



Hackathon Documentation

Hackathon Project Title: *Learning Management System (LMS)*

Duration: 24 Hours

Objective: Build and deploy a basic Learning Management System with increasing levels of complexity. Teams will push their final code to GitHub and host their projects for evaluation.

Challenge Overview

Participants will work in teams to design and implement a **Learning Management System (LMS)** with features for **students** and **teachers**. The project will be judged on **functionality**, **design/UX**, and **code quality**. Performance will be ranked as **Bronze, Silver, Gold, or Platinum** depending on how many user stories are completed.

User Stories

Each user story is designed to gradually increase in difficulty — starting simple and progressing to advanced features.

Bronze Level (Basic Functionality)

User Story 1 – User Registration & Login

- As a **new user**, I want to register with my name, email, password, and role (Student/Teacher) so that I can create an account.
- As a **user**, I want to log in using my email and password so that I can access the system.
- As a **system**, I want to store user data securely in a database.

User Story 2 – Course Management (Teacher role)

- As a **teacher**, I want to create a course by providing a title, description, and duration so that students can see available courses.
 - As a **student**, I want to view a list of available courses on the homepage so that I can choose one to enroll in.
-

Silver Level (Intermediate Functionality)

User Story 3 – Course Enrollment

- As a **student**, I want to enroll in a course so that I can participate in it.
 - As a **student**, I want to see a list of courses I am enrolled in.
 - As a **teacher**, I want to see a list of students enrolled in my course so that I can track participation.
-

Gold Level (Advanced Functionality)

User Story 4 – Assignment Submission

- As a **teacher**, I want to create assignments for a course so that students can complete them.
 - As a **student**, I want to submit my assignments to the course so that the teacher can evaluate them.
 - As a **teacher**, I want to see all submissions for a course so that I can review and grade them.
-

Platinum Level (Expert Functionality)

User Story 5 – Grading System

- As a **teacher**, I want to grade student assignments so that students receive feedback.
 - As a **student**, I want to view my grades so that I can track my performance.
 - As a **system**, I want to calculate and display overall grades for each student.
-

User Story 6 – Optional Advanced Features

- As a **student/teacher**, I want a course discussion forum so that we can share knowledge and questions.
 - As a **teacher**, I want to upload course materials (PDF, PPT, etc.) so that students can access them.
 - As a **user**, I want to receive notifications for new assignments, grades, and updates so that I don't miss important information.
-

Ranking System

- **Bronze:** Completes User Story 1 & 2
 - **Silver:** Completes User Story 1 to 3
 - **Gold:** Completes User Story 1 to 4
 - **Platinum:** Completes User Story 1 to 5 (+ optional features for bonus points)
-

Technical Requirements

- **Frontend:** HTML, CSS, JavaScript (React.js/Angular encouraged)
 - **Backend:** Node.js with Express.js
 - **Database:** MySQL or MongoDB
 - **Deployment:** Must be hosted (e.g., Vercel, Netlify, Render, Railway, or AWS)
 - **Version Control:** Code must be pushed to the team's GitHub repository.
-

Submission Requirements

1. Working hosted link of the project.
 2. GitHub repository link with clear commit history.
 3. A short README file in GitHub including:
 - Team name & members
 - Tech stack used
 - Features implemented
 - Hosting link
-

Judging Criteria

1. **Functionality** – Required features working correctly.
2. **Design & User Experience** – Clean, user-friendly interface.
3. **Code Quality** – Organized, readable, maintainable code.
4. **Deployment** – Successfully hosted and accessible project.