

BAB III PELAKSANAAN KERJA PROFESI

3.1 Bidang Kerja

Dalam melakukan kegiatan Merdeka Belajar Kampus Merdeka (MBKM) di PT. Matahari Putra Prima Tbk, praktikan diberikan posisi di departemen Human Resource (HR) khususnya pada bagian *Human Resource Information System* (HRIS). HRIS memiliki berbagai tugas penting terhadap perancangan sistem operasional departemen HR, yaitu menyediakan berbagai macam aplikasi berbasis *website*.

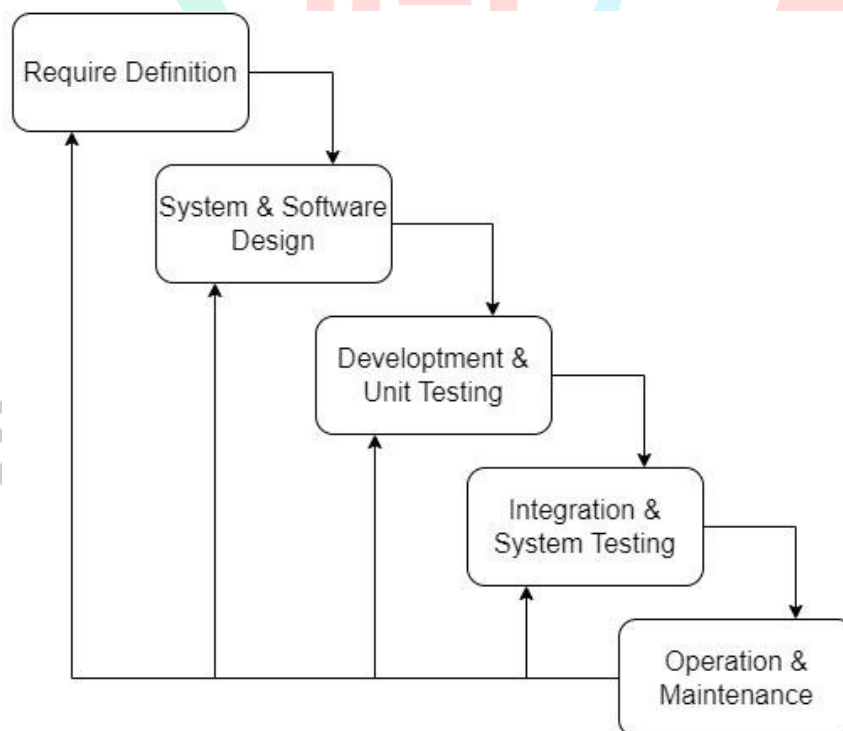
Sebagai seorang HRIS, praktikan diharapkan mampu memiliki keahlian di bidang *Front-end* dan *Back-end*. Praktikan diarahkan oleh pembimbing kerja untuk mengembangkan, memelihara, dan mengelola seluruh tugas yang berkaitan dengan teknologi. Dengan adanya penerapan HRIS menjadi lebih mudah untuk melakukan pengumpulan informasi dan melakukan analisis dengan tujuan berkontribusi pada implementasi strategi bisnis (Safaâ & Mohamed, 2020). Kemudian, praktikan juga melakukan diskusi secara berskala dengan Departemen *Human Resource* khususnya bidang *Talent Acquisition* untuk melakukan pembahasan lebih lanjut apa saja yang dibutuhkan di dalam sebuah aplikasi HR. Praktikan juga diberi tanggung jawab untuk membuat dan mengelola aplikasi berbasis *website* antara lain, *MPP-Recruitment*, Portal HR, dan *Logbook* Ritra, dengan adanya proses integrasi data yang telah dibentuk di dalam *database* dan ditampilkan ke sebuah wadah atau komponen yang berada pada *website*.

Aplikasi *MPP-Recruitment* yang dikembangkan oleh praktikan, merupakan sebuah aplikasi menggunakan bahasa pemrograman *PHP Native* versi 8 berbasis *website*. Aplikasi ini digunakan untuk memberi fasilitas kepada masyarakat agar dapat melamar kerja secara efisien. Pada aplikasi ini, praktikan ditugaskan untuk membangun relasi antara variabel pada data, konfigurasi XAMPP, dan konfigurasi *database*. Oleh karena itu, proyek ini sangat sesuai dengan program studi yang dijalani oleh praktikan karena sejalan dengan kurikulum yang sudah dipelajari selama masa studi.

Aplikasi *Logbook* Ritra dan Portal HR merupakan aplikasi berbasis *website* yang dibuat serta dikembangkan oleh praktikan dengan menggunakan *Framework Laravel 10*. Oleh karena itu, praktikan memperoleh pengetahuan baru, yaitu proses instalasi *Laravel 10* dan *Tailwind CSS*, struktur folder *Laravel 10*, sistem *route* pada *Laravel 10*, pemecahan masalah, dan beberapa *library* baru yang sebelumnya tidak pernah dipelajari selama masa studi.

3.2 Pelaksanaan Kerja

Pelaksanaan kerja yang dilakukan oleh praktikan mencakup berbagai macam fitur yang dirancang melalui beberapa tahap untuk mendapatkan hasil yang baik dari sisi pengguna maupun sisi pengembang. Tahapan utama, dimulai dengan melakukan analisa yang meliputi, kebutuhan *user*, kenyamanan *user*, dan fleksibilitas *user*, kemudian melakukan pemodelan sebagai perancangan aplikasi yang terdiri dari, struktur aplikasi, *Unified Modelling Language (UML)*, *sequence diagram*, dan *activity diagram*. Setelah tahap tersebut berhasil dibentuk, kemudian dilakukannya proses implementasi pembuatan aplikasi baik dari segi operasi *Front-end* dan *Back-end*.

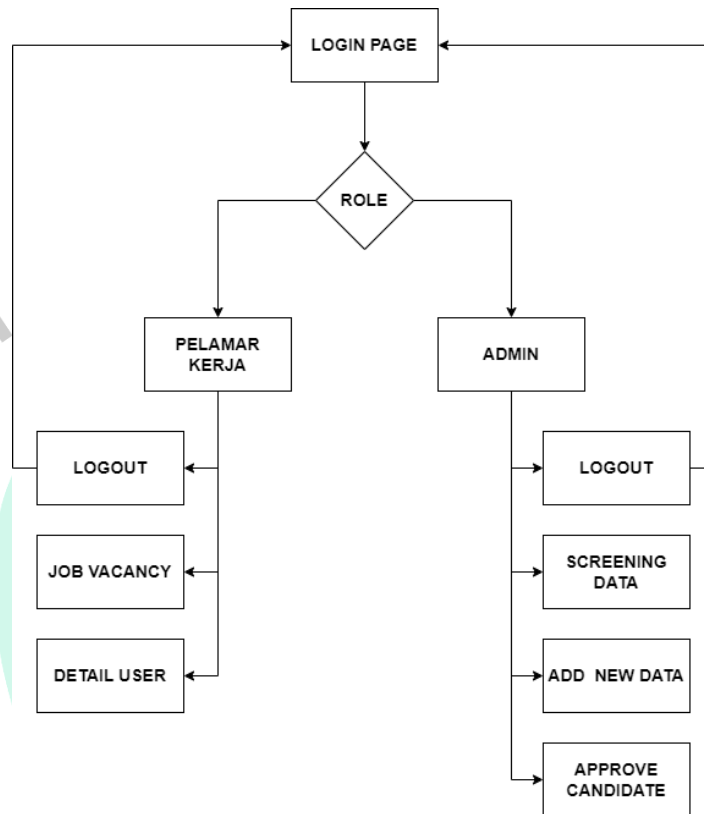


Gambar 3.1 Tahapan Perancangan Aplikasi

Gambar 3.1 merupakan tahapan perancangan yang dilakukan oleh praktikan dalam pembuatan aplikasi pada kegiatan MBKM di PT. MPPA Tbk dengan memiliki alur kerja secara sistematis. Selama membangun aplikasi tersebut, praktikan juga mendapatkan pengawasan serta pengarahan oleh pembimbing kerja supaya perancangan aplikasi tersebut dapat terealisasi secara sistematis dan maksimal.

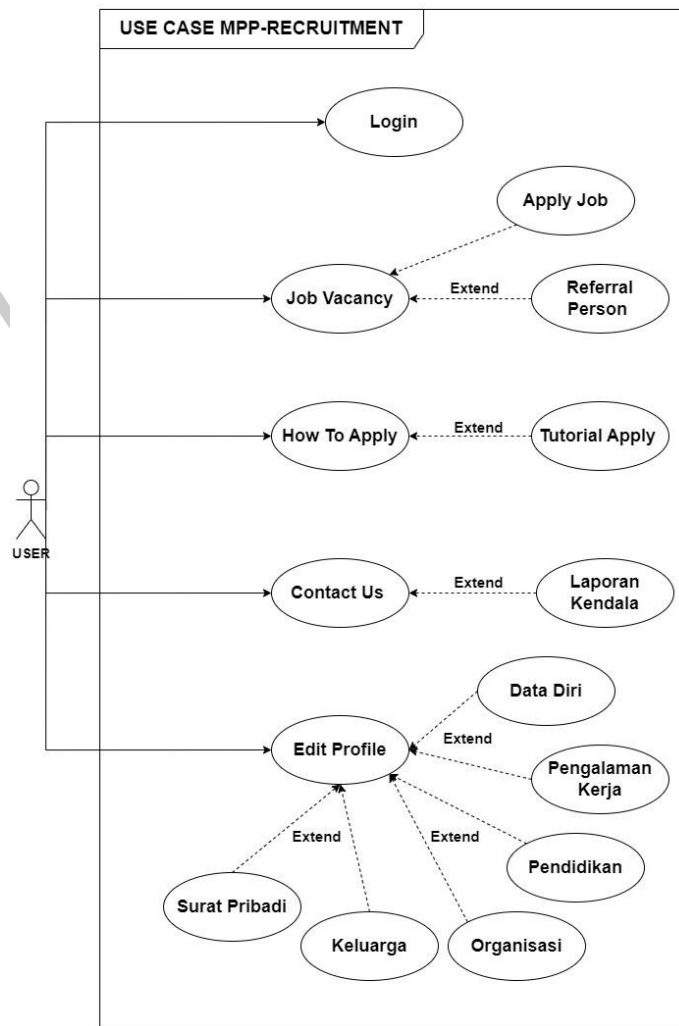
3.2.1. Perancangan

Pada tahap perancangan aplikasi *MPP-Recruitment* berbasis *website*, praktikan diberikan arahan oleh pembimbing kerja untuk melakukan pengembangan aplikasi berdasarkan yang sudah ada sebelumnya sehingga dapat beroperasi kembali. Aplikasi tersebut dibuat khusus calon pelamar kerja agar lebih mudah mendapatkan keseluruhan informasi mengenai *job vacancy*, *job description*, *job scope*, dan *required*. Dengan dikembangkannya aplikasi ini departemen HR juga lebih mudah dalam melakukan seleksi karyawan karena tidak perlu menggunakan aplikasi ketiga dalam mencari informasi lengkap mengenai calon pelamar. Pembimbing eksternal memberikan alur serta contoh aplikasi terdahulu dan kemudian praktikan kembangkan menjadi sebuah aplikasi yang lebih baru. Struktur tersebut mencakup seluruh informasi mengenai aplikasi *MPP-Recruitment* agar dapat mudah dipahami oleh pengembang selanjutnya.



Gambar 3.2 Struktur Aplikasi

Gambar 3.2 merupakan rancangan struktur aplikasi *MPP-Recruitment*. Rancangan sistem ini merupakan fase perancangan sistem yang melibatkan pembuatan gambaran atau representasi proses-proses elemen-elemen dari sebuah komponen yang akan dibentuk (Pertiwi & Taufiq, 2020). Pada saat melakukan perancangan ini, praktikan perlu melakukan analisis persyaratan untuk mengetahui fungsi dan fitur apa yang dibutuhkan pada sistem, selama tahap perancangan pembimbing kerja dari perusahaan memandu dan memberikan masukan dan saran kepada praktikan dalam menyiapkan *website* yang baik serta memenuhi kebutuhan *user*.

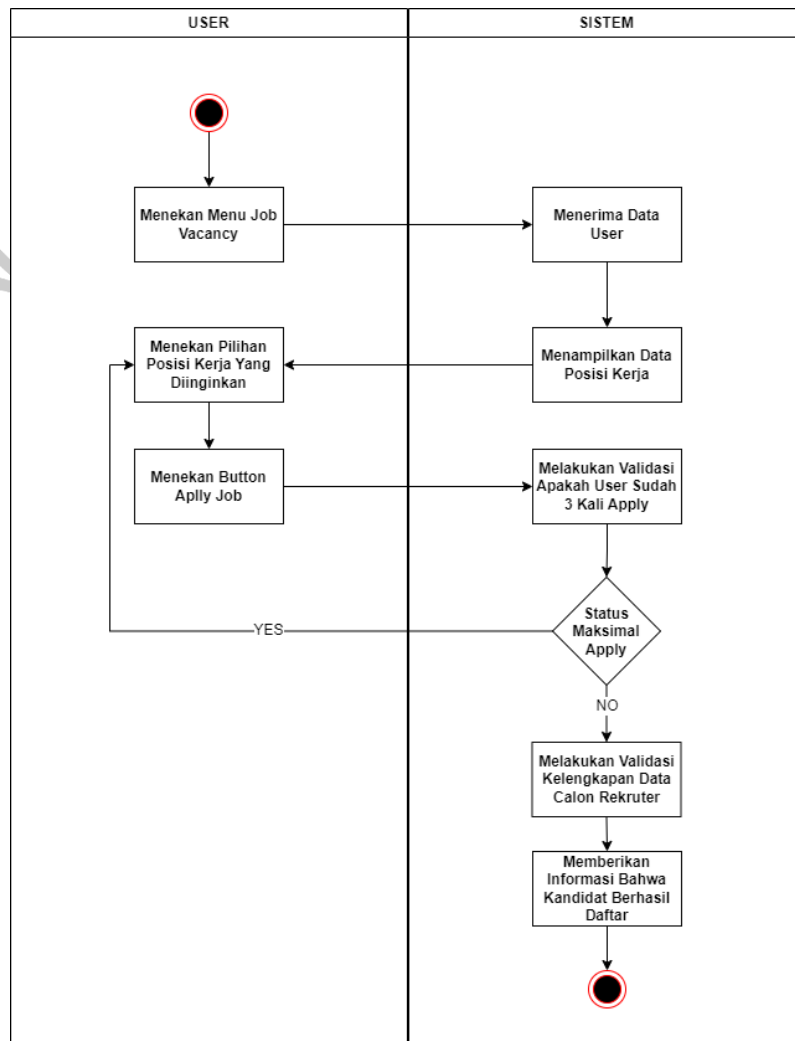


Gambar 3.3 Use Case Diagram

Gambar 3.3 adalah Use Case Diagram dari sisi pelamar kerja pada aplikasi *MPP-Recruitment*. Aplikasi ini memiliki empat use case utama, yaitu *job vacancy*, *how to apply*, *contact us*, dan *edit profile*. Use Case Diagram merupakan diagram yang menunjukkan interaksi antara aktor dengan sistem. Diagram ini dapat menggambarkan sebuah hubungan antara satu atau lebih aktor dengan sistem yang akan dibuat hubungan antara satu atau lebih aktor dengan sistem yang dibuat. (Julianto & Setiawan, 2019).

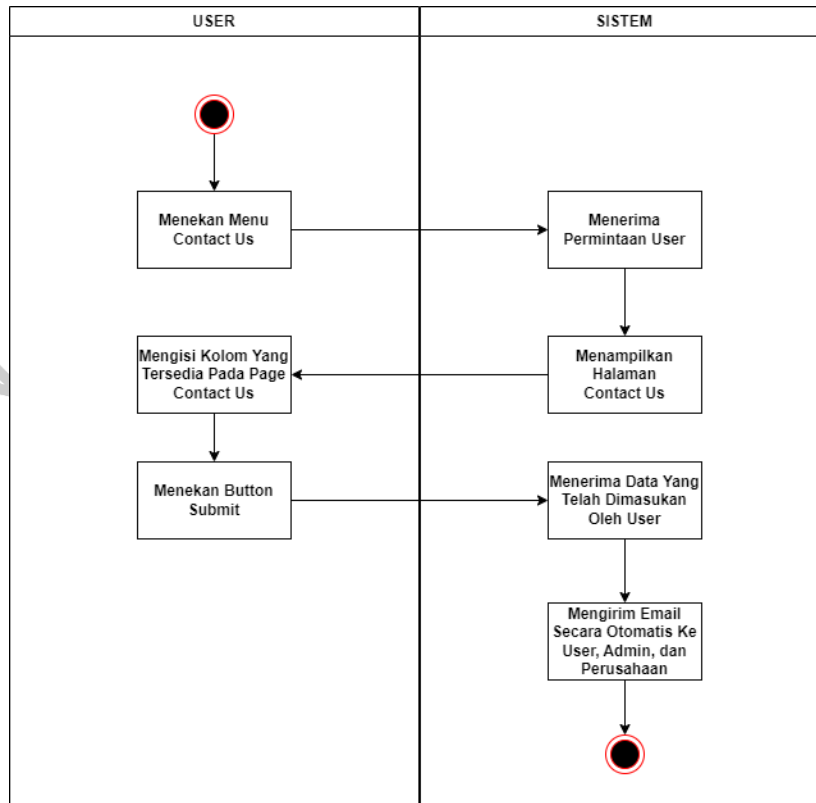
Setelah membuat *use case diagram*, praktikan melanjutkan proses perancangan *activity diagram* untuk menggambarkan alur kerja setiap fitur yang terdapat pada aplikasi *MPP-Recruitment*. *Activity Diagram* adalah diagram yang

menggambarkan alur kerja secara urutan dari proses awal sampai akhir sebuah sistem.



Gambar 3.4 Rancangan Activity Diagram Proses Lamar Kerja

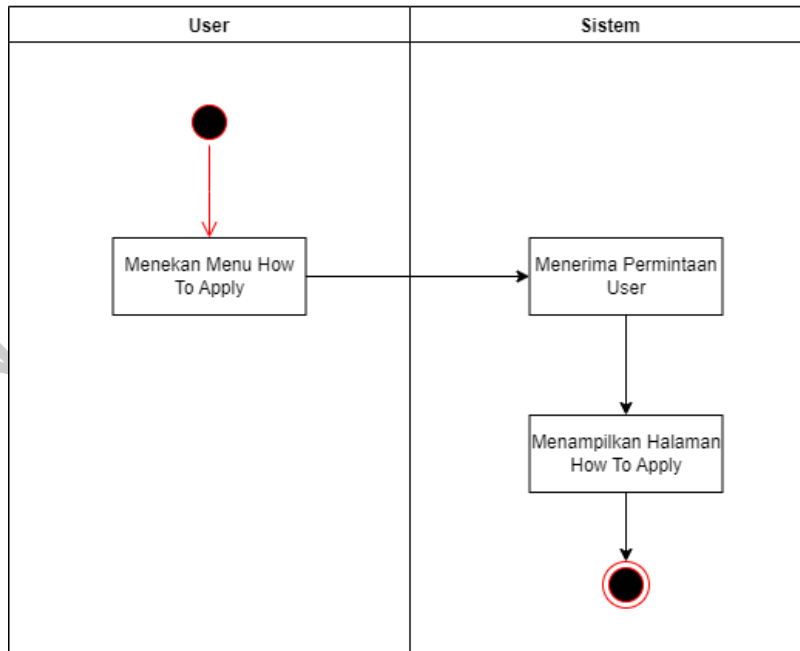
Gambar 3.4 merupakan alur kerja yang terjadi ketika kandidat akan melakukan aksi lamar pekerjaan. Proses ini dapat berjalan ketika seluruh data pribadi kandidat sudah terpenuhi, apabila kandidat berhasil melamar pekerjaan maka secara otomatis kandidat dapat melihat tahap tahapan yang sudah dilakukan oleh tim HR.



Gambar 3.5 Rancangan *Diagram Activity* Fitur *Contact Us*

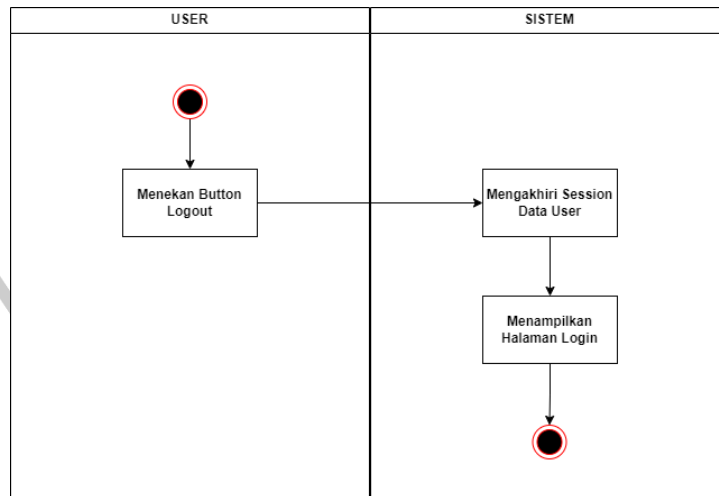
Gambar 3.5 merupakan alur kerja yang terjadi ketika *user* melakukan laporan kendala pada halaman *contact us*. Ketika *user* sudah melengkapi kolom yang diperlukan untuk melakukan proses laporan, maka secara otomatis juga sistem mengirimkan *email* kepada pihak perusahaan, admin, dan *user* menggunakan *library Simple Mail Transfer Protocol (SMTP)*. Kemudian data ini secara otomatis akan tersimpan ke dalam *database*.

Activity Diagram How To Apply



Gambar 3.6 Rancangan *Diagram Activity How To Apply*

Gambar 3.6 merupakan alur kerja yang terdapat pada halaman *how to apply*. Pada halaman ini hanya berisikan tahapan-tahapan yang telah disediakan oleh admin untuk menjelaskan proses lamar kerja pada aplikasi *MPP-Recruitment berbasis website*.

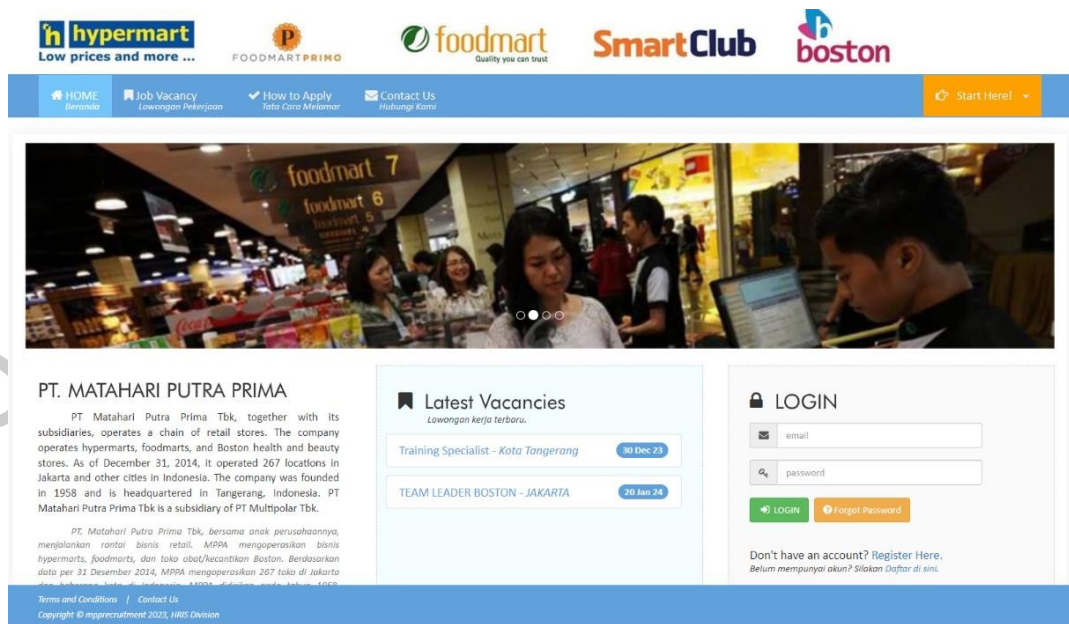


Gambar 3.7 Rancangan Activity Diagram Fitur Logout

Gambar 3.7 merupakan alur kerja yang terjadi ketika *user* melakukan proses *logout*. Sistem secara otomatis akan mengakhiri seluruh akses yang tersimpan di dalam *session user* dan menghapus *session* yang telah dibentuk sebelumnya di aplikasi. Kemudian *user* akan dikembalikan ke halaman *login*.

Setelah seluruh *activity diagram* berhasil dirancang. Kemudian praktikan mendapatkan arahan untuk melakukan implementasi sesuai dengan rancangan *use case* yang telah dibentuk oleh praktikan sebelumnya. Dalam *website* ini terbagi menjadi 2 *role* yaitu admin dan kandidat, dimana masing-masing *role* memiliki hak akses yang berbeda. Jika *login* sebagai kandidat akan diarahkan terlebih dahulu ke *dashboard user*, yang mana kandidat memiliki akses untuk melakukan *apply job*, *referall person*, dan melihat daftar pekerjaan. Namun jika *login* sebagai admin maka akan diarahkan terlebih dahulu ke *dashboard admin*, yang mana memiliki hak akses untuk *update* status kandidat pelamar, *create* kandidat, *create job*, *screening* kandidat, dan *update* banner. Oleh karena itu praktikan melakukan perancangan terlebih dahulu dalam mengembangkan aplikasi *MPP-Recruitment*, perancangan tersebut melalui banyak tahap antara lain, rancangan UML, rancangan *mockup* desain, pembuatan *database* berserta relasi *database*. Jika sudah, praktikan melakukan tahap selanjutnya yaitu *development*. *Development* adalah proses merealisasikan hasil dari rancangan yang telah dibangun menjadi sebuah aplikasi berbasis *website* yang digunakan

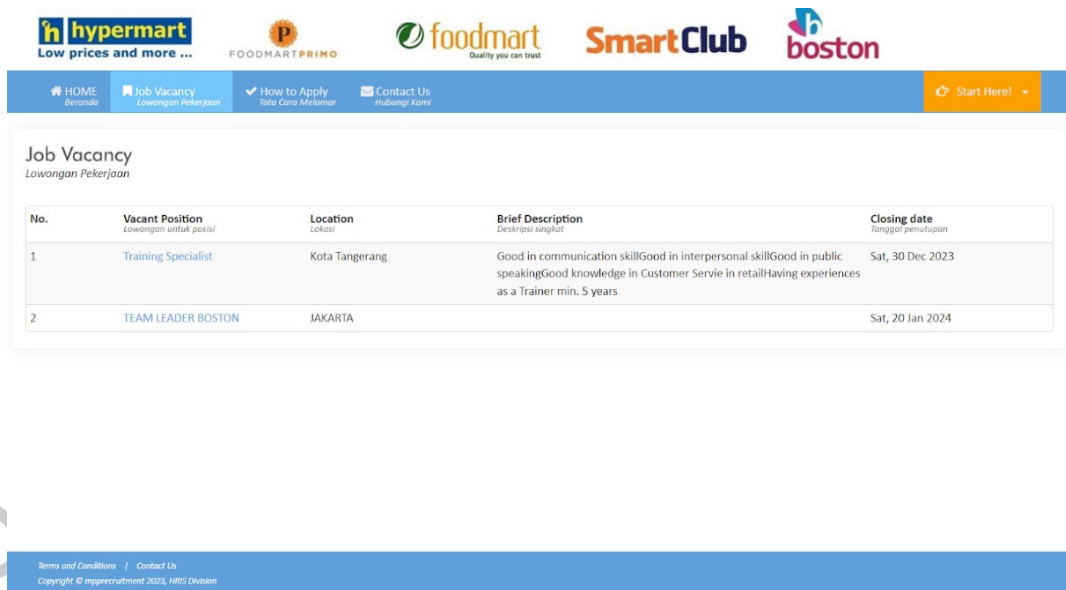
sesuai dengan fungsinya. Hasil dari *development* dapat dilihat pada gambar-gambar yang tertera dibawah ini.



Gambar 3. 8 Dashboard Sebelum Login

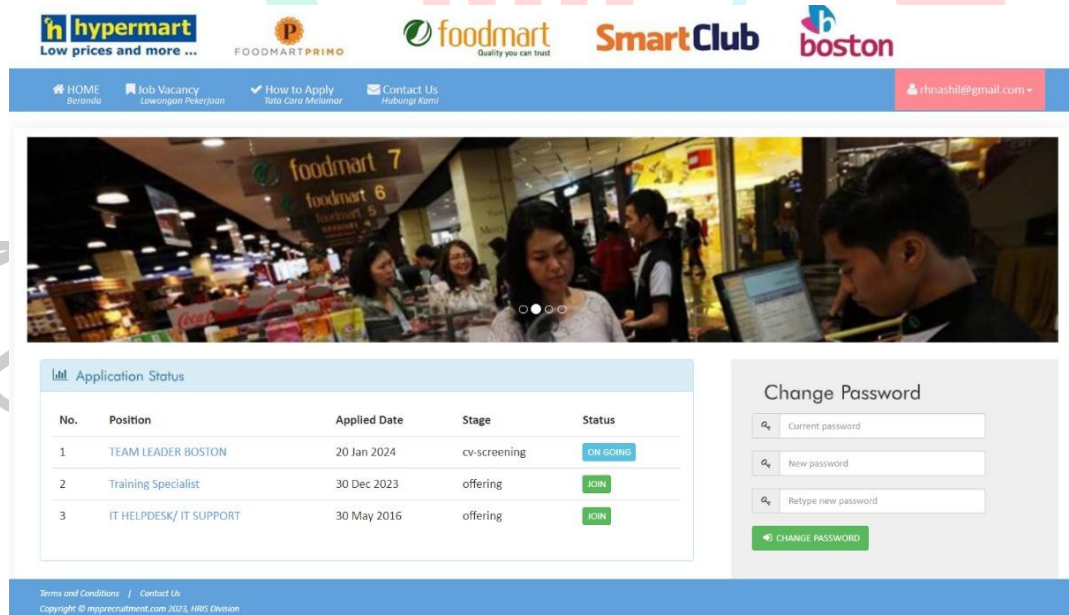
Gambar diatas merupakan hasil implementasi yang telah dikembangkan oleh praktikan sebagai tampilan utama ketika *user* akses *website*. Dalam *dashboard* ini terdapat informasi singkat mengenai perusahaan, daftar lowongan pekerjaan terbaru, dan *form login*. Dengan adanya halaman ini, dapat mempermudah pengguna untuk mendapatkan informasi singkat mengenai perusahaan dan daftar pekerjaan terkini.

Ketika *user* memilih menu “*Job Vacancy*”, akan beralih ke halaman yang berbeda dengan menampilkan daftar lowongan pekerjaan. Ketika *user* ingin melihat informasi lengkap mengenai posisi pekerjaan yang sedang dibutuhkan perusahaan, maka interaksi dapat dilakukan dengan menekan nama posisi tersebut.



Gambar 3.9 Job Vacancy

Kemudian apabila *user* sudah berhasil melaksanakan *login*, maka akan berpindah ke halaman *user* dengan tampilan yang berbeda. Seperti gambar dibawah ini.



Gambar 3.10 Tampilan Halaman User

Gambar 3.10 merupakan tampilan yang diberikan kepada pengguna setelah berhasil menyelesaikan proses *login*. Tampilan tersebut menampilkan informasi mengenai status aplikasi, yang mencakup posisi lowongan pekerjaan yang telah dilamar, tanggal lamaran, tahap berdasarkan *review* HRD, dan status.

Pengguna dapat melakukan interaksi dengan menekan posisi, yang akan mengarahkannya ke detail lowongan pekerjaan.

Tampilan “Detail Job Vacancy” ketika *user* sudah berhasil menyelesaikan tahap proses *apply* maka status akan berubah menjadi “*Already Applied*”. Kemudian terdapat maksimal 3 kali kesempatan dalam melakukan *apply* pekerjaan.

Detail Job Vacancy
Detil Lowongan Pekerjaan

The screenshot shows a job vacancy detail page for a 'Training Specialist' position. The page is titled 'Training Specialist' and includes the following information:

- Work Location (Lokasi Pekerjaan):** Kota Tangerang
- Job Description (Deskripsi Pekerjaan):**
 - Candidate must possess at least a Bachelor's Degree, Education/Teaching/Training, Human Resource Management, Others or equivalent.
 - Required skill(s): communication skill, coaching skill, public speaking, Customer Services.
 - Required language(s): English, Bahasa Indonesia
 - At least 5 year(s) of working experience in the related field is required for this position.
 - Applicants must be willing to work in BANTEN.
 - Preferably Supervisor / Coordinators specializing in Sales - Retail/General or equivalent.
 - Full-Time position(s) available.
- Minimum Degree (Pendidikan minimal):** Bachelor - S1
- Closing Date (Penutupan lowongan):** Please make sure to apply for this position latest on **Saturday, 30 Dec 2023**
Mohon kiriman lamaran Anda selambat-lambatnya tanggal 30 Dec 2023

At the bottom of the page, there are three buttons:

- You have already applied for this position** (orange button, *Anda sudah melamar posisi ini*)
- Return to List of Job Vacancy** (blue button, *Kembali ke list Lowongan*)
- Referall Person to This Position** (green button, *Referall Calon Kandidat*)

Gambar 3.11 Tampilan Detail Job Vacancy

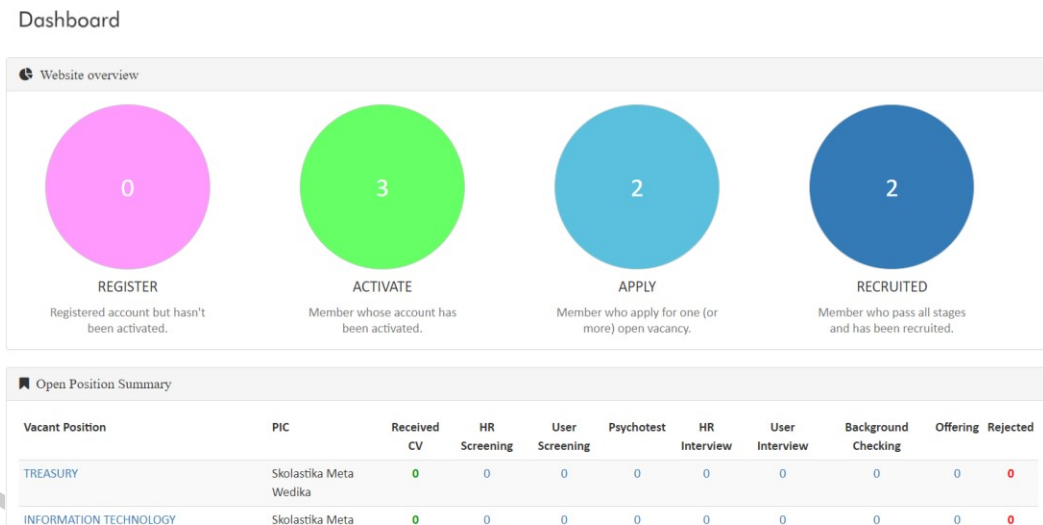
Selain *apply job*, *user* dapat melakukan interaksi dengan menekan tombol “*Referall Person*” yang digunakan untuk memberikan rekomendasi seseorang, dimana seseorang tersebut memiliki kompetensi yang sesuai dengan posisi pekerjaan tersebut. Seperti gambar di bawah ini terdapat *form referall person* yang membutuhkan nama, *email*, dan nomor *handphone* yang akan direkomendasikan.

The image shows a web form titled "Referral Person" with a close button (x) in the top right corner. The form contains the following fields:

- Nama:** Raihan Ashil Zaki
- Email:** rhnashil@gmail.com
- Posisi:** Training Specialist
- Nama Calon Rekomendasi:** (empty text input)
- Email Calon Rekomendasi:** (empty text input)
- No. Handphone Calon Rekomendasi:** (empty text input)
- Submit Referral:** A green button with a right-pointing arrow icon.

Gambar 3.12 Tampilan *Referall Person*

Selanjutnya apabila *login* sebagai role admin, hak akses yang dimiliki oleh *role admin* sangat banyak terutama dapat melakukan proses CRUD data, filterisasi data, *update status* dan *stage applicant job*. Seluruh data yang ditampilkan pada *interface user* dikelola secara langsung oleh *admin*.



Gambar 3.13 Tampilan *Dashboard Admin*

Gambar 3.13 merupakan tampilan *dashboard admin* yang digunakan untuk menampilkan data yang telah dikelola dan dikelompokkan per-klasifikasi. Selain itu, terdapat daftar lowongan pekerjaan beserta jumlah calon kandidat yang telah melamar pada posisi tersebut. Admin memiliki hak akses untuk melakukan pembaruan *stage* pada *user*.

Terdapat beberapa tahapan proses yang harus dijalani oleh *user*, antara lain, *HR Screening*, *User Screening*, *Psychotest*, *HR Interview*, *User Interview*, *Background Checking*, *Offering*, *Received*. Hasil dari proses ini akan ditampilkan pada halaman indeks pengguna sebagai informasi bahwa mereka telah mencapai tahap yang ditentukan oleh *admin*. *Admin* dapat memberikan status *Ongoing*, *Pass*, *Pending*, dan *Reject* seperti yang terlihat pada gambar di bawah ini.

Processing Status for Testing

Position : BUDGETING

PIC : Skolastika Meta Wedika

Stage : HR Screening Completion Date : yyyy-mm-dd hh:ii

Processing Status : ON GOING Scheduled Date for next step : yyyy-mm-dd hh:ii

Add notes to history :

PROCESS

Close

Gambar 3.14 Tampilan *Processing Status*

Kemudian *admin* dapat melakukan klasifikasi data berdasarkan kategori yang dibutuhkan antara lain, *candidate name*, *education degree*, *education major*, *job position*, *religion*, *candidate email*, *line of business*, *graduation year*, *current city*, dan *candidate sex*. Interaksi yang dapat dilakukan oleh admin pada halaman ini dapat memilih data berdasarkan kriteria yang dibutuhkan, fitur ini digunakan untuk mempermudah admin dalam mengelola data.

Candidate Screening

Choose criteria(s) for screening

Candidate's name

Education degree

Education major

Job Position

Religion

Candidate's email

Line of Business

Graduation year

Current city

Candidate's sex

SEARCH

Gambar 3.15 Tampilan *Screening Data*

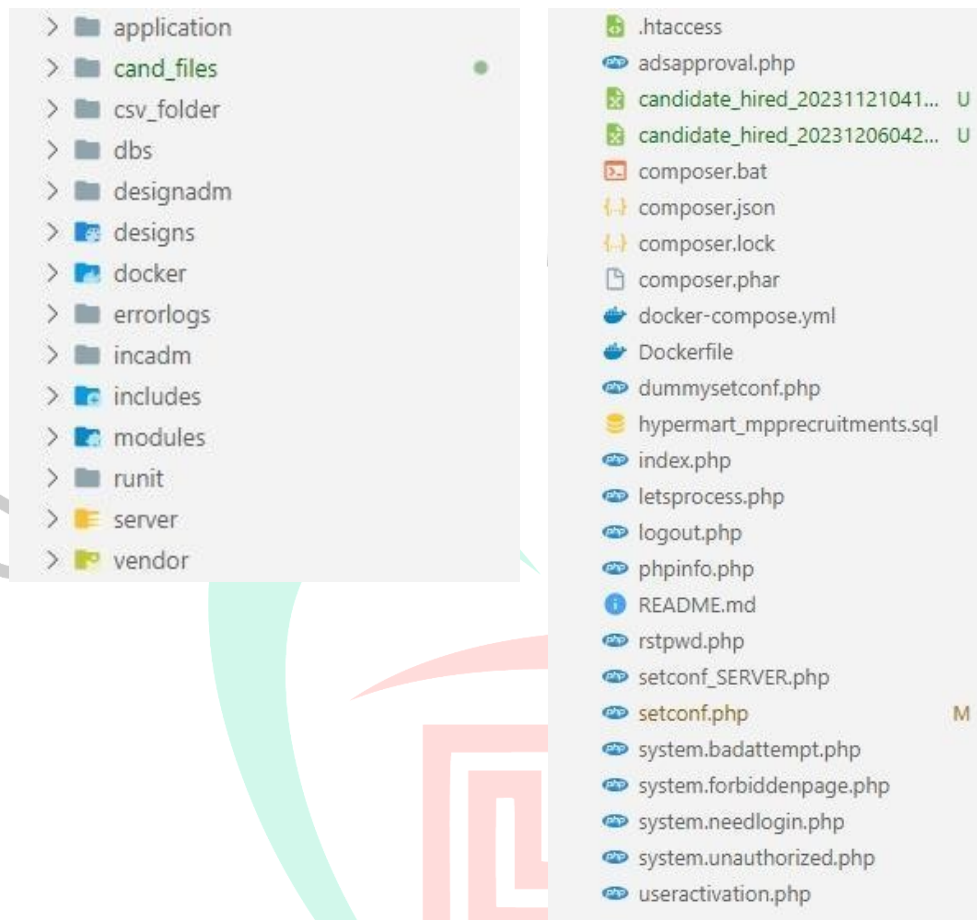
Pembuatan *Back-end* serta *Front-end* pada Aplikasi *MPP-Recruitment* pada kedua fitur tersebut menggunakan berbagai macam alat. Pada bagian *back-end* praktikan ditugaskan untuk menggunakan bahasa pemrograman PHP

versi 8, PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman *script server side* dirancang lebih condong untuk pembuatan dan pengembangan *website*. (Yudhanto & Prasetyo, 2019). Sedangkan pada bagian *front-end* menggunakan sebuah *framework*.

Adapun *framework* yang digunakan dalam membuat tampilan aplikasi *MPP-Recruitment*, yaitu menggunakan Bootstrap 5. Framework ini bersifat fleksibel untuk mengembangkan aplikasi berbasis *website*. Bootstrap merupakan *framework* css yang digunakan untuk membuat tampilan web. Bootstrap menyediakan *class* dan komponen yang sudah siap dipakai. (Nugroho & Setiyawati, 2019).

3.2.2. Development

Dalam melakukan tahap implementasi aplikasi *website MPP-Recruitment* yang sudah dirancang sebelumnya, berikut *database* dan *source code* yang praktikan implementasikan kedalam pengerjaan aplikasi *MPP-Recruitment*. Praktikan menggunakan bahasa pemrograman *PHP Native* versi 8.1 dengan bantuan *library bootstrap* untuk mempercantik tampilan aplikasi, kemudian untuk penyimpanan data praktikan menggunakan *phpMyAdmin* sehingga konfigurasi antara aplikasi dengan penyimpanan data dapat lebih mudah untuk diterapkan kedalam aplikasi.



Gambar 3.16 Struktur Folder Aplikasi *MPP-Recruitment*

Gambar 3.16 merupakan struktur folder yang diterapkan dalam aplikasi *MPP-Recruitment*, folder *modules* digunakan untuk menyimpan fungsi-fungsi yang akan diterapkan dalam folder lainnya, folder *designs* untuk membuat tampilan yang akan diberikan kepada calon pelamar kerja, folder *designadm* digunakan untuk membuat tampilan yang akan diberikan kepada *admin*. Folder lainnya memiliki fungsi masing-masing yang sangat dibutuhkan dalam pembuatan aplikasi tersebut.

Dalam proses pengembangan aplikasi, praktikan memulai perancangan *database* yang dijalankan secara *local* terlebih dahulu dengan menggunakan *phpMyAdmin*. Dengan menggunakan *database phpMyAdmin* praktikan membutuhkan *software* tambahan yaitu XAMPP yang berfungsi sebagai server

untuk menjalankan *MySQL* dan *Apache*. Selain itu *XAMPP* juga dibutuhkan supaya praktikan dapat menjalankan aplikasi *PHP* yang tersimpan di dalam folder *htdocs* *XAMPP*.

Table	Action	Rows	Type	Collation	Size	Overhead
city	★ Browse Structure Search Insert Empty Drop	1,123	MyISAM	latin1_swedish_ci	60.3 KIB	-
eduinstitution	★ Browse Structure Search Insert Empty Drop	443	MyISAM	latin1_swedish_ci	30.9 KIB	-
edumajor	★ Browse Structure Search Insert Empty Drop	182	MyISAM	latin1_swedish_ci	13.4 KIB	-
familyrelation	★ Browse Structure Search Insert Empty Drop	31	MyISAM	latin1_swedish_ci	3.1 KIB	-
log_auth	★ Browse Structure Search Insert Empty Drop	12	MyISAM	latin1_swedish_ci	11.2 KIB	-
m_banner	★ Browse Structure Search Insert Empty Drop	9	MyISAM	latin1_swedish_ci	2.4 KIB	-
m_candidate	★ Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	7.5 KIB	-
m_candidate_edu	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	6.1 KIB	-
m_candidate_family	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	3.1 KIB	-
m_candidate_file	★ Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	2.2 KIB	-
m_candidate_fileothers	★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1.0 KIB	-
m_candidate_jobexp	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	3.1 KIB	-
m_candidate_language	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	3.1 KIB	-
m_candidate_organization	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	3.1 KIB	-
m_candidate_skill	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	3.1 KIB	-
m_candidate_training	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	3.1 KIB	-
m_city	★ Browse Structure Search Insert Empty Drop	497	MyISAM	utf8_general_ci	20.3 KIB	-
m_degree	★ Browse Structure Search Insert Empty Drop	7	InnoDB	latin1_swedish_ci	16.0 KIB	-
m_directorate	★ Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	2.2 KIB	-
m_division	★ Browse Structure Search Insert Empty Drop	4	MyISAM	latin1_swedish_ci	2.2 KIB	-
m_employee	★ Browse Structure Search Insert Empty Drop	9	MyISAM	latin1_swedish_ci	5.8 KIB	-

Gambar 3.17 Database Aplikasi MPP-Recruitment

m_faq	★ Browse Structure Search Insert Empty Drop	5	MyISAM	latin1_swedish_ci	4.3 KIB	-
m_homebase	★ Browse Structure Search Insert Empty Drop	1,156	InnoDB	latin1_swedish_ci	144.0 KIB	-
m_howto	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	66.0 KIB	-
m_job_function	★ Browse Structure Search Insert Empty Drop	77	MyISAM	latin1_swedish_ci	4.0 KIB	-
m_job_vacancy	★ Browse Structure Search Insert Empty Drop	129	MyISAM	latin1_swedish_ci	26.0 KIB	-
m_job	★ Browse Structure Search Insert Empty Drop	59	MyISAM	utf8_general_ci	3.2 KIB	-
m_mail_template	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KIB	-
m_menu	★ Browse Structure Search Insert Empty Drop	54	MyISAM	latin1_swedish_ci	9.1 KIB	-
m_question	★ Browse Structure Search Insert Empty Drop	25	MyISAM	latin1_swedish_ci	6.2 KIB	-
m_referall	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KIB	-
m_register	★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	2.1 KIB	140 B
m_required	★ Browse Structure Search Insert Empty Drop	14	MyISAM	latin1_swedish_ci	3.5 KIB	-
m_university	★ Browse Structure Search Insert Empty Drop	4,521	MyISAM	latin1_swedish_ci	370.2 KIB	-
race	★ Browse Structure Search Insert Empty Drop	67	MyISAM	latin1_swedish_ci	3.8 KIB	-
t_answer	★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1.0 KIB	-
t_apply_history	★ Browse Structure Search Insert Empty Drop	2	MyISAM	latin1_swedish_ci	2.2 KIB	-
t_candidate_apply	★ Browse Structure Search Insert Empty Drop	2	MyISAM	latin1_swedish_ci	3.2 KIB	-
t_sync_point	★ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16.0 KIB	-

Gambar 3.18 Database Aplikasi MPP-Recruitment

Gambar 3.17 dan **Gambar 3.18** merupakan daftar seluruh tabel yang terdapat pada *database* yang digunakan dalam mengembangkan aplikasi *MPP-Recruitment*. Beberapa tabel tersebut saling berhubungan dalam operasional data untuk menghindari adanya duplikasi pada suatu atribut/kolom. Masing-masing tabel digunakan dengan kebutuhannya masing-masing seperti tabel *register* untuk menampung data calon kandidat yang telah melakukan registrasi

pada halaman *website*, *m_city* untuk menyimpan seluruh data kota yang berada di Indonesia, dan tabel *t_candidate_apply* untuk menyimpan data calon kandidat yang telah melakukan lamar kerja pada posisi yang ditentukan oleh *user*.

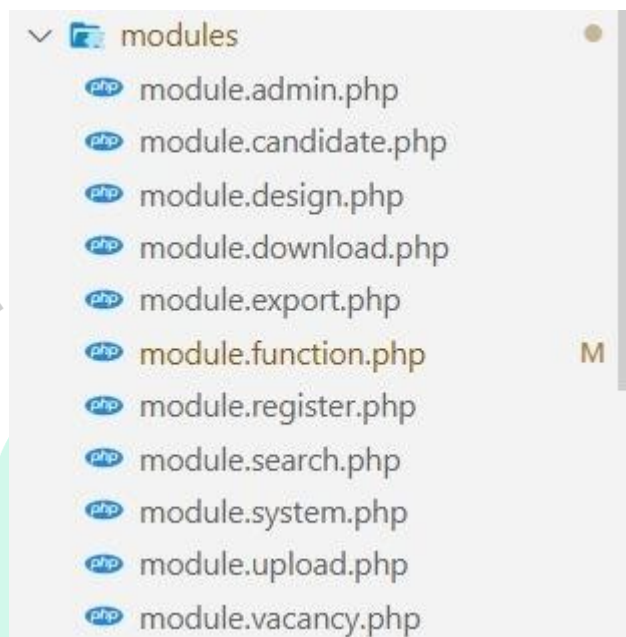
Pada dasarnya seluruh tabel yang terdapat pada *database* memiliki peran yang sangat penting berdasarkan kebutuhannya. Namun, *MPP-Recruitment* memiliki beberapa tabel yang merupakan bagian utama dibangunnya aplikasi berbasis *website* antara lain, *m_candidate*, *t_candidate_apply*, dan *log_auth*. Tabel *m_candidate* berfungsi untuk menerima seluruh data lengkap calon pelamar berdasarkan data-data yang dibutuhkan oleh departemen *Human Resource*, Tabel *t_candidate_apply* berfungsi untuk melakukan pembaharuan status calon pelamar berdasarkan tahap-tahap yang telah ditentukan oleh tim, dan *log_auth* untuk memberikan akses *login* berdasarkan *role* yang telah ditentukan oleh *super admin*.

Pada pengembangan aplikasi ini, implementasinya dilakukan menggunakan bahasa pemrograman *PHP Native* versi 8.1.

```
function adm_getNextStage($current_stage) {  
    switch ($current_stage) {  
        case "cv-screening":  
            $nextstage="user-screening"; break;  
        case "user-screening":  
            $nextstage="psychotest"; break;  
        case "psychotest":  
            $nextstage="hr-interview"; break;  
        case "hr-interview":  
            $nextstage="user-interview"; break;  
        case "user-interview":  
            $nextstage="background-check"; break;  
        case "background-check":  
            $nextstage="offering"; break;  
        case "offering":  
            $nextstage="final"; break;  
  
        default:  
            $nextstage="cv-screening";  
    }  
  
    return $nextstage;  
}
```

Gambar 3.19 Module Admin

Gambar 3.19 merupakan sebuah fungsi dengan nama `adm_getNextStage` yang berfungsi untuk melakukan *update* status lamaran pekerjaan masing-masing kandidat. Alur kerja pada potongan sumber kode di atas melibatkan kondisi “*switch*” dengan parameter `$current_stage`, kondisi *switch* memerlukan berbagai macam kasus antara lain, *cv-screening*, *user-screening*, *psychotest*, *hr-interview*, *user-interview*, *background-check*, dan *offering*. Kondisi status calon pelamar akan mengikuti sesuai dengan Keputusan yang telah ditentukan oleh admin, dengan kasus bawaan yaitu “*cv-screening*”. Apabila admin melakukan *update* dengan memberikan kondisi berhasil maka akan menuju ke tahap selanjutnya.



Gambar 3.20 Modules Aplikasi

Gambar 3.20 merupakan kumpulan seluruh *modules* yang ada pada aplikasi *MPP-Recruitment*. Fungsi dari *folder* ini adalah untuk menyatukan seluruh fungsi-fungsi yang nantinya akan diimplementasikan ke dalam aplikasi, sehingga struktur *project* dapat lebih mudah dipahami dan terstruktur. Kegunaan *module admin* untuk membangun sistem *create*, *read*, *delete*, dan *update* serta memonitoring seluruh data yang terdapat pada database, *module candidate* digunakan untuk melakukan eksekusi perintah *apply job* sesuai dengan pilihan pengguna, melakukan update data diri, dan *referall person*, lalu *module function* digunakan untuk proses eksekusi bagian dalam sebuah aplikasi antara lain, *simple mail transfer protocol*, *resize image*, dan proses enkripsi *password*. Kemudian *module export* digunakan oleh admin dalam melakukan export data kandidat ke dalam bentuk excel atau csv, *module download* digunakan oleh admin dalam memperoleh data kandidat dalam format pdf, mengenai cara kerja fungsi tersebut dengan mengubah penulisan struktur kode HTML ke dalam format PDF menggunakan bantuan library *mpdf*. Selanjutnya *module system* digunakan untuk proses utama pada aplikasi, dengan adanya *module system* aplikasi dapat berjalan dalam mengeksekusi proses login, reset password, change password, dan alert. Lalu *module register* digunakan untuk proses

registrasi akun baik untuk admin maupun pengguna dengan menerapkan fungsi decode password yang telah dibentuk dalam file module function. Module search digunakan pada halaman admin untuk melakukan pencarian data berdasarkan kategori yang dibutuhkan oleh admin, fungsi ini dibentuk oleh praktikan untuk mempermudah dalam proses pengelolaan data yang sangat banyak. Module vacancy digunakan untuk proses pengecekan required dalam melakukan proses *apply job, required* yang dimaksud adalah kelengkapan data diri serta dokumen-dokumen yang dibutuhkan antara lain, *curriculum vitae*, Nomor Induk Kependudukan, NPWP, dan dokumen pendukung. Module upload pada aplikasi ini digunakan untuk proses unggah foto maupun dokumen yang telah disediakan pada aplikasi.

```

if($server=="localhost"){
    $_database["sqlhost"] = "localhost"; // local
    $_database["username"] = "root";
    $_database["password"] = "";
    $_database["databasename"] = "mpp_recruitment"; //develop
    //$_database["databasename"] = "mpprecruitment0419";
    define("_PATHURL","http://localhost/mpprecruitment");
}

define("_PATHDIRECTORY",dirname(__FILE__));
define("_PATHERRORLOG",_PATHDIRECTORY."/errorlogs");
define("_DESIGNPATH",_PATHDIRECTORY."/designs");
define("_DESIGNWEBPATH",_PATHURL."/designs");
define("_INCLUDEPATH",_PATHURL."/vendor");
define("_INCLUDEDIRECTORY",_PATHDIRECTORY."/vendor");

define("_IMAGEWEBPATH",_DESIGNWEBPATH."/images");
define("_CANDFILES",_PATHURL."/cand_files");
define("_DIRFILES",_PATHDIRECTORY."/cand_files");

define("_DATAONNTYPE",2); //1: mysql 2: mysqli

define("_WITHCAPTCHA","n");
define("_DEFAULTLANG","en");
define("_MAXAPPLY",3);

$connection_type = _DATAONNTYPE;

```

Gambar 3.21 Config Website MPP-Requirement

Gambar 3.21 merupakan sebuah fungsi untuk menghubungkan koneksi antar *website* dan *database*, alur kerja dari potongan sumber kode di atas yaitu terdapat kondisi "if" dimana jika status server tersebut berisikan parameter

localhost maka akan menjalankan perintah-perintah koneksi *database* dengan membutuhkan *sqlhost*, *username*, *password*, dan *databasename*. Selanjutnya, praktikan mendefinisikan seluruh *path* dengan lokasi tujuan *folder* yang sesuai dengan kebutuhan di dalam *website*, seperti “_DESIGNPATH” yang digunakan untuk mengarahkan lokasi tujuan ke dalam *folder/design*. Nantinya alamat *website* akan mengikuti ketentuan yang sudah ditetapkan, misalnya <http://localhost/mpprecruitment/designs>.

```
<?php
function url_slug($v)
{
    $slug = preg_replace('/[^A-Za-z0-9-]+/', '-', $v);
    return strtolower($slug);
}

function sha256mod($keyword,$salt="shaktirec") {
    $hashed = crypt($keyword,$salt);
    //echo $hashed."<br><br>";
    $hashed = str_replace('$5$rounds=5000$', "", $hashed);
    //echo $hashed; exit;
    return $hashed;
}

function coded ($keyword, $pwd="shaktimpp") {
    $codedkey = base64_encode(openssl_encrypt($keyword, 'AES-128-CBC', md5($pwd), OPENSSL_RAW_DATA, md5(md5($pwd), TRUE)));
    $codedkey = base64_encode($codedkey);
    return $codedkey;
}

function decoded($keyword,$pwd="shaktimpp") {
    $keyword = base64_decode(strval($keyword));
    $decodedkey=openssl_decrypt(base64_decode($keyword), 'AES-128-CBC', md5($pwd), OPENSSL_RAW_DATA, md5(md5($pwd), TRUE));
    return $decodedkey;
}
```

Gambar 3.22 Module Function

Gambar 3.22 merupakan sejumlah fungsi yang berhasil dikembangkan oleh praktikan. Fungsi-fungsi ini dirancang untuk meningkatkan tingkat keamanan pada sebuah *website*. Salah satu fungsi yang terdapat adalah *url_slug*, yang bertujuan mengonversi string menjadi *slug* agar lebih mudah dibaca dalam penulisan URL. Penggunaan *preg_replace* dalam fungsi ini membantu menggantikan karakter *non-alphanumeric* dan *non-dash*, Selanjutnya, fungsi *sha256mod* memiliki dua parameter, *\$keyword* dan *\$salt*, dengan nilai default "shaktirec". Fungsi ini digunakan untuk menghasilkan nilai *hash* sebagai bagian dari upaya untuk memperkuat keamanan. Sementara itu, fungsi *coded* dan *decoded* juga memiliki dua parameter yang sama. Fungsi *coded* bertugas menghasilkan nilai yang telah berhasil dikonversi menjadi *hash*, sedangkan fungsi *decoded* berfungsi mengembalikan nilai yang telah dikonversi

menjadi nilai awalnya. Keseluruhan dari fungsi-fungsi ini bertujuan untuk meningkatkan tingkat keamanan pada *website*.

```
function sendConfirmationEmail($data)
{
    $variablemail = array();
    $variablemail["sender"] = 'rhnashil@gmail.com';
    $variablemail["from"] = "rhnashil@gmail.com";
    $variablemail["fromname"] = "Recruitment PT. Matahari Putra Prima";
    $variablemail["to"] = $data["candidate_email"];
    $variablemail["toName"] = $data["candidate_name"];
    // $variablemail["bcc"] = "shakti.santoso@hypermart.co.id";
    $variablemail["subject"] = "[PT. Matahari Putra Prima] Job Application";
    $variablemail["content"] = "
    <p>Dear ".$data["candidate_name"].", Thank you for applying for ".$data["job_vacancy_name"]." position.<br><br>
    Your application has been sent to our recruitment team on ".date("d F Y H:i:s").".
    We will review your application and contact you if you are selected for further selection process.<br><br>
    Sincerely,<br>
    Recruitment PT. Matahari Putra Prima</p>";

    $emailSent = function_sending_email($variablemail);
    return $emailSent;
}
```

Gambar 3.23 Module Vacancy

Gambar 3.23 merupakan sebuah fungsi untuk melakukan proses pengiriman *email*, fungsi ini berguna untuk memberikan pesan informasi secara langsung ke alamat email calon pelamar. Dalam proses pengiriman email membutuhkan *library* tambahan yaitu *Simple Mail Transfer Protocol* yang telah dikonfigurasi sesuai dengan *host domain* perusahaan, fungsi tersebut sudah menetapkan secara *default* untuk pengirim *email*, *subject*, dan *content*, tetapi tujuan dari *email* akan menyesuaikan dengan alamat *email* calon kandidat yang telah berhasil melakukan *apply job*.

```
if (isset($_SESSION["loggedin"]) && $_SESSION["loggedin"]==1 && isset($_SESSION["priv_id"])
&& in_array($_SESSION["priv_id"],array("admin","hrd","pic"))) //admin template
{
    $menuadm = system_getMenu();
    $titleadm = (isset($menuadm[0]["menu_title"]) && $menuadm[0]["menu_title"]<">" )?$menuadm[0]["menu_title"]." | ":"";

    include_once(_PATHDIRECTORY."/designadm/adm.head.php");
    include_once(_PATHDIRECTORY."/designadm/adm.body.php");
    include_once(_PATHDIRECTORY."/designadm/adm.footer.php");
}
else
{
    $menu = system_getMenu();

    $title = (isset($menu[0]["menu_title"]) && $menu[0]["menu_title"]<">" )?$menu[0]["menu_title"]." | ":"";

    include(_DESIGNPATH."/ds.head.php");
    include(_DESIGNPATH."/ds.body.php");
    include(_DESIGNPATH."/ds.footer.php");
}
```

Gambar 3.24 File Index

Gambar 3.24 merupakan sebuah kondisi, dimana kondisi tersebut akan terpenuhi jika berhasil melalui tahap pengecekan terhadap suatu variabel, variabel tersebut mencakup *session* akun yang akan login dan akun tersebut memiliki role admin, hrd, dan pic. Kondisi tersebut digunakan untuk pembatas antara *role* akun, Apabila *role* yang sedang *login* adalah admin, hrd, dan pic maka akan menjalani proses pemanggilan *directory* “designadm” untuk *head*, *body*, dan *footer*. Jika kondisi tidak terpenuhi maka akan menjalani proses pemanggilan *directory* “ds” untuk *head*, *body*, dan *footer*.

3.3 Kendala Yang Dihadapi

Selama kegiatan MBKM di PT. Matahari Putra Prima Tbk, praktikan mengalami beberapa kendala, baik dari perancangan aplikasi serta proses implementasi rancangan tersebut. Berikut merupakan beberapa kendala yang dialami oleh praktikan.

1. Kurangnya sumber daya manusia pada departemen HR yang memiliki keahlian dibidang IT, sehingga praktikan merasa kesulitan dalam berdiskusi mengenai aplikasi yang sedang dikembangkan.
2. Selama kegiatan MBKM, praktikan mendapatkan tugas untuk melakukan dokumentasi pada sebuah acara. Hal tersebut menyebabkan terjadinya perubahan jadwal kegiatan yang telah disusun oleh praktikan.

3.4 Cara Mengatasi Kendala

Berikut merupakan beberapa cara yang dilakukan oleh praktikan dalam menghadapi kendala selama melaksanakan kegiatan MBKM di PT. Matahari Putra Prima, diantaranya:

1. Melakukan konsultasi dengan pembimbing eksternal dan kepala HRIS terkait menyesuaikan alur aplikasi, mencari sumber informasi dari luar perusahaan menggunakan platform-platform yang memberikan edukasi mengenai IT.
2. Membuat penjadwalan terkait kegiatan apa yang harus dikerjakan terlebih dahulu dan mengatur tugas-tugas dengan skala prioritas.

3.5 Pembelajaran Yang Diperoleh dari Kerja Profesi

Selama kegiatan MBKM di PT. Matahari Putra Prima Tbk, praktikan memperoleh berbagai macam pembelajaran, antara lain :

1. Praktikan mendapatkan pembelajaran mengenai kemampuan menganalisis dan perancangan pembuatan aplikasi berbasis *website*.
2. Praktikan mendapatkan pembelajaran *software-software* yang sebelumnya tidak pernah digunakan seperti *docker*, *liferay*, dan *nginx*.
3. Praktikan mendapatkan pembelajaran mengenai berbagai macam library yang dapat diterapkan dalam aplikasi berbasis *website*.
4. Praktikan mendapatkan pengalaman serta pembelajaran bagaimana mengatur waktu dan membagi waktu agar lebih baik.