LIBRARY MANAGEMENT SYSTEM

TEAM MEMBERS:

UMANG SAXENA	PL0122181
KAMANA TRIVEDI	PL0122189
SHAILJA THAKUR	PL0122202
MADHAVI KENGUVA	PL0122205
CHAKALI PAVAN KUMAR	PL0122201

CONTENTS

ABSTRACT

Library management system is an application which is very easy to access for the admin and highly secured application which only admin can access. This application's hardware and software are easily available for any institution required.

It is used by Admin to manage the library using a computerized system where he/she can record various transactions like issue of books, return of books, addition of new books, addition of new students etc. The library management system software makes the library a smart one by organizing the books systematically by author, title and subject.

An advanced organized library is an integral part of any educational institution. The library management system software of digital libraries allows the admin to login, search, select, issue and return books by student's request. The software calculates the fine due for non-return books. The members are intimated of the fines by the system.

OBJECTIVE

- To develop an online system to help the users(librarian's).
- To manage the distribution and collection books.
- To track the records and to modify.

Functionalities provided by Library Management System are as follows:

- ✓ Admin can issue and collect books.
- ✓ New admin can be registered.
- ✓ Admin can add new students and books.
- ✓ Admin can delete existing students and books.
- ✓ Admin can modify existing data of users and books.
- ✓ Our application is secure and safe as the user details are encrypted in the data base.

INTRODUCTION

The project titled Library Management System is Library Management Software for monitoring and controlling the transactions in a library. The project "Library Management System" is developed by using java, spring boot which mainly focuses on basic operations in a library like adding new students, new books, and updating, deleting the information, searching books and members and facility to issue and return books. these transactions data stored in database.

It is designed to help users to maintain and organize library. Our software is easy to use for everyone. It has an attractive user interface and it has strong searching, insertion and reporting capabilities

SCOPE

Our project aims to safe and easy handling of library. i.e. we have made a computerized process to store data and distribution.

1. It satisfies the admin(librarian).
2. It is easy and safe to store data.
3. It is easy to operator.
4. Have a good user interface.
5. It saves time and function faster.
6. It helps the Librarian to handle the library.

We have tried to develop safe and secure software with above mentioned specifications.

- Scope of change: barcoding reading.

SIGNIFICATIONS

- Easy to update information.
- Work becomes speedy.
- Access of any information individually.
- Decrease the load of the person involve in existing manual system.
- Well-designed reports.
- Easy & fast retrieval of information.
- Accuracy in work.
- It contains better storage capacity.
- Robust database back-end.
- Creating and changing data at ease.

MODULES

Our system has one main modulo admin module.

Along this we have some more modules mentioned below.

LOGIN MODULE:

Only admin can log in to the system as the end-user of the system on the behalf of the user. The user will get only those privileges that are given to the user for which one has registered.

MENU MODULE:

After successfully login, user(admin) enters to menu. The user can add books and students, issue and return book, can retrieve information of books and students.

SPECIFIC MODULE:

In this module admin can check and update all the information like all transactions, availability of books, book id's, search of book and student, delete book, fines.

USER INTERFACE DESIGN

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

The following steps are various guidelines for User Interface Design:

- The system user should always be aware of what to do next.
- The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.
- Use display attributes sparingly.
- Default values for fields and answers to be entered by the user should be specified.
- A user should not be allowed to proceed without correcting an error.
- The system user should never get an operating system message or fatal error.

TECHNOLOGIES USED

FRONT END:

HTML:

HTML (Hypertext Markup Language) is the code that is used to structure a web page and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.

CSS:

Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML.

JAVASCRIPT:

JavaScript (JS) is a scripting language, primarily used on the Web. It is used to enhance HTML pages and is commonly found embedded in HTML code. JavaScript is an interpreted language. Thus, it doesn't need to be compiled. JavaScript renders web pages in an interactive and dynamic fashion. This allowing the pages to react to events, exhibit special effects, accept variable text, validate data, create cookies, detect a user’s browser, etc.

THYME LEAF:

Thyme leaf is a Java XML/XHTML/HTML5 template engine that can work both in web and non-web environments. It is better suited for serving XHTML/HTML5 at the view layer of MVC-based web applications, but it can process any XML file even in offline environments. It provides full Spring Framework integration.

BACK END:

JAVA:

Java is a general-purpose, class-based, object-oriented programming language designed for having lesser implementation dependencies. It is a computing platform for application development. Java is fast, secure, and reliable, therefore. It is widely used for developing Java applications in laptops, data centers, game consoles, scientific supercomputers, cell phones, etc.

SPRING BOOT:

Spring Boot is an open-source micro framework maintained by a company called Pivotal. It provides Java developers with a platform to get started with an auto configurable production-grade Spring application. With it, developers can get started quickly without losing time on preparing and configuring their Spring application.

SPRING DATA JPA:

JPA is a Java specification that is used to access, manage, and persist data between Java object and relational database. It is a standard approach for ORM.

SPRING SECURITY:

Spring Security is a powerful and highly customizable authentication and access-control framework. It is the de-facto standard for securing Spring-based applications. Spring Security is a framework that focuses on providing both authentication and authorization to Java applications.

DATABASE:

ORACLE SQL:

SQL (pronounced sequel) is the **set-based, high-level declarative computer language** with which all programs and users access data in an Oracle database. Although some Oracle tools and applications mask SQL use, all database tasks are performed using SQL.

PROJECT LIFE CYCLE

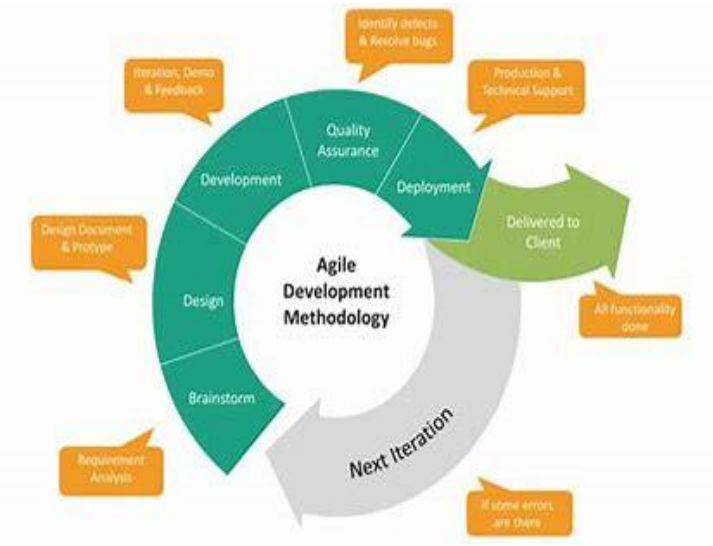
AGILE SOFTWARE DEVELOPMENT METHODOLOGY:

The Agile software development methodology is one among the only and effective processes to show a vision for a business need into software solutions. Agile may be a term want to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to vary.

The agile software development emphasizes on four core values.

- Individual and team interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

PHASES OF AGILE METHODOLOGY:



Agile Methodology Phases

Phase-1: Requirement Analysis: - In this phase, we gather data and analyses how library management system works. Also collected requirements after reviewing earlier papers & websites.

Phase-2: Design: - On the basis of gathered information we designed and build a model.

Phase-3: Development: - Deliver the working software based on iteration, requirements or feedback.

Phase-4: Quality Assurance: - This is a testing phase where we test our model.

Phase-5: Deployment: - In this phase we deploy our final release of the iteration into production.

Phase-6: Feedback: - Accept the user feedback and work it into the requirements.

ADD BOOK:

Brief Description	**Add new book.**
Basic Flow	This use case to describe how an admin can add new book to the system. 1. The admin should add book to the system before keeping book in library. 2. After the successful adding, shows a popup. 3. The following information is required during adding the book. • Book Title. • Author name. • No. of copies. • Name of the category. • Book price.
Alternate Flow	1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message.
Validation	1. Book title, Author, Book category and Minimum 3 characters and max 30 characters.
Pre-Conditions	Admin should have network access and Browser with latest updates.
Post-Conditions	Success popup should be shown.

ADD STUDENT:

Brief Description	**Add Student.**
Basic Flow	This use case describes how an admin can add new member to system. 1. Admin have to login. 2. Select add student bar. 3. Enter details and add new student. 4. The following information is required during adding the member. • Name. • Branch. • Year. • Division. • Roll number.
Alternate Flow	1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message.
Validation	1. Name, roll number, branch, year and division are required and minimum 3 characters and max 30 characters. 2. Provided details should valid.
Pre-Conditions	Admin should have network access and Browser with latest updates.
Post-Conditions	Success popup should be shown.

ISSUE BOOK:

Brief Description	**Issue book.**
Basic Flow	This use case describes how a admin issue book to member. 1. Admin have to login. 2. Select the issue bar. 3. Enter Admission number and Name of the member. 4. Enter following details. • Book Id. • Registration number.
Alternate Flow	1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message.
Validation	1. Book id, registration number are required and minimum 3 characters and max 30 characters. 2. Provided details should be valid.
Pre-Conditions	Admin should have network access and Browser with latest updates.
Post-Conditions	Success popups should be shown.

RETURN BOOK:

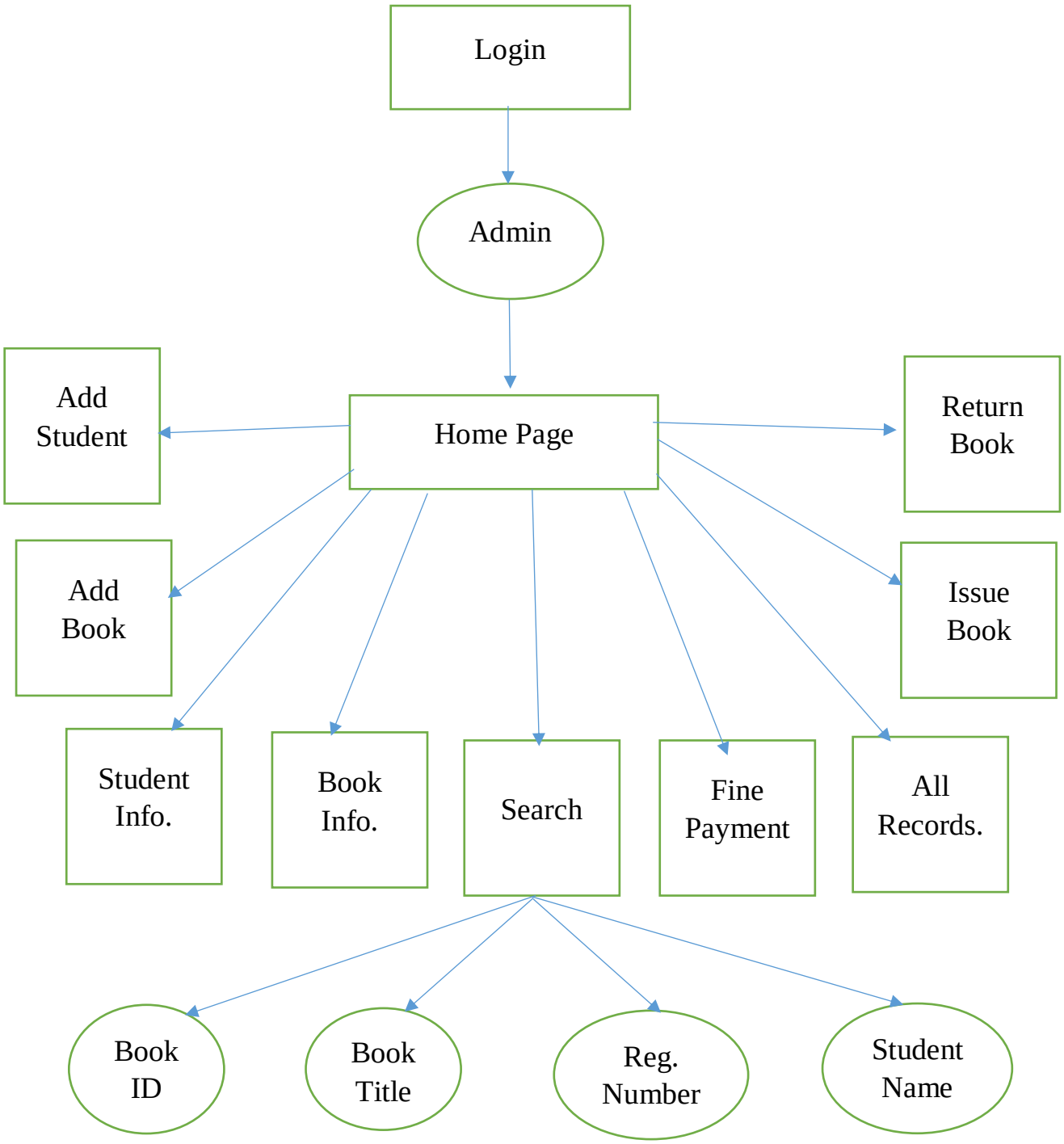
Brief Description	**Return book.**
Basic Flow	This use case describes how a admin to collect book from member. 1. Admin have to login. 2. Select the issue bar. 3. Enter Admission number and Name of the member. 4. Enter following details. • Book Id. • Book name. • Registration id. • Student name.
Alternate Flow	1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message.
Validation	1. Book Id is required and minimum 3 characters and max 30 characters. 2. Provided details should be valid.
Pre-Conditions	Admin should have network access and Browser with latest updates.
Post-Conditions	Success popups should be shown.

SEARCH:

Brief Description	**Search book.**
Basic Flow	This use case describes how a admin search for member. 1. Admin have to login. 2. Select the book in search bar. 4. Book id, • Book title, • Registered no. • Student name.
Alternative flow	1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message.
Validation	1. Student name, book title, book id, registered number are required and minimum 3 characters and max 30 characters. 2. Provided details should be valid.
Pre-Conditions	Admin should have network access and Browser with latest updates.
Post-Conditions	Success popups should be shown.

DATA FLOW DIAGRAM

A data flow diagram should be the first mechanism used by framework analyst to model system fundamental. These fundamentals are the system operations; the information used by this operations and external organization that interact with the framework and the data flows in the system



USE CASE DIAGRAM

A use case depicts a series of steps that give something significant value to an actor and is drawn as a horizontal ellipse. A use case diagram apprehends the functional features of a system with processes implement in the system. While discussing the functionality and procedures of the framework, you discover important characteristics of the framework that you represent in the use case diagram