

**REMPMS - RECRUITMENT AND EMPLOYEE MANAGEMENT SYSTEM**

**PROJECT PROPOSAL**

**Version : Draft**

**Author : P. A. Maleesha Sandakalum**

**Date : 04th January 2023 - Present**

**Project : REMPMS - Recruitment and Employee Management System**

Introduction: The proposed system of REMPS-Recruitment and Employee Management System includes modules like,

1st Stage :

**Admin Module: Roles** | Administrator | Administrator Dashboard

1. The administrator should be able to create user roles.
2. The administrator should be able to create user accounts for employees and system users according to the roles and Sync them to user service.
3. The administrator should be able to create communication and location information for the use of candidates and employees.
4. The administrator should be able to create job categories and sync them to the recruitment service db for the recruitment officer’s use.

**Recruitment Module: Roles** | Recruitment Officer | Recruitment Off.Dashboard

1. The recruitment officer should be able to create job categories.
2. The recruitment officer should be able to create employers.
3. The recruitment officer should be able to create a job vacancy with a poster. (KT: Multipart file).
4. The recruitment officer should be able to receive and view the applied CVs’ of each candidate.
5. Generate reports:
   1. The recruitment officer should be able to view candidates' detailed list of who applied for each job vacancy. (Generate the details through Excel sheet and pdf - with the Google Drive link in Excel sheet) by filtering particular job vacancies. (KT: PDF and Excel generate)
   2. The recruitment officer should be able to create PDFs of all the vacancy’s candidate details compressed into a single bulk. (KT: Zip with multiple pdf’s)

**Candidate Module: Roles** | Candidate | Website - Candidate Dashboard

* Under website:

1. A candidate should be able to view all the posted job vacancies.
2. A candidate should be able to search for job vacancies by job category.
3. A candidate should be able to search for job vacancies by name.
4. A candidate should be able to preview a job poster with details by vacancy.
5. A candidate should be able to send a CV pdf for a vacancy.
6. A candidate should be able to receive the sent CV for his provided email address for verification of the application. (KT: Email service)
7. A candidate should be able to sign up to the system.
8. A candidate should be able to log into the system.
9. A candidate should be able to generate his/her own Cv by system.

* Under candidate dashboard | My Jobs section: (Information on jobs, submitted applications, and companies you are watching.)

1. My Application: View all the applied vacancies with the applied CV document.
2. Flagged Jobs: View all the flagged / favorite job vacancies.
3. Preferred Companies: View all the preferred companies and their available vacancies.

* Under candidate dashboard | My Profiles section: (Manage your CV, photograph, certificates, online profile, and view/edit your login details.)

1. My CV/Documents: view the candidate's CV documents.
2. Common Profile: View and edit all the personal, professional, higher education, school education, membership, language proficiency, achievements, referees, family information, job preference, preferred job location, and uploaded CVs/Certificates.
3. My Login Details: Update login details, user name, password, c-password, email, and c-email.

* Under candidate dashboard | My Preferences section: (Job alert preferences with notifications.)

1. The candidate should be able to create job alerts by selecting category and employer.
2. The candidate should be able to remove created job alerts.
3. The candidate should be able to give an email address to which alerts will be sent.

2nd Stage :

**Employee Module: Roles** | HR Officer| Finance Officer | Employee Dashboard

* Under employee dashboard | Employee section: (A stepper with Employee, Profile, Document)

1. An HR officer should be able to manage employee details.
2. An HR officer should be able to create employee profiles by assigning a company, a job title, and a branch.
3. An HR officer should be able to upload employee documents.

* Under employee dashboard | Attendance section: (A stepper with Attendance, and Report.)

1. An HR officer should be able to mark attendance for each day.
2. An HR officer should be able to get attendance reports in Excel for each employee's attendance.
3. An HR officer should be able to confirm each employee's attendance.) (For salary calculation).

* Under employee dashboard | Finance section: (A stepper with Salary calculation, and Report.)

1. A finance officer should be able to calculate the salaries of each employee for each month.
2. A finance officer should be able to generate an employee salary slip and issue it to the employee through an email for each month.
3. A finance officer should be able to generate bulk zips with employees' salary report pdfs company-wise, section-wise, or branch-wise in a particular month.

**Technology Stack:**

MICROSERVICES

+

MICRO FRONT END(New)

*(Micro-frontend architecture is a design approach in which a front-end app is decomposed into individual, semi-independent “micro-apps” working loosely together. The micro-frontend concept is vaguely inspired by, and named after, microservices.)*

**Back-End :**

| **Programming language** | Java 21 |
| --- | --- |
| **Framework** | Spring Boot |
| **Architectural pattern** | Microservices with consul discovery agent(new) |
| **IDE** | IntelliJ IDEA Ultimate |
| **Git-based code hosting and collaboration tool.** | Bitbucket |
| **Graphical user interface (GUI) desktop client, interacts with Git repositories to fully concentrate on coding.** | Source Tree |
| **Free and open-source relational database management system (RDBMS).** | Postgresql(new) |
| **Developers, database administrators, analysts, and everyone.** | DBWear(new) |
| **API platform for building and using APIs.** | POSTMAN |
| **Testing** | JUnit and Mokito Testing Framework |

**Front-End :**

| **Open-source, JavaScript framework written in TypeScript.** | Angular 17(new) |
| --- | --- |
| **Front End Architectural pattern** | MicroFront End(new) |
| **IDE** | WebStorm(new) |