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
The CRMS Faculty Kit provides an evaluation strategy for assessing the progress and quality of the project at various milestones. It includes guidelines and tips for assessors along with useful resources and documents for evaluation.

Requirements Specification
Key Evaluation Points:

- Clarity and completeness of the functional and non-functional requirements.
- Validity and feasibility of assumptions made.
- Team-wide understanding of the requirements.
- Presentation and documentation quality (organized and professionally formatted).

Technology Familiarization
Mode: Team presentation + short quiz

Technologies to be covered:

- HTML, CSS, JavaScript (Frontend)
- Node.js with Express.js (Backend)
- PostgreSQL (Database)
- Vercel (Frontend Deployment)
- Neon DB (Cloud-hosted PostgreSQL)

Evaluation Criteria:

- Clear understanding of each technology's role.
- Ability to explain architecture and tech stack choices.
- Quiz scores (optional) on selected topics.

Database Creation
Evaluation Parameters:

- Normalized, clear schema with well-defined primary/foreign keys.
- Proper use of datatypes and constraints.
- Avoidance of redundancy.
- Inclusion of backup/recovery strategies and indexing where necessary.
- Documentation of relationships (one-to-many, many-to-many).

High-Level and Detailed Design
Evaluation Points:

- Completeness in covering all requirement specifications.
- Availability of component breakdowns and responsibilities.
- Clarity of flowcharts, wireframes, or pseudocode for core modules.
- Alternatives or scalability considerations discussed.
- Error handling and fail-scenario designs (e.g., Neon DB connection failure).

Front-end implementation

Demo Requirement: Partial working front-end without backend integration

Evaluation Criteria:

- Clean and user-friendly UI using HTML/CSS.
- Proper layout and accessibility practices.
- Responsiveness across devices.
- Intuitive navigation and meaningful content hierarchy.
- Presence of tooltips, modals, and basic validations (JS).

Integrating the front-end with the database
Evaluation Points:

- CRUD operations on criminal records functioning correctly.
- Real-time updates reflecting in Neon DB.
- Robustness: no app crashes, correct error messages on failure.
- Demonstration of:
    - Login/logout session handling.
    - New user creation and deletion.
    - Record addition, editing, searching, and deletion.
- Deployment demo on Vercel (frontend) and Neon (DB).

Test-plan review
Evaluation Criteria:

- Coverage of all functional requirements.
- Descriptions of how to execute each test case.
- Inclusion of negative test cases (e.g., SQL injection attempts, invalid input).
- Exception testing (e.g., DB disconnection, Vercel deployment issues).
- Use of tools like Postman for API testing is a plus.

Final review
Deliverables:

- Final working project (deployed).
- Project Report.
- All documentation from prior stages.

Evaluation Points:

- Completeness and functionality of the CRMS system.
- Smooth demo across modules (login, admin control, record management).
- Quality of code (structure, readability, comments).
- Project Report includes architecture, challenges, screenshots, future scope.
- Deployment links and accessibility.

Documents/References that may aid the process of evaluation

- **HTML/CSS/JS**: https://www.w3schools.com/
- **Node.js**: https://nodejs.dev/
- **PostgreSQL**: https://www.postgresql.org/docs/