RevWorkforce

(Application Development)

Center of Excellence
10-May-2023

Version 1.0

Table of Contents

Application Overview 3
Core Functional Requirements 3
Standard Functional Scope 4
Definition of Done 4
Non-Functional Expectations 8

Application Overview

The Rev Workforce project is an enterprise application focused on HRM (Human Resource Management). The project has core functional requirements for employees, managers, and admins, which includes features such as leave management, performance reviews, goal setting, and employee management.

Core Functional Requirements

Employee:

As an employee, I should be able to:

1. Login to my account.

Leave Module:

1. Get details about available leaves.
2. Apply for a leave/leaves.
3. Get details about applied leaves.
4. Get holiday data.
5. View the manager's name while applying for leave.
6. Edit basic profile information like phone number and address.
7. Get email notification when leave is approved or rejected.

Performance Module:

1. Enter the performance review document with deliverables, achievements, and areas of improvement in a textual form.
2. Create the goals and targets for the year.
3. Each goal and target will have a deadline and weightage.
4. Get notified when the manager provides feedback for the performance review.

Additional:

1. View the birthdays and work anniversaries of the employees.

Manager:

Leave Module:

As a manager, I should be able to:

1. See the list of the employees who report to me.
2. Approve or reject leaves.
3. Get email notification when an employee applies for leave.
4. Interact with a dummy API to update a payroll service for changes in paid or unpaid leave.
   - This may include endpoints to update payroll records based on approved leaves.
   - The dummy service will periodically send error status codes, requiring proper handling of HTTP status codes

Performance Module:

As a manager, I should be able to:

1. Review and provide feedback and score for the self-performance review provided by the employees who report to me.
2. Get an email notification when an employee submits the performance review.
3. Include reports representing the results of the performance reviews for their team.

Note: All managers are employees, so an employee's core functional scope is applicable to the manager.

Standard Functional Scope

Registered users should be able to log in, change the password and request for a forgotten password (will be sent to their registered email).

Definition of Done

- Working application demonstration.
- Sharing the associates' code repo for technical evaluation with:
  - ERD Diagram

- Architecture Diagram

Competency wise scoping

Competency	Application Type	Expectations
Web Fundamentals (HTML, CSS and JS)	Web Navigational Prototype	**User Experience:** 1. Have an intuitive design for the user to work with the application without any training or guidance 2. Have clean & consistent UI, color theme and easy to use navigations **User Inputs & outputs:** 1. Have appropriate HTML fields for the user inputs 2. Wherever possible use the client-side validations for the user input 3. Display the appropriate user info/error message with appropriate colors and icons **Performance:** 1. Use compressed images / assets to increase the page performance **Dataset:** 1. For any prepopulated data such as to render table rows use JSON fileas the DataSource and use standard open-source library to read and render in the web pages. **General standards:** 1. Ensure the w3 standards are implemented for better accessibility. E.g., Using alt attribute for image tag. 2. Ensure the SEO recommended meta tags are added.
Web Development with React	Enhanced Web Navigational Prototype using React	**Same expectation as like for the Web fundamental competency with the below additions** **Framework Specific** 1. Ensure the appropriate APIs are used for any of the API calls 2. Ensure the routing is centrally configured 3. Best practices & design patterns are to be followed 4. Implement the end-to-end testing framework and get to know the headless execution of end-to-end framework. **Deployment artifacts:** 1. The deployment artifacts should be minified and obfuscated if required. **Security**: 1. Ensure the CORS restriction is applied, if applicable. 2. Ensure Route Guarding/Authenticated Routing is implemented. 3. Ensure that the secrets are stored as environment variables using secure credential storage.
ASP.NET API REST	Identifying and Developing All	**REST Standards:**

	APIs for any front-end application to consume	1. Ensure the REST standards are followed for API naming, HTTP Operation and Response (output definition) 2. Secure the protected APIs 3. Define a common URL pattern for public and secure APIs 4. Proper documentation of APIs with Input and Output Samples to be documented 5. Provide / enable to API gateway to route the request through a single channel **Logging:** 1. Ensure the application is using proper logging framework and methods. 2. Ensure the application's log level is configured using configuration files so that it can be changed without changing the code. 3. Also ensure that the application logging is configured to output to the mentioned log file. 4. Ensure the centralized logging implementation **Testing**: 1. Ensure sufficient test cases are written using appropriate testing frameworks. 2. Ensure the code coverage closed to be 80% **Security**: 1. Ensure the CORS restriction is applied. 2. SQL Injection thread is taken care 3. Throttling is to be taken care of. 4. Cross site Scripting to be avoided. 5. Ensure that the secrets are stored as environment variables using configuration files or secure credential storage.
DevOps No Cloud with Podman for Developers	Implementing DevOps Practices and CI/CD Pipelines for Containerized Applications with Jenkins and Podman	1. Implement DevOps best practices, emphasizing collaboration, automation, and continuous improvement throughout the software development and deployment lifecycle. 2. Leverage containerization with Podman to meet specific application hosting and deployment needs, reducing reliance on cloud services. 3. Implement security measures tailored for containerized environments to protect applications and data.

Non-Functional Expectations

1. Application development supposed to follow the Scrum process
2. Application password should be encrypted using appropriate hashing algorithms