Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

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

These guidelines are intended to help faculty assess the progress, completeness, and quality of the Django-React Effort Tracking project. They are supplementary to your internal assessment methods and provide specific evaluation criteria for each milestone.

Requirements Specification

Key points to evaluate:

1. Clarity of the specification document

2. Validity of the assumptions made

3. Team-wide understanding of the requirements

4. Quality of documentation and presentation (formatting, structure, diagrams, etc.)

Technology Familiarization

Evaluation method:

- A short team presentation demonstrating understanding of Django, React, REST APIs, and PostgreSQL/SQLite.

- A follow-up quiz covering:

    o Basics of Django models/views/serializers

    o React component structure and state

    o REST API concepts

    o Database schema design

Database Creation

A concise document should describe:

1. Database schema (tables/models, field types, primary & foreign keys)

2. Clarity and normalization (avoid redundant fields)

3. Data storage estimations (approximate records, field sizes)

4. Consideration of backup/recovery plans

High-Level and Detailed Design

Evaluation can be done through:

- Design document submission or oral presentation + viva

- Review for:

    1. Full coverage of requirements

    2. Detailed pseudocode or flowcharts for major modules (login, leave approval, etc.)

    3. Discussion of alternative design approaches

    4. Error handling and notifications

    5. System robustness (e.g., behavior during DB failures)

Front-end implementation

Demo the initial UI including login and employee dashboard.

Evaluation points:

1. UI look-and-feel: Clean layout, not cluttered, visual cues (e.g., error/success messages)

2. User help: Tooltips, inline help, user-friendly instructions

3. Usability: Should be intuitive without needing extensive training

4. Validations and error messages: Should be clear, relevant, and helpful

Integrating the front-end with the database

Conduct a complete system demo with backend integration.

Faculty should look for:

1. Stable operation without crashes or hangs

2. Proper demonstration of all major features (CRUD, leave management, login)

3. Ability to create a new user and verify login functionality

4. Proper deletion of user accounts from both UI and DB

Test-plan review

Submit a test plan covering:

1. Coverage of all functionalities from the Requirements Spec

2. Clear instructions on test case execution

3. Inclusion of input validation and error-handling cases

4. Consideration of exception scenarios (e.g., server downtime)

Final review

Final evaluation based on:

• Demo of full working system

• Submission of all project artifacts (documentation, code, user manual, test results)

• Quality of final report

• Clarity and confidence in presentation

Documents/References that may aid the process of evaluation

1. **React Official Documentation**

   – https://react.dev/reference/react

2. **Django Official Docs**

   – https://docs.djangoproject.com

3 **Node.js API Docs** – https://nodejs.org/docs/latest/api