Faculty Kit

The faculty kit contains the evaluation strategy for the different milestones of the project and any other documents/links that may aid in the evaluation process (like sample quizzes on technologies etc)

Evaluation Strategy/Tips for the different milestones of the project

Objective

This Faculty Kit outlines the guidelines and evaluation strategies for assessing the Work Planner application. It provides a structured framework for evaluating student progress and deliverables across project milestones, ensuring quality in both design and implementation.

Requirements Specification

Evaluation Criteria:
- **Clarity**: The specification document should clearly define the objectives, scope, and functionality of the Work Planner.
- **Validity**: Ensure that assumptions and features are appropriate and within the problem domain.
- **Understanding**: Each team member must articulate the requirements and their role in shaping the system's behavior.
- **Presentation Quality**: Evaluate the structure, formatting, and presentation of the document.

Technology Familiarization

Evaluation Approach:
- **Team Presentation**: Assess the team's grasp of the following technologies through a detailed presentation:
  - Frontend: HTML, CSS, JavaScript, React.js, React Router, Lucide React, Tailwind CSS
  - Backend (BaaS): Supabase (Authentication, Database, APIs)
  - Build Tools: Vite, TypeScript
- **Quiz**: Conduct a quiz on key concepts including component-based development, routing, RESTful APIs, PostgreSQL integration, and state management.

Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Database Creation

Write-up Review:

- **Clarity and Conciseness**: Assess table structures, relationships, keys, and avoidance of redundant fields.
- **Data Storage Calculations**: Evaluate projected storage usage based on anticipated planner data volume.
- **Backup and Recovery**: Review the use of Supabase features for data backup, versioning, and recovery.

High-Level and Detailed Design

Evaluation Methods:
- **Presentation/Viva**: Review the architectural design including:
- **Key Evaluation Points**:
- - Alignment with the RS document and user workflows.
- - Use of flowcharts, wireframes, or component trees for planning.
- - Explanation of state management using React and data flow through props/hooks.

Front-end implementation

- **Demo Evaluation:**
- **User Interface**: Visual quality, responsiveness (via Tailwind CSS), and consistent styling.
- **Help Functionality**: Onboarding guides, tooltips, or contextual help components.
- **Intuitive Design**: Logical navigation using React Router and clear layout design.
- **Error Messaging**: Clear and user-friendly error prompts (e.g., failed login, validation errors).

Integrating the front-end with the database

Full Demo Evaluation:
- **Stability**: Application should operate reliably with Supabase backend.
- **Feature Demonstration**: Show planner CRUD operations, real-time updates (if used), and authentication.
- **User Operations**: Demonstrate account registration, login, task creation/editing, and deletion via Supabase APIs.

Test-plan review

Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Document Evaluation:
- **Coverage**: Test cases must address all requirements, including edge cases.
- **Execution Clarity**: Include clear setup steps and expected outcomes.
- **Error Scenario Testing**: Include invalid login attempts, API failure simulation, and input validation.

Final review

- **Final Evaluation:**
- **Demo and Project Report**: Evaluate completeness of functionality and coherence of the written report.
- **Intermediate Submissions**: Consider quality and timely delivery of milestone documents and prototypes during final assessment.