# Library Management System
# Version 1.0

# Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 02/14/25 | 1.0 | Initial template | Abhiroop Goel |

# Table of Contents



# Database Development Plan

## 1. Introduction

The *Software Development Plan* covers the entire document and describes the purpose of the project. This gives the manager and team members an overview of what they are doing and also informs anyone who needs to look at the development process for this project.

### 1.1 Purpose

The purpose of the Software Development Plan is to consolidate all necessary information for effectively managing the project. It describes the approach taken for the software's development and acts as the primary reference for project planning, scheduling, and resource allocation.

Project managers and team members will use this document to:

- Plan the project timeline and allocate resources.
- Track development progress against the schedule.

- Define roles and responsibilities for each team member.
- Ensure that all project milestones and deliverables are met within the assigned deadlines.

### 1.2 Scope

This Software Development Plan outlines a structured approach to designing and implementing the Library Management System using relational database principles. The system will be developed with SQL and other database management tools to ensure efficient storage, retrieval, and management of library records. The project will follow an iterative process, encompassing conceptual, logical, and physical database design, the development of core functionalities such as user management, book tracking, borrowing rules, and reporting, and the implementation of constraints, triggers, and stored procedures to maintain data integrity. Additionally, rigorous testing and deployment will be conducted to ensure the system meets all functional and non-functional requirements.

### 1.3 Definitions, Acronyms, and Abbreviations

Any uncommon definitions or abbreviations used in any of the project documents will be added here.

EECS: Electrical Engineering and Computer Science (at the University of Kansas) -

EECS 477: Introduction to Database Systems

### 1.4 References

Reference documents will be added to the table as they become available.

| Title | Date | Source | Link |
| --- | --- | --- | --- |
| EECS447: Project Description | Feb 14, 2024 | EECS 447 Software Engineering I Lecture on Canvas | https://canvas.ku.edu/courses/152575/files/folder/Project?preview=12791194 |

### 1.5 Overview

This *Software Development Plan* contains the following information:

Project Overview — Outlines the purpose, scope, and objectives of the project, detailing the system's functionalities and expected deliverables.

Project Organization — Defines the team structure, assigning responsibilities to each team member and providing their contact information. This section ensures clarity on individual roles and contributions.

Management Process — Establishes the project timeline, including major iterations, milestones, and deadlines. It also outlines the methods for tracking progress, ensuring that the project stays on schedule and meets all requirements.

## 2. Project Overview

### 2.1 Project Purpose, Scope, and Objectives

The goal of this project is to develop a Library Management System that facilitates efficient management of a library's collection, tracks borrowing activities, and enforces lending rules. Through this project, students will gain hands-on experience in database design, SQL implementation, and data integrity management, while also enhancing their understanding of collaborative software development, project documentation, and long-term system planning. The project will produce several key deliverables, including a fully functional relational database, a conceptual, logical, and physical database design, and a

requirements document outlining system functionalities and constraints. Additionally, a design document will detail the database structure and relationships, supported by SQL scripts for database creation, constraints, triggers, and stored procedures. A test case document will ensure system integrity, and the project will conclude with a final report and demonstration showcasing its features and functionality. These deliverables may evolve based on project progress and requirements.

## 2.2 Assumptions and Constraints

The successful execution of this project depends on several factors. The constraints include the timeline and structure defined by EECS447: Database Systems, the availability of team members for meetings, and adherence to project milestones. The project will be collaboratively developed using GitHub, ensuring team members have 24/7 access to the codebase and documentation, provided they have an internet connection. Changes to project scope, database design, or feature set may be subject to discussions and adjustments based on instructor guidance and project complexity.

## 2.3 Project Deliverables

| Deliverable | Target Delivery Date |
| --- | --- |
| Project Plan | Feb 19, 2025 |
| Requirements Document | |
| Conceptual Design | |
| Logical Design | |
| Physical Design and Data Population | |
| Project Demonstration | |

## 2.4 Evolution of the Software Development Plan

| Version Number | Description |
| --- | --- |
| | |
| | |

| | |
| --- | --- |
| | |

# 3. Project Organization

## 3.1 Organizational Structure

The project team for the Library Management System is structured into key roles, including Project Leader, Assistant Project Leader, Team Administrator, Assistant Team Administrator, Technical Leader, and Quality Assurance Engineer. While each role has specific responsibilities—Project Leaders focusing on overall project management, Team Administrators handling coordination and documentation, and Technical and Quality Assurance Leaders overseeing database development and system integrity—all team members will collaborate across various tasks as needed. A detailed breakdown of each role's responsibilities and specific assignments is provided in Section 3.3.

## 3.2 External Interfaces

All interactions with external stakeholders will be managed primarily by the Project Leader. The primary external contacts for this project are Professor Hossein Saiedian, who will provide guidance on project requirements, deadlines, and expectations. They will also serve as a resource for resolving unforeseen challenges and will be responsible for evaluating the final project deliverables and documentation.

## 3.3 Roles and Responsibilities

| Person | Unified Process for EDUcation Role | Phone Number | Emails | Expertise |
| --- | --- | --- | --- | --- |
| | Project Leader | | | |
| | Assistant Project Leader | | | |
| | AssistantTeam Administrator | | | |
| Roop Goel | Team Administrator | 785-979-9540 | abhiroopgoel@gmail.com | C, java, python, SQL |
| Spencer Sliffe | Technical Leader | 8162164610 | spenceronw@ku.edu spencer@studious.pro | Full Stack Dev, Engineer & Architect |
| | Quality Assurance Engineer | | | |

| Role | Responsibility |
| --- | --- |
| Project Leader | Compile and submit all deliverables, project planning, management, interaction with external entities, and meeting logs. |

| | |
| --- | --- |
| Assistant Project Leader | Project planning, management, additional correspondence with PL, and document management. |
| Team Administrator | Meeting planning, facilitation of inter-team discussion, help to provide resources where needed |
| Assistant Team Administrator | Provide general administrative support to team members, such as preparing documents and presentations |
| Technical Leader | Primary code maintenance, and GitHub management. Help with code for other team members, test case management |
| Quality Assurance Engineer | Test cases, code management, and requirements tracking. |
| All Roles | Code writing, documentation, testing, requirement management, issue tracking |

Anyone on the project can perform any role activities, and responsibilities may be added or changed at any time.

# 4. Management Process

## 4.1 Project Plan

Resources for the project will be added as they become available. An outline of the project implementation is provided in 4.1.1. Each update to the project implementation will be recorded in 4.1.2 as they are released, along with a brief description of the update. A schedule of important dates for major milestones and project deadlines is available under 4.1.3 and will be updated as additional information is released.

### *4.1.1 Iteration Objectives*

The following is a table providing the major project iterations, with descriptions defining the major objectives that should be done by that iteration. Iterations are not limited to those listed here.

| Iteration Version/Name | Description/Objectives |
| --- | --- |
| | |

#### *4.1.2 Releases*

| Software Release Version | Description |
| --- | --- |
| V 1.0 | Document outlines. |
| | |

#### *4.1.3 Project Schedule*

A rough estimation of the project timeline is provided below. These dates may be updated as more information becomes available.

### 4.2 Project Monitoring and Control

The Library Management System will be monitored and controlled by tracking progress against predefined milestones and objectives. Our approach includes regular monitoring of deliverables, assessing risks, ensuring quality control, and maintaining comprehensive project documentation. Weekly meetings will play a crucial role in updating project artifacts, addressing potential issues, and ensuring that the project remains aligned with its timeline and requirements.

### 4.3 Quality Control

Quality control will be integrated throughout the development process. It will include code reviews, validation of requirements, and rigorous testing to ensure correctness and performance. All code modifications will be documented through a change log, and any defects identified during reviews will be logged for resolution. The Quality Assurance Engineer will lead this effort, ensuring that the system meets the expected functionality, performance, and reliability standards.

### 4.4 Risk Management

Risk management will involve identifying potential challenges such as technical constraints, resource availability, and dependencies on external tools or libraries. Each risk will be analyzed for its probability and impact, with high-priority risks addressed proactively to minimize disruptions. Risks will be continuously reviewed during weekly meetings, where new risks will be assessed, and mitigation strategies will be updated.

### 4.5 Configuration Management

Configuration management will be handled using GitHub for version control and a ticketing system for tracking change requests, risks, and deliverables. Source code and project documentation will be systematically managed with clearly defined branches and tags representing different development stages. Retention policies will ensure regular backups of the repository and critical documents, including both incremental and full backups. In case of system failures or unforeseen issues, these backups will allow for fast recovery and minimal data loss.

## 5. Annexes

.

The project will follow the UPEDU process.