalan dengan baik, akan dilakukan beberapa testing pada sistem. Adapun *testing* yang akan dilakukan adalah *Unit Testing* dan *User Acceptance Test (UAT)*.

## Unit Testing

*Unit Testing* adalah pengujian yang berfokus pada unit terkecil yang terdapat pada modul. *Unit testing* yang dilakukan pada sistem ini hanya akan dilakukan pada satu modul yaitu modul *recruitment*. Setiap fungsi yang terdapat pada modul *recruitment* akan diuji menggunakan *test cases* sesuai dengan parameter yang digunakan dan melihat apakah *output* yang diinginkan sesuai dengan *input* yang dimasukkan.

## User Acceptance test (UAT)

Dikenal dengan *beta testing* adalah *application testing* yang dilakukan ketika proses implementasi program sudah selesai dan akan dilakukan oleh *client.* UAT dilakukan untuk memastikan bahwa fungsi-fungsi yang dikembangkan berjalan sesuai dengan *requirement* yang sudah ditentukan sekaligus untuk menguji apakah *user interface* yang dikembangkan sudah baik atau belum.

Pada proses *development,* pengujian sistem akan dilakukan menggunakan bantuan Laravel Debugbar. Laravel Debugbar adalah sebuah *package* yang mengintegrasikan Laravel 5 dengan PHP Debugbar.  Semua *testing* tersebut akan dilakukan oleh *tester* yang dimiliki tim pengembang kecuali pada saat UAT, dimana dilakukan oleh *client*. Dalam proses melakukan *testing*, *tester* memerlukan *tool* yang akan digunakan yaitu laptop dan *browser* Google Chrome atau Firefox versi terbaru. Semua *testing* tersebut akan dilakukan oleh *tester* yang dimiliki tim pengembang. Dalam proses melakukan *testing*, *tester* memerlukan *tool* yang akan digunakan yaitu laptop dan *browser* Google Chrome atau Firefox versi terbaru.

# References

Tim Dosen Basis Data. (2015). *Pemetaan ER-EER ke Model Relational.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen Basis Data. (2015). *Pemodelan Basis Data dengan EER.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen Basis Data. (2015). *Pemodelan Basis Data dengan ER.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen RPL. (2015). *Architecture Design.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen RPL. (2015). *Design Concept.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen RPL. (2015). *Interaction Diagram.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen RPL. (2015). *Software Testing Strategies.* Depok: Fakultas Ilmu Komputer UI.

Tim Dosen RPL. (2015). *Use Case Modelling.* Depok: Fakultas Ilmu Komputer UI.