



Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Faculty Kit

These evaluation guidelines are meant for the assessor to check the **progress, quality, and health** of the student's **CRMS (Crime Record Management System)** project. This document mirrors the format provided in the official faculty kit while adapting it to suit a web-based crime record management system developed using PHP and MySQL.

Evaluation Strategy/Tips for the different milestones of the project

1. Requirements Specification

- **Check whether students clearly defined:**
 - Modules for login, crime records, FIR, court cases, etc.
- **Evaluate:**
 - Clarity of the specification document.
 - Validity of assumptions (e.g., secure login, case linking).
 - Understanding of requirements by all team members.
 - Quality of formatting and visuals in the document.

2. Technology Familiarization

- **Conduct a presentation + viva or quiz covering:**
 - PHP syntax & logic
 - MySQL table design
 - HTML/CSS structure
 - Session handling in PHP
- **Check for actual code samples and conceptual clarity.**

3. Database Creation

Evaluate a short write-up + practical demo:

1. Schema design and primary/foreign key clarity.
2. Normalization and redundancy minimization.
3. Use of SQL constraints (e.g., NOT NULL, UNIQUE).
4. Whether any data-backup strategies are planned (e.g., export via phpMyAdmin).

4. High-Level and Detailed Design

Review documents and/or a presentation covering:

1. Whether all use cases are addressed in the design.
2. Flowcharts or pseudocode for login, FIR filing, etc.
3. Discussion of design alternatives (e.g., AJAX forms vs. classic POST).
4. Clear, user-friendly error messages included.
5. Robustness (e.g., failure scenarios when DB is down).

5. Front-End Implementation

Check GUI demo (HTML/CSS/JS):



Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

1. Clean, minimal layout — no UI clutter.
2. Forms are logically grouped with placeholders and tooltips.
3. Navigation is intuitive (dashboard, logout, side menu).
4. Error messages (e.g., invalid login) are clearly shown.

6. Front-End & Database Integration

Check working feature demo:

1. Forms submit data to MySQL without crashes or timeouts.
2. Test CRUD operations — Add/Edit/Delete criminals/FIRs.
3. Admin can create/delete users and see it reflect in the DB.
4. All functionalities like login/logout, search, and export work seamlessly.

7. Test Plan Review

Evaluate written test plan:

1. All RS requirements are matched with test cases.
2. Each test case includes clear instructions and expected output.
3. Test invalid inputs (e.g., wrong password, empty fields).
4. Include edge cases (e.g., SQL injection attempt, session timeout)

8. Final Review

Evaluate final report + project demo:

1. All modules are complete and operational.
2. Data flows properly from UI to DB.
3. Documentation includes screenshots and flow diagrams.
4. Includes improvements based on previous review feedback.

Additional Resources for Faculty

- PHP Quiz: https://www.w3schools.com/php/php_quiz.asp
- MySQL Docs: <https://www.mysqltutorial.org>
- HTML/CSS Reference: <https://www.w3schools.com>