Student Name: MOHAMMEDSAMEER S
Seat No: 249
Project ID: 9
Project title: RewardPoints Calculator

Technical Components:

Components	**TechStack**
Backend	Express js
Frontend	Vue js
Database	Mongo db
API	RESTful services

Implementation Timeline:

Phase	**Deadline**	**Status**	**Notes**
Stage 1	02/05/2024	Under review	Planning and Requirement gathering
Stage 2	-	In progress	Design and Prototyping
Stage 3	-	Not started	DB Designing
Stage 4	-	Not started	Backend Implementation
Stage 5	-	Not started	Testing & Implementation

PROBLEM STATEMENT:

1. **Inefficient Evaluation Process:** The absence of a centralized platform for evaluating project submissions leads to inefficiencies in the assessment process. Admins and reviewers may use disparate methods and tools, resulting in inconsistencies and delays in providing feedback.

2. **Lack of Accountability:** Without a structured system in place, it's challenging to track the progress of evaluations and ensure accountability among reviewers and admins. This can lead to overlooked submissions, delayed assessments, and dissatisfaction among project stakeholders.

3. **Difficulty in Feedback Management:** Managing feedback from multiple reviewers and admins poses a significant challenge. Collating and synthesizing diverse feedback becomes cumbersome, making it difficult for project teams to implement necessary improvements effectively.

4. **Risk of Bias and Unfairness:** The decentralized nature of evaluation allows for the potential introduction of biases or unfairness in the assessment process. Without proper oversight and control mechanisms, certain projects may receive preferential treatment or face unjust criticism, impacting the overall fairness and credibility of the evaluation system.

5. **Complexity in Data Management:** With various assessment components such as review marks, team communication scores, worklogs, and plagiarism checks, handling and organizing the associated data become complex. Without a centralized database and streamlined data management practices, there's a risk of errors, data loss, and difficulty in generating comprehensive reports for stakeholders.

PROJECT-FLOW:

Purpose:

To develop a comprehensive web application that facilitates the evaluation of project submissions, ensuring fairness, efficiency, and accountability in the assessment process. The aim is to streamline the evaluation workflow, resolve inconsistencies, and enhance the overall quality of feedback provided to project teams.

Scope:

The system encompasses user authentication, submission evaluation forms, result management, and data visualization features. It allows admins and reviewers to assign marks, monitor progress, and adjust assessments as needed. Additionally, it includes functionalities for managing worklogs, plagiarism checks, and overall project performance tracking.

Business Context:

The "RewardPoints Calculator" is designed to optimize the evaluation process within the organization, promoting transparency, fairness, and effectiveness. By providing a centralized platform for assessment, it aims to improve communication between stakeholders, enhance project quality, and ultimately contribute to organizational success. Primary stakeholders include project teams, admins, reviewers, and management.

Considerations:

- Authentication: Utilizing username and password authentication for all users ensures secure access to the system.
- Accessibility: Assuming users have access to internet-enabled devices ensures widespread accessibility and usability of the web application.

This project aims to revolutionize the evaluation process, addressing critical challenges faced by the organization and fostering a culture of excellence and collaboration among project stakeholders.

Dependencies:

1. **axios:** A promise-based HTTP client for making asynchronous HTTP requests in JavaScript environments. It's commonly used for handling AJAX requests and interacting with RESTful APIs.

2. **core-js:** A modular standard library for JavaScript that provides polyfills for ECMAScript standard features that are not implemented in older browsers. It ensures compatibility across different browser versions.

3. **mitt:** A tiny event emitter library for JavaScript. It provides a simple and efficient way to manage events and event listeners within your application.

4. **vue:** The core Vue.js library, which is used for building user interfaces and single-page applications. It provides reactive data binding, component-based architecture, and a rich ecosystem of tools and libraries.

5. **vue-router:** The official router for Vue.js applications, used for managing application navigation and routing. It enables the creation of dynamic, single-page applications with multiple views and routes.

User Personas:

1. Faculty: Responsible for submitting project evaluations, providing constructive feedback, and ensuring fairness in assessments.

2. Admin Staff: Manages system operations, including resolving conflicts, modifying marks as necessary, and approving evaluation submissions.

User Stories:

1. As a faculty member, I need to be able to log in to the system to evaluate project submissions, provide detailed feedback to students, and adjust marks if necessary based on the evaluation criteria.
2. As an admin staff member, I require access to the system to manage user accounts, resolve disputes, ensure consistency in evaluation processes, and generate reports on project evaluations and overall system performance.
3. These user stories reflect the diverse needs and roles of the stakeholders involved in the "RewardPoints Calculator" system, ensuring that it meets the requirements of students, faculty, and administrative staff alike.

Functional Requirements:

1. **User Authentication and Roles:**

    - Implement authentication using username and password for both admins and reviewers.
    - Define two user roles: admin and reviewer, each with specific permissions.

2. **Admin and Reviewer Dashboards:**

    - Provide separate dashboards for admins and reviewers.
    - Admin dashboard allows assigning marks for initial submission, final report, and modifying reviewer-entered marks.
    - Reviewer dashboard facilitates giving marks for review quality and team communication, and viewing/editing own marks.

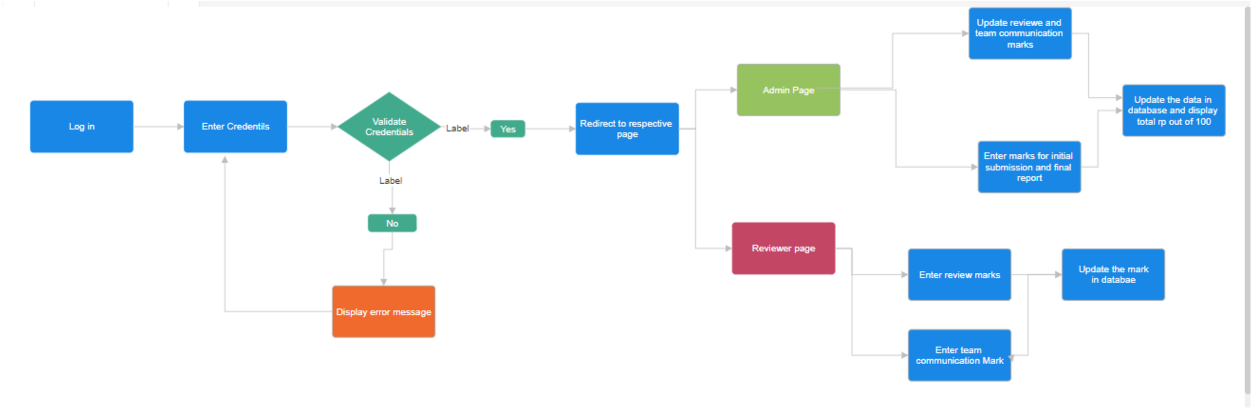
3. **Project Data Management:**

    - Store project details, worklog, and plagiarism marks in the database.
    - Each project identified by a unique project ID.

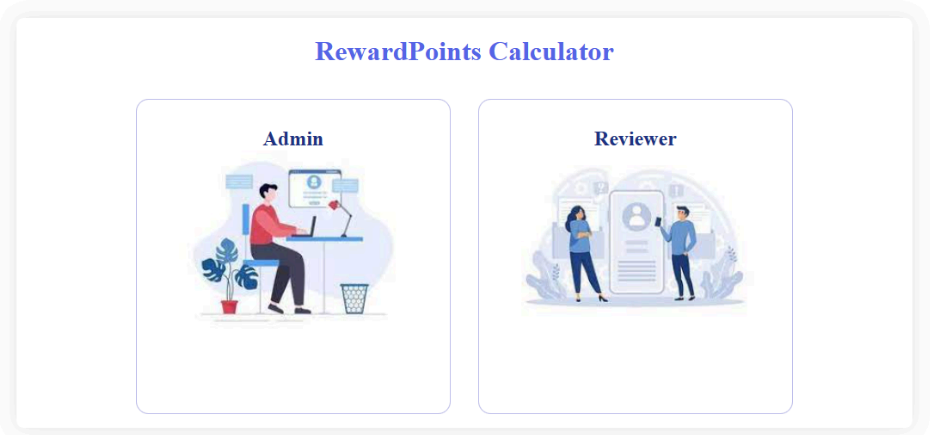
4. **Calculation of Reward Points (RP):**

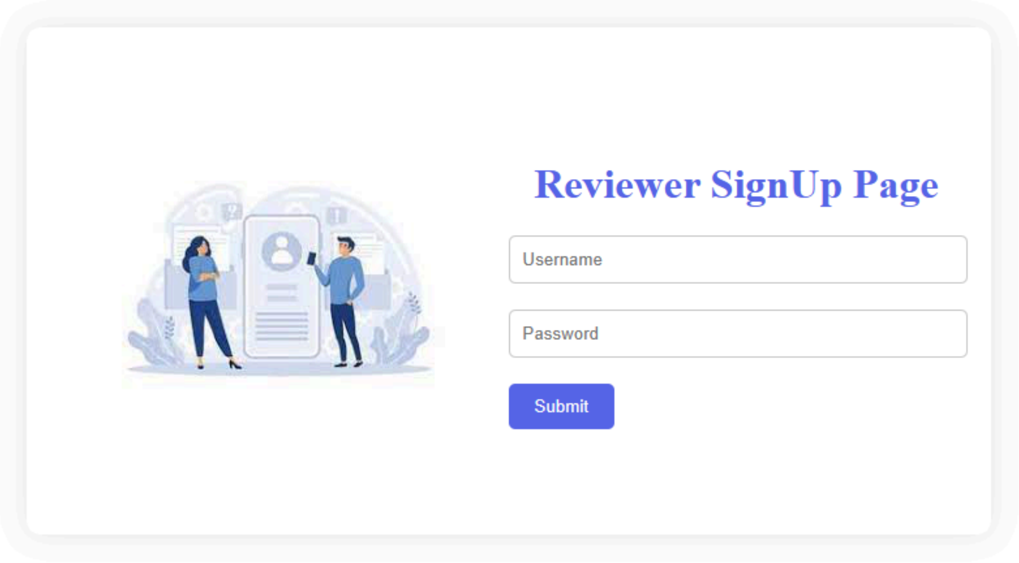
- Calculate RP out of 100 based on admin and reviewer-assigned marks, worklog, and plagiarism marks.
- Summarize RP calculation for each project.

FLOW CHART: ER DIAGRAM (DB DESIGN):

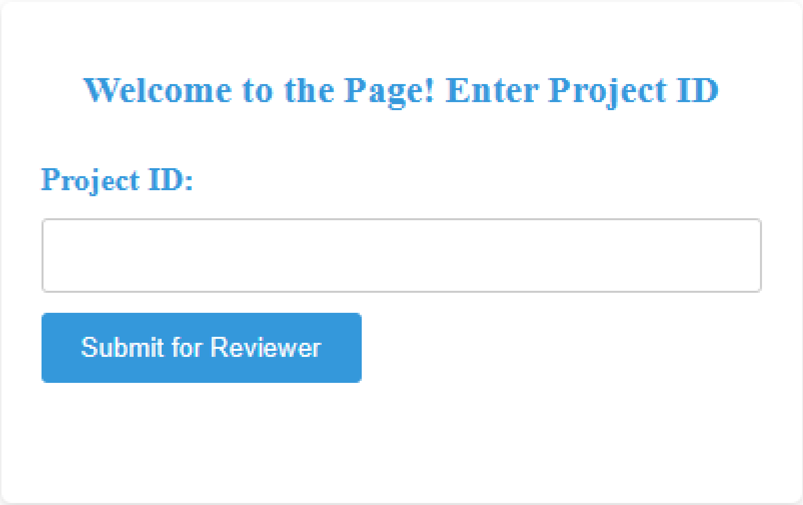

Log in
Enter Credentils
Validate Credentials
Label
Yes
Label
No
Display error message
Redirect to respective page
Admin Page
Update reviewe and team communication marks
Enter marks for initial submission and final report
Update the data in database and display total rp out of 100
Reviewer page
Enter review marks
Enter team communication Mark
Update the mark in databae

User Inteface:


RewardPoints Calculator

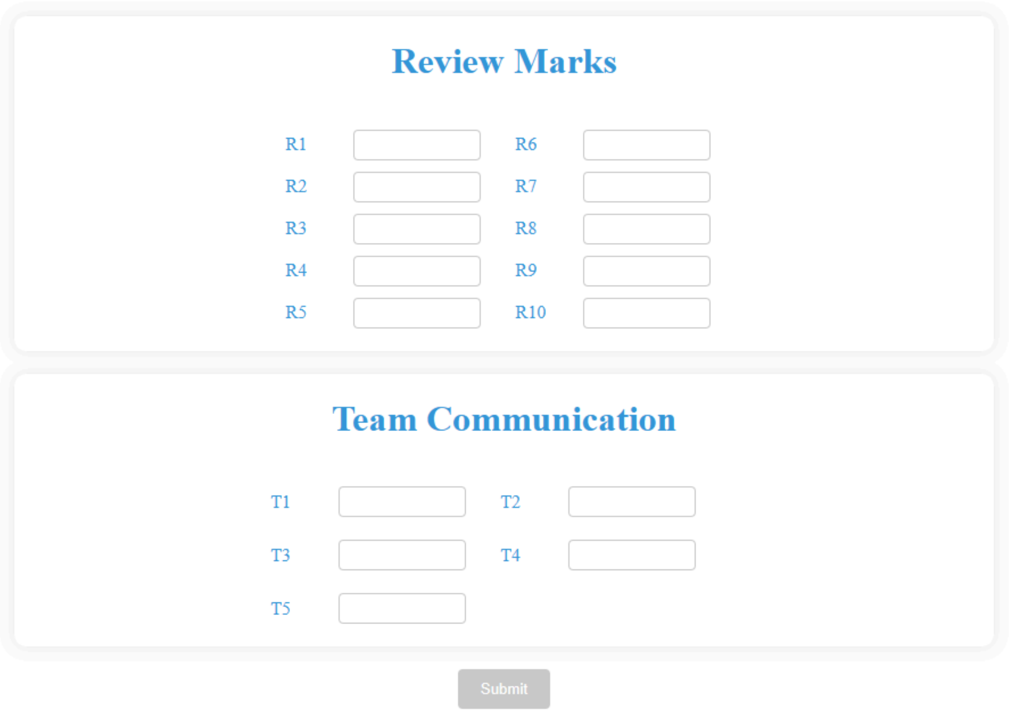


Reviewer SignUp Page



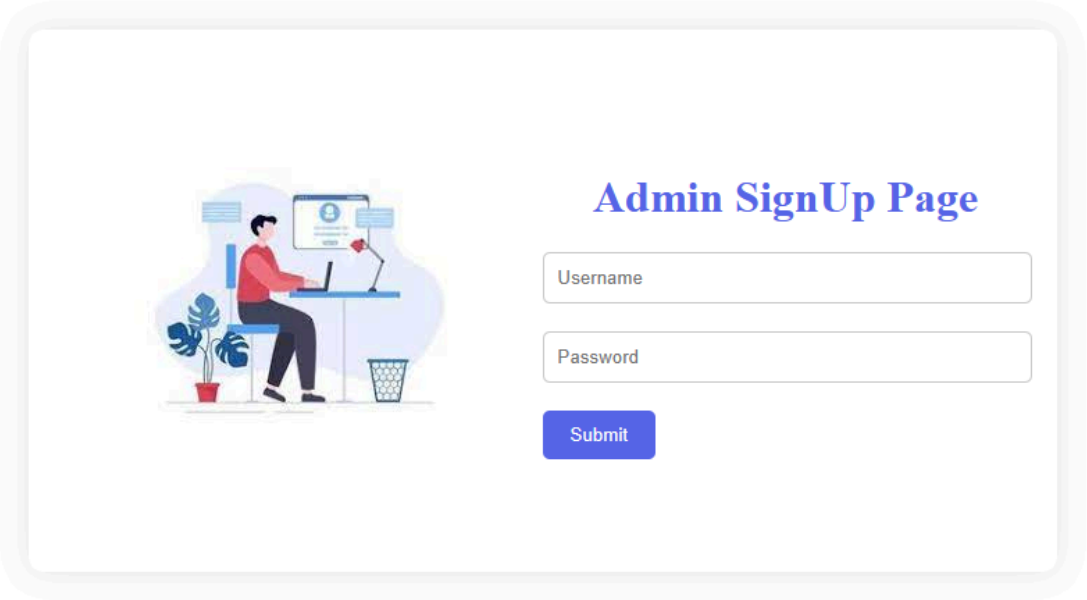
Welcome to the Page! Enter Project ID

Project ID:

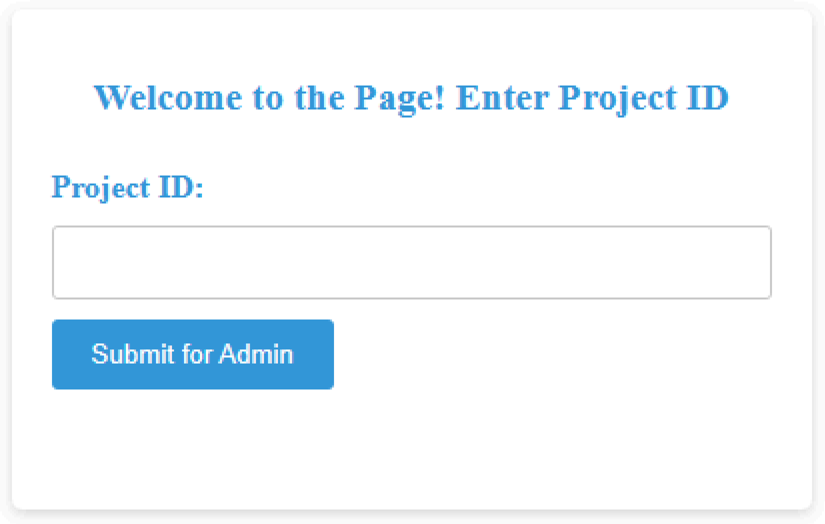


Review Marks

Team Communication

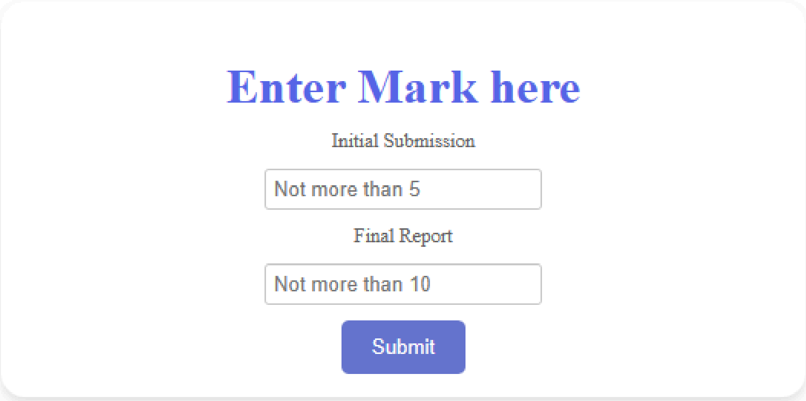


Admin SignUp Page



Welcome to the Page! Enter Project ID

Project ID:



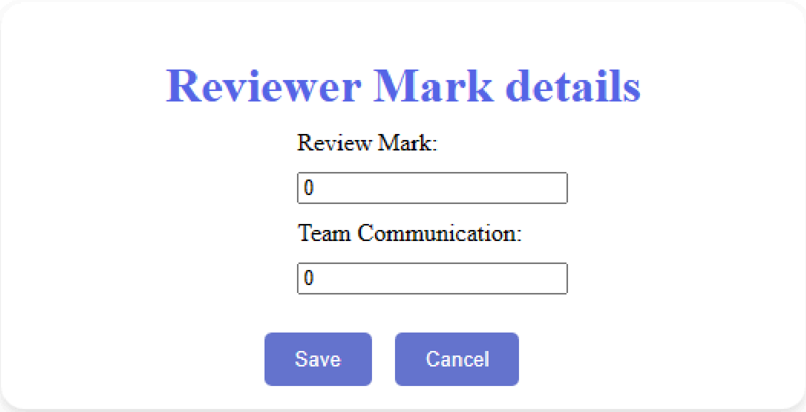
Enter Mark here

Initial Submission

Not more than 5

Final Report

Not more than 10



Reviewer Mark details

Review Mark:

0

Team Communication:

0