.

Project Planning and Management

1.1 Project Proposal

Overview

Imagine a library where everything works like magic. Instead of endless paper logs and manual processes, this Library Management System will make everything smooth and hassle-free. Users will be able to search for books instantly, check them out, and return them without any hassle. The system will even handle penalties for late returns, making sure everything stays organized. With a clean and simple interface, both librarians and members will find it easy to use and efficient. In this chapter, we provide an overview of our proposed project.

Project Objectives

We will build a system that simplifies and automates the process of managing library resources and user interactions. Here is a list of our project objectives:

1- **Efficient Book Management**

- Create an easy-to-use system for adding, updating, deleting, and searching books.
- Ensure real-time availability status for books.

2- **Seamless Member Management**

- Provide functionalities for registering members, updating their details, and tracking their borrowing history.
- Ensure data privacy and security for member details.

3- **Transaction Handling**

- Implement features for checking books in and out smoothly.

-
- Automate penalty calculations for overdue books and notify members accordingly.

4- **Search and Discovery**

- Offer a powerful search feature to help users find available books by title, author, or category.

5- **Reporting and Insights**

- Generate reports for administrators to monitor the status of books and member activities.
- Provide insights to enhance the user experience.

6- **User-Friendly Interface**

- Design an intuitive dashboard that allows users to interact with the system easily.
- Make the interface responsive and accessible across devices.

7- **Scalability and Reliability**

- Build the system to handle multiple users and transactions without performance issues.
- Ensure data integrity and reliability of the system.

8- **Security and Authentication**

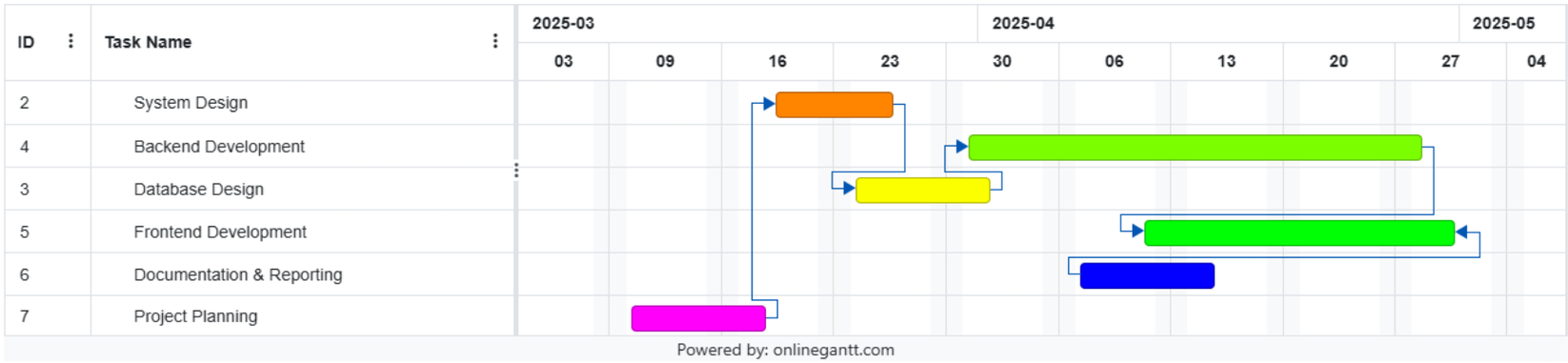
- Implement secure authentication mechanisms for both administrators and members.
- Protect user data and transactions against unauthorized access.

Project Scope

The Library Management System is a digital solution designed to enhance the management of libraries by replacing traditional, paper-based systems with an automated and streamlined approach. The system will allow users to search for available books, process checkouts and returns, and calculate penalties for late returns. It will also provide member management features, enabling administrators to add, update, and monitor members' activities. The system will be accessible through a web-based interface using HTML, CSS, and JavaScript for frontend functionalities, while backend processes will be handled using ASP.NET Core with SQL Server for data storage. The system will primarily target libraries of various sizes, from small community libraries to larger institutional libraries, providing a scalable solution for managing resources effectively

1.2 Project Plan

Timeline



Total duration: 8 weeks

Milestones

- **Project Planning Completed:** March 9, 2025
- **System Design Completed:** March 16, 2025
- **Database Design Completed:** March 23, 2025
- **Backend Development Completed:** April 20, 2025
- **Frontend Development Completed:** April 27, 2025
- **Documentation & Reporting Completed:** May 4, 2025

Deliverables

- Project Proposal Document
- System Architecture Design
- Database Schema
- Functional Backend System
- User Interface & Frontend System
- Complete Documentation (User Guide, Technical Documentation, Reports)

Resource Allocation

- **Team Members:** 5 People
- **Tools & Technologies:** ASP.NET Core, MVC, Entity Framework, SQL Server, HTML, CSS, JavaScript
- **Hardware Resources:** Laptops/PCs, Server for hosting

1.3 Task Assignment & Roles

- **Backend Developers**:
  - Abdelrahman Mohamed:
    Responsible for implementing backend APIs using ASP.NET Core, managing data storage in SQL Server, and ensuring secure communication between frontend and backend.
  - Mohamed Helmy:
    Handles implementing database models, building API endpoints, managing authentication, and performing database operations.
  -
- **Frontend Developers:**
  - Mazen Elsady:
    Focuses on developing reusable components, enhancing accessibility, optimizing frontend performance, and collaborating with backend developers.
  - Shahd Ashraf:
    Creates user interfaces using HTML, CSS, and JavaScript, ensuring responsive design and integrating frontend with backend APIs.
  - Mariam Zenhom:
    Designs UI components, implements user interactions, styles pages for better UX, and integrates frontend with backend APIs.

1.4 Risk Assessment & Mitigation Plan

1. **Technical Challenges:** Research, task breakdown, expert consultation.
2. **Project Delays:** Progress monitoring, realistic deadlines, buffer time.
3. **Communication Issues:** Regular meetings, effective tools, promote collaboration.
4. **Data Security Risks:** Secure authentication, input validation, safe data storage.
5. **Integration Issues:** Thorough testing, testing plans, incremental integration.

1.5 KPIs

 Response **Time:**
Measure the time taken for the system to respond to user requests. Target: Less than 1 second for standard requests.

 System **Uptime:**
Monitor system availability and ensure it remains operational. Target: 99.9% uptime.

 User **Adoption Rate:**
Track the number of users actively using the system compared to the total number of target users. Target: 80% adoption rate within the first month of launch.

 Bug **Resolution Rate:**
Measure the percentage of reported bugs fixed within a specific time frame. Target: 95% of bugs fixed within 48 hours.

 Data **Accuracy:**
Ensure data is accurately stored, retrieved, and processed. Target: 99.5% data accuracy.

 Security **Breaches:**
Track the number of security incidents or breaches. Target: Zero breaches.