



North South University

Department of Electrical & Computer Engineering (ECE)

Project Plan, Design, and Scheduling

iNSUre - A Campus Event Management System

Student IDs

Md. Abul Bashar Nirob – 2022198042

Istiak Ahasan – 2010282042

Submitted To: MdSH

iNSUre: A Campus Event Management System

Prepared by: Md Abul Bashar Nirob

Date: 13 August 2025

Client/Customer

This is an academic project developed for the course CSE-327, under the supervision of Prof. Dr. Md. Sazzad Hossain. North South University (NSU) – Targeting students, faculty, and event organizers within the campus.

Team Members

1. Md. Abul Bashar Nirob – 2022198042
2. Istiak Ahasan – 2010282042

Outline Description

The "iNSUre" Campus Event Management System is a web-based platform designed to centralize, streamline the management of university events. It provides a comprehensive solution for event organizers, participants, administrators. It features a role-based access control system with three distinct user types:

- ✓ **Administrator:** Has full control over the system, including managing all user accounts and events.
- ✓ **Organizer:** Can create, manage, and delete their own events, view participant lists, see feedback their events.
- ✓ **User (Student/Faculty):** Can browse all public events, register for them, submit feedback after attending.

The core objective is to replace disparate and manual event management methods with an efficient, single, and integrated digital platform.

Key features include:

- ✓ Event creation, editing, and deletion.
- ✓ User registration and management.
- ✓ Participant tracking and management.
- ✓ Feedback collection and display.
- ✓ Automated email notifications using PHPMailer.
- ✓ Mobile-friendly responsive design.

The system leverages PHP, MySQL, HTML, CSS, and JavaScript, ensuring reliability, scalability, and maintainability.

Current Status:

This report is structured to provide a comprehensive overview of the CEMS project. Section 1 introduces the project's purpose and scope. Section 2 details the project management plan, including the lifecycle model, risk analysis, and scheduling. Section 3 outlines the requirement specifications, including stakeholders and use cases. Section 4 describes the system's architecture. Section 5 delves into the detailed design of the system, including GUI, static, and dynamic models. Finally, Section 6 presents the test plan, detailing the testing strategy and cases.

Previous Work:

- Complete backend and frontend development completed.
- Role-based access control implemented (Admin, Organizer, User).
- CRUD functionalities for events and users operational.
- PHPMailer successfully integrated for email notifications.
- UI designed with responsive and modern styling.
- Basic JavaScript form validation implemented.
- Currently in testing and debugging phase.

Plan

Architectural Design: The system is built using a 3-Tier Architecture, which separates the application into three logical layers:

- 1. Presentation Tier (Front-End):** The user interface built with HTML, CSS, and JavaScript. This is what the user interacts with in their browser.
- 2. Application/Logic Tier (Back-End):** The core of the application, built with PHP. This tier handles all business logic, processes user requests, manages sessions, and communicates with the database.
- 3. Data Tier (Database):** A MySQL database that stores all persistent data, including user credentials, event details, registrations, and feedback.

Major Stages & Task Assignments

Stage 1: Planning & Requirement Analysis

- Define project scope and objectives.
- Identify stakeholders and user roles.
- Create use cases and define functional/non-functional requirements.
- Assigned to: Both members.

Stage 2: Design & Architecture

- ✓ Design the database schema.
- ✓ Create wireframes and mockups for the GUI.
- ✓ Finalize the 3-tier architecture and technology stack.
- ✓ Assigned to: Both members (Database: Istiak, GUI: Bashar).

Stage 3: Development

The project was developed by a team of two members, with responsibilities divided to ensure efficient workflow and comprehensive coverage of the project's requirements:

- **Backend:** Set up the database and develop PHP scripts for user authentication, session management, and CRUD operations for users and events. (Assigned to: Istiak)
- **Frontend:** Develop the user interface using HTML/CSS for all pages, including dashboards, forms, and event listings. (Assigned to: Bashar)
- **Integration:** Connect the frontend forms and actions to the backend PHP scripts. Implement dynamic data display. (Assigned to: Both members)
- **Advanced Features:** Integrate PHPMailer for email notifications. (Assigned to: Both members)

Both team members collaborated on system design, testing, and documentation.

Stage 4: Testing & Deployment

- ❖ Conduct unit testing and system-level testing.
- ❖ Perform user acceptance testing to ensure all requirements are met.
- ❖ Debug and fix any identified issues.
- ❖ Assigned to: Both members.

Stage 5: Documentation & Finalization

- ✓ Prepare the final project report and presentation.
- ✓ Assigned to: Both members.

Technical Environment

1. **Backend:** PHP, MySQL
2. **Frontend:** HTML, CSS, JavaScript
3. **Web Server:** Apache (run locally via XAMPP)
4. **Libraries:** PHPMailer
5. **Version Control:** Git / GitHub
6. **Development Tools:** Visual Studio Code

Project Scheduling

Deliverable	Planned Date	Actual Date
Project Proposal	Week 1	Week 2
Requirement Specification	Week 4	Week 4
System Architecture & Design	Week 6	Week 7
Initial Prototype (Login & Dashboards)	Week 8	Week 8
Mid-term Presentation	Week 9	Week 9
Feature Complete System	Week 12	Week 13
Final Testing & Debugging	Week 13	Week 14
Final Report & Presentation & Submission	Week 14	Week 14

Any Other Relevant Information

The project successfully meets its primary goal of creating a centralized event management platform. The choice of a 3-Tier architecture provides a solid foundation for future scalability, such as adding new user roles or more complex event management features. The code is modular, with separate folders for different user roles (admin, organizer, user) and shared components, which makes it easier to maintain and update.

Conclusion

We would like to express our sincere gratitude to our course instructor MdSH Sir for CSE-327 for their guidance, support, and valuable feedback throughout the duration of this project. Their insights were instrumental in shaping the project and helping us navigate the challenges of software development.