REQUIREMENTS SPECIFICATION DOCUMENT FOR HIGH SCHOOL BOOK TRACKING APPLICATION

TEAM 13

Sandeep Inampudi	A20379958
Poojitha Vagale Duggappa	A20370777
Vaibhav Hongal	A20378220
Divija Sai Konanki	A20377662

TABLE OF CONTENTS

1. Introduction

The main objective of this document is to illustrate the requirements for the project High School Book Tracking system.

1.1 Goals and Objectives

The overall goal of this project is to develop a Web-based application for bookkeeping in a high school library. It is used for controlling and monitoring transactions in the library.

This document describes all the data, functional and behavioral requirements of the system. This project is developed in Java and mainly focuses on basic operations in library like adding new members (students and teachers), adding new books, searching for books and facilitate students and teachers to access the library books and use them on a rental basis for a specified amount of time.

It also does a reporting job for producing the following reports.

1. List of all the students with books checked out showing due date and date checked out
2. List of all authorized users and their access level
3. List of all books in the system with search capabilities
4. List of all books checked out by class or book name/number
5. List of students and parents' info with amount due

This report generation functionality of the system helps to get a good idea of which books are borrowed by the members and makes users generate the reports' hard copy.

1.2 Scope of Project

The High School Book Tracking system basically upgrades the manual library system into an internet-based application so that the users can know the details of their accounts, availability of books, maximum borrowing capacity etc. This system can be used in any existing library or new library to manage the books and transactions. The project can easily be implemented for various situations. We can add new features as and when we require. The language used for development is Java which make the application platform independent and can be used on any system.

The application will be accessible over internet on any computer and provides the following capabilities.

1. The system provides login facility to the users [students/parents/teachers/admin].
2. The system provides the library staff with the ability to manage the library user accounts including add, edit or remove accounts.
3. The system lets the library staff to manage book inventory including add, edit or remove book information.
4. The system provides the members with the option to check their account and/or change their options like password of the account whenever needed.

5. The system lets the library staff to check which all members have blocked the books and whether they can borrow any more books or not. The application will track books, classes and students who check out books.
6. The system provides its registered users with an access to the system before they can use it, based on their level (students/teachers).
7. The application will generate reports for administrative purposes.
8. The application will provide a search function on books based on their ISBNs, school generated ids, subjects, author names etc.

The success criteria for the system is based in the level up to which the features described in this document are implemented in the system.

1.3 Software context

The application we are going to design will be web based and hence the clients shall use a browser to access the application. The application server (mostly tomcat) will handle the total application. The application will communicate to the database (SQL Server) to retrieve data about students and teachers and Google Books API as well to retrieve information about the books.

The application helps libraries of any size to track its books efficiently. It shall allow teachers to check out books to students, as per their registered courses. It shall also let the students track their due dates, and the late fee incurred if any.

1.4 Major constraints

This system is web-based and hence there will be a need to provide a PC hardware connected to the internet. It should be easy to use, providing help instructions and appropriate error messages for invalid operations. System should be reliable and should not have any unscheduled down time during high school library operational hours.

2. Usage scenario

2.1 User profiles

The system provides different types of services based on the types of users such as admin, students, parents and teachers.

The admin features are as below:

1. Issue the book to the students and teachers
2. Ability to view the different categories of the books and keep track of number of books available in each category
3. Add/update/delete books information
4. Generate and view different reports such as number of books checked out, existing books, due dates of books issued etc.
5. Add/update/delete the information of other user categories (students/teachers)
6. Contact parents of the students, if the issued books not returned within specified time

The features available for to the teachers are:

1. Create an account in library system
2. View/search the different categories and list of books available in library
3. View/update self/student information
4. Check out the books to the students
5. View the books issued to the students
6. Get the report of the history of the books issued to a student

The students are availed with the following features:

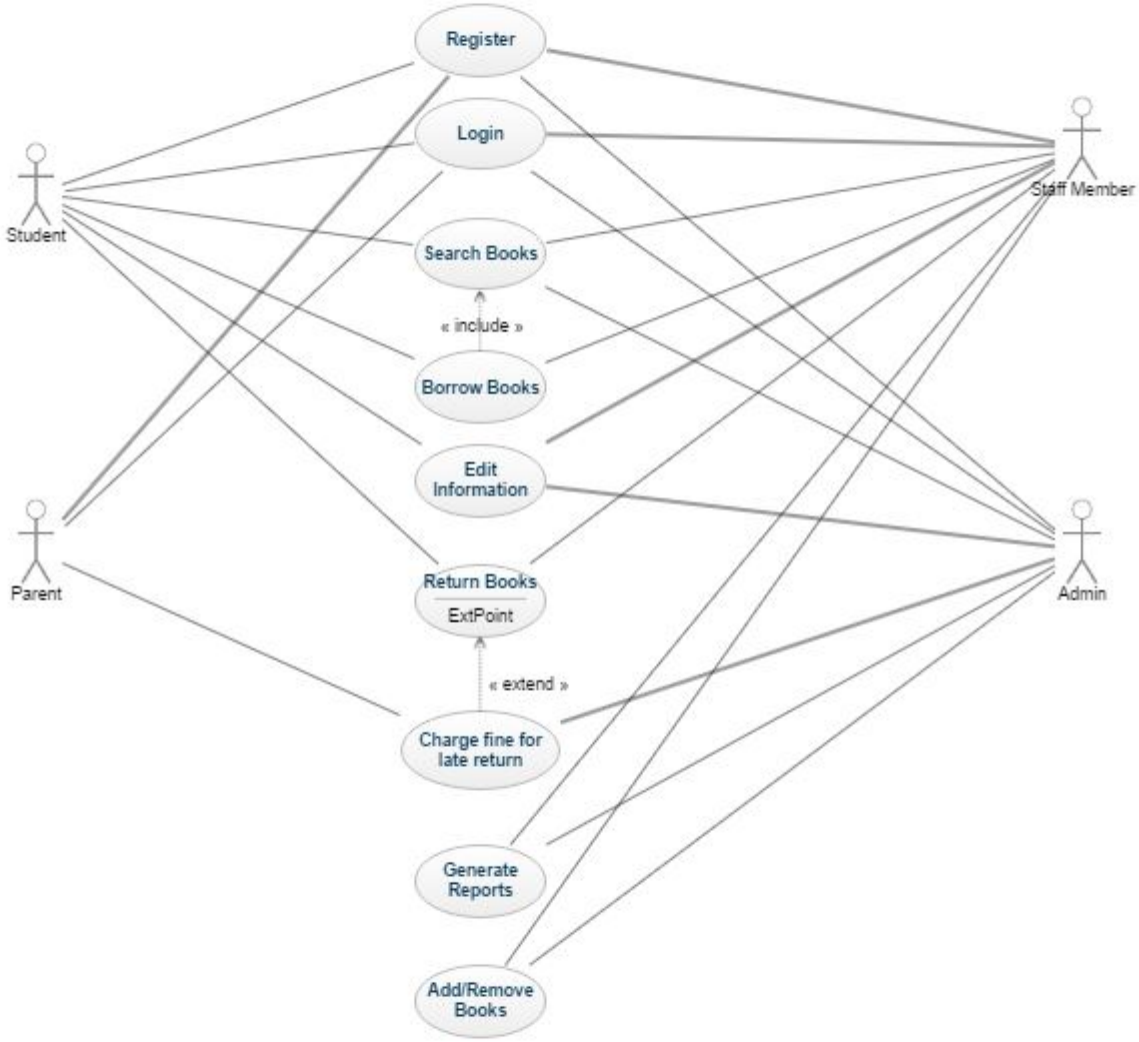
1. Create an account in library system
2. View/search the different categories and list of books available in library
3. View/update self-information
4. Request for a new book
5. View the list of books issued to him/her
6. Get the history of the books issued to a student

The features available for parents are as follows:

1. View the child’s book borrowing history
2. Pay the penalty for the delay in returning the books

2.2 Use-cases

2.2.1 Use Case Diagram



2.2.2 Use Case Descriptions

1. **Use Case #1 Registration**

   Primary Actors: Admin, Students, Staff Members (Teachers), Parents

   Pre- Condition: The student should have a valid school membership document which contains his name, date_of_birth, course registered etc. to obtain library membership and borrow the books. The same criteria applies for registration of teachers and parents of the student members.

   Scenario:

   1. To register, a student goes to the register page and fills the form.
   2. After the form is submitted it goes to admin for validation and approval.

3. The admin searches for the existing students' list. If the student is already registered there is no need of any action.
4. If the student is not registered then all of his/her information will be entered into a DB.
5. A unique student_id for library membership is generated and provided to the student.
6. To register, a staff member goes to the register page and fills the form.
7. After the form is submitted it goes to admin for validation and approval.
8. The admin searches for the existing members' list. If the staff is already registered, no need of any action.
9. If the staff is not registered then all of his/her information will be entered into a DB.
10. After successful registration the staff member is provided with a unique staff_id.

2. Use Case #2 Login

Primary Actors: Admin, Staff, Students

Pre-Condition: If the student or staff or admin wants to login then he should have been first registered.

Scenario:

1. To login the student or staff or admin should first open the Login page.
2. They should then enter their email ids and passwords.
3. Once the id and password are verified they are moved to the options page where they can search, view and perform other operations.

Alternate Scenario:

1. The login fails after entering login and password.
2. It is the responsibility of the admin to see that all staff and students are properly registered and can login to the system.

3. Use Case #3 Searching Books

Primary Actors: Admin, Students, Staff Members

Pre-Condition: The book or student to be searched should have been registered in the database of the library management system.

Scenario:

1. The student or staff logins to the system with their email id and password.
2. If the login is successful, student or staff enters the book name or ISBN or author name and clicks on search.
3. If the search is successful, then that book is displayed on the screen.
4. To search for a student, the admin or staff member logins to the system.
5. If the login is successful then it is possible to search for any student by entering the student's id.
6. To search for a staff member the admin enters the staff's id.
7. If successful he can search for any of the staff members.

Alternative Scenario:

1. The student or staff can re-register themselves.
2. If the search is unsuccessful then the admin should add that members.
3. If the book search is unsuccessful then that book should be added.

4. Use Case #4 Transaction (Borrow/Return)

Primary Actors: Admin, Students, Staff Members

Pre-Condition: To return or borrow any book it is important that the student or staff member is registered with the library and the book to be borrowed is available with the library. To return the book the precondition is that the student and the staff member is registered and that book borrowing record is available with the library.

Scenario:

1. Borrow a book:
    1) If a student wants to borrow a book, the student should login to the system and send a borrowal request.
    2) Once the request is received, the admin checks for the criteria on the particular student such as validate the student information, how many books have already been borrowed, maximum number of books allowed to be borrowed etc.
    3) If the condition is success and book is available it can be borrowed.

2. Return a book:
    1) If the book is available the staff and admin should check the report data for any pending fine.

2) If no fine is pending the book can be returned.

Alternate Scenario:

1. If the student search is unsuccessful and no data is found then the admin should first register the student.
2. If the book search is unsuccessful and book data is not found then admin must enter the book in requisition report.

5. Use Case #5 View/Edit Information

Primary Actors: Admin, Staff

Pre-Condition: To view the details of any book or edit book details that book should be part of the library database. Similar is the case with the student and the staff members.

Scenario:

1. To view details or edit details of any book the admin should first login to the library system.
2. If login successful, they must search for that book by putting book id or title or isbn.
3. If the book is found they should enter book id to view the details and also edit it.
4. To view details or edit details of any student or staff the admin or staff should first login to the library system.
5. If login successful, they must search for that student by putting student id.
6. If the student is found, they can view the details and also edit them.

Alternate Scenarios:

1. If book search is unsuccessful then that book is unavailable and cannot be viewed or edited.
2. If staff search is unsuccessful then that staff member is unavailable and cannot be viewed or edited.

6. Use Case #6 Book (Add/Remove)

Primary Actors: Admin, Staff

Pre-Condition: To add any book that book should requested by a staff member and to delete the book the book must be part of the library database.

Scenario:

1. The staff should login to the system to request for any new book to be added to the library.

2. If login is successful, then to add a book, the staff should fill in a form and submit the request.
3. Once received the request, the admin should login to system and check for the requested book.
4. If the book is not currently available, it can be added to the book database by hitting the online web service and getting the details of the book.
5. To remove a book, it is again searched in the library system by admin.
6. If it is found, it should be checked in borrowers_list.
7. If it is not in the borrowers_list it can be removed from the database.

Alternate Scenarios:

1. If a book is to be added and on search it is already found it should not be added again.
2. If book is to be removed and on search it is not found and is also not part of borrowers_list then there is no need to remove it.

7. Use Case #7 Report Generation

Primary Actors: Admin, Staff

Pre-Condition: To generate any report the staff or admin should be registered with the library and to generate report on any book or student they should be part of the library system.

Scenario:

1. The staff or admin logins to the system.
2. Once logged in successfully, admin can generate different reports such as a report of all the checked out books, report of list of all the authorized users and their access level.
3. To generate report on defaulters the staff or admin should first search for that student.
4. If the student is found then check for the due date on the borrowed books and accordingly add in the defaulters list along with the parents info and the due amount.

3. Data Model and Description

3.1 Data objects

Data objects and their major attributes in our database:

1. User: Stores their unique ID and password combination.
2. Admin: Can Add/modify/delete book information. Stores Unique ID's, and Names of the Staff members.
3. Student: Stores Student ID, Name, Address, contact details, Department information, and book information.
4. Faculty: Stores ID, Name, Contact details, and department Information.
5. Books: Stores Book ID generated by the school, ISBN, Publisher details, Total number of books, Books available
6. Publisher: Store Publisher ID, Name and year of publication.
7. Department: Has Department ID, Name, and courses it offers.
8. Courses: Stores Course ID and name of the course.
9. Parents: Parents can login using their student's ID.

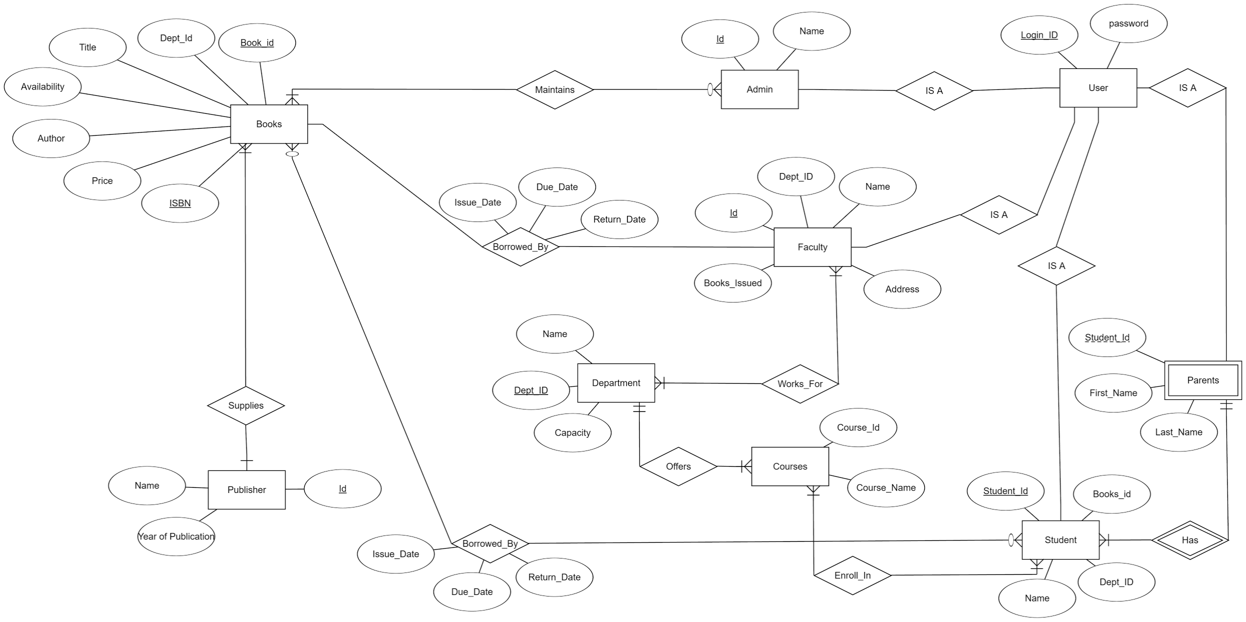
3.2 Relationships

Relationships among data objects are described below:

1. Admin-Book: Admin/Librarian issues books to students and faculty. Admin can Add/modify/delete book information
2. Student-Book: Students borrow Books. Maximum number of books that can be issued to a student should not be more than the number of courses he registered for.
3. Faculty-Book: Books are issued to the Faculty members. They can search for books, check out books related to their respective departments.
4. Book-Publisher: The publisher supplies Books.
5. Student-Parent: Parents can use student id to log in.
6. Faculty- Department: Faculty works for department.
7. Department- Courses: Department offers courses.
8. Student-Course: Student enrol for courses.

3.3 Complete data model

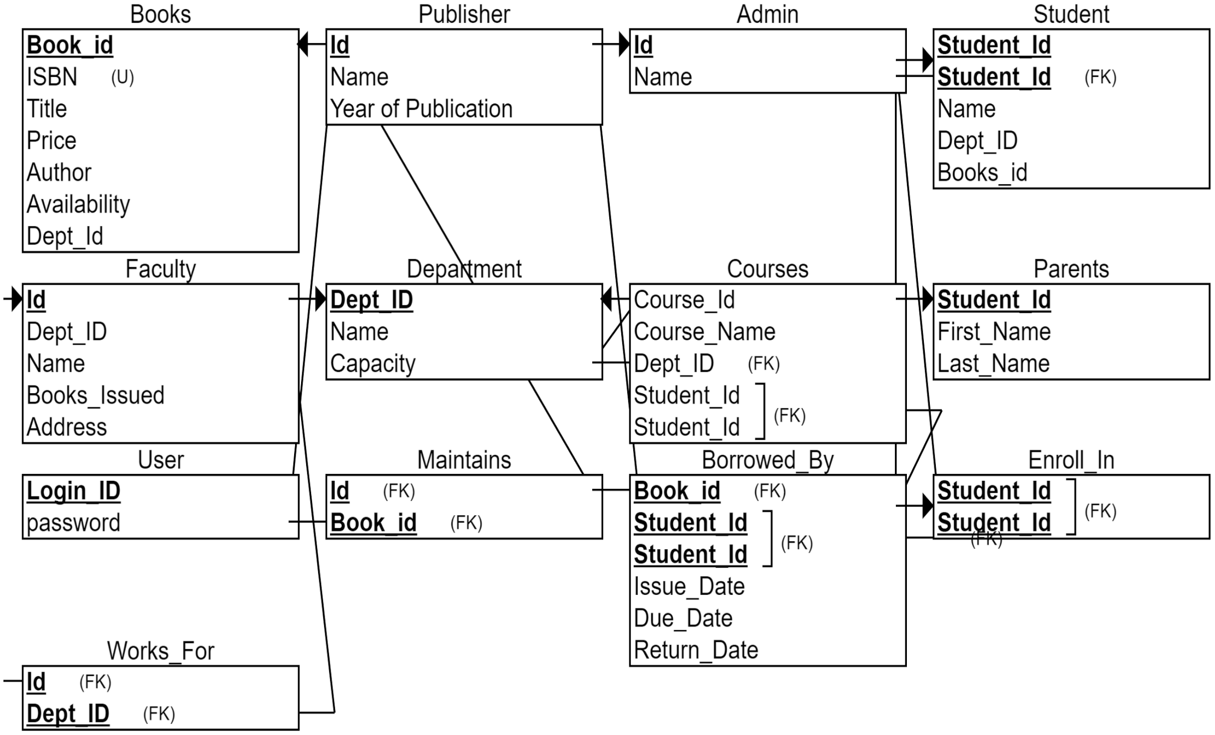
ERD model of the data objects and relationships:



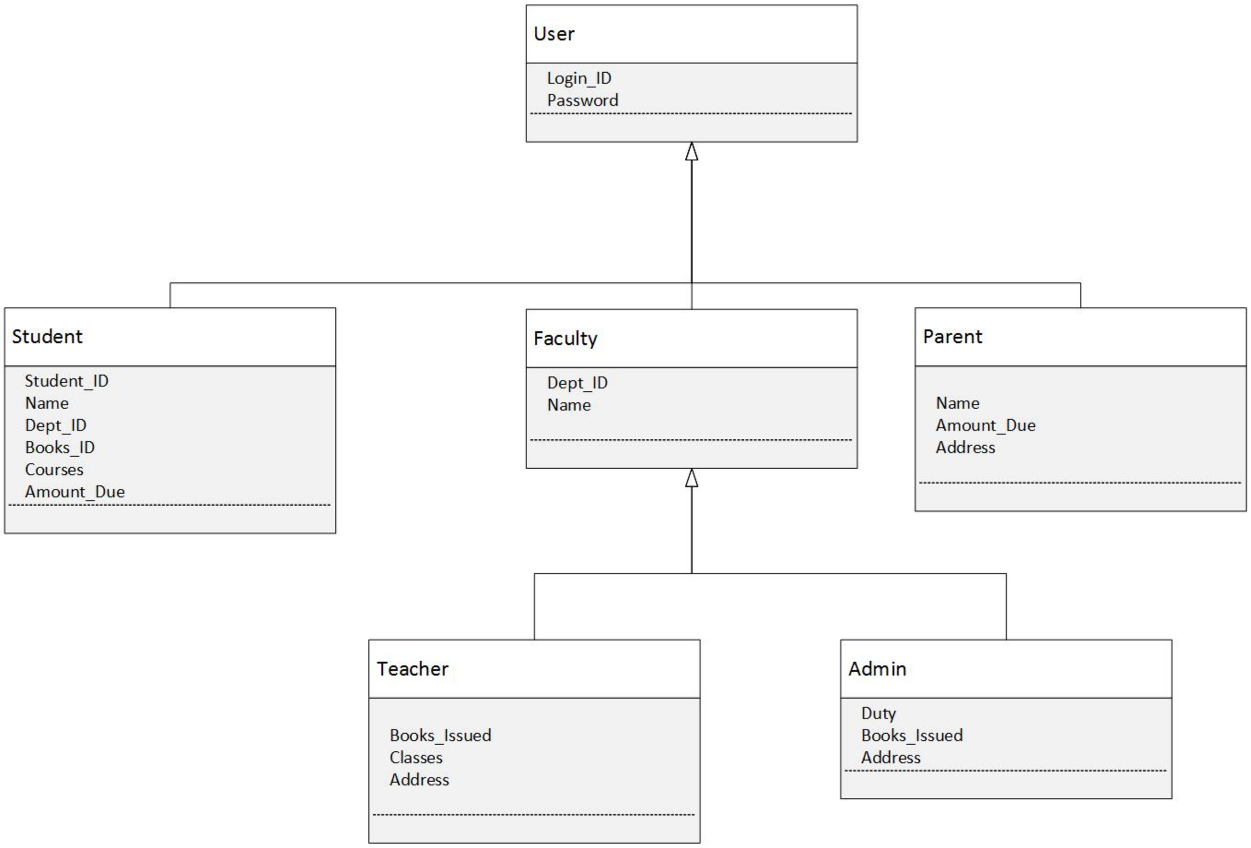
4. Functional Model and Description

4.1 Class diagrams

4.1.1 Class Diagram



4.1.2 User class hierarchy diagram



4.2 Software Interface Description

In this section, we discuss how does our software interacts with other components of the project like users, databases, APIs, and servers etc.

4.2.1 External machine interfaces

For this project to work, there must be interactions with the following machines, they are:

1. A web server to load the web application onto the user's computer.
2. A database server to hold the information of students, teachers, and all.
3. Google books API to get the info of books used.

4.2.2 External system interfaces

As we are developing a web based application and a User Interface, the users need a browser to interact with the Software and check in books.

4.2.3 Human interface

All users will be directed to a login/sign up page, where they can log in or sign up. From there on, different users will be provided with different interfaces:

1. Students will be provided with an interface where they can search for books and their availability, their checked books, return books, check amount due and all
2. Teachers/admins will be provided with an interface containing the list of student requests and their classes and all. They can accept or reject these requests. They can also search for books, students and their classes.

5. Behavioral Model and Description

5.1 Description for software behavior

A detailed description of states and events of different users is made in this section

5.1.1 Student Event and Action table

Current State	Event	Input	Action	Next State
Display Home Page	Login/Sign Up	Login – User ID and Password Sign Up – Fill up the registration page	Validate Login OR Store new user details in database	If login fails, loop to the current state. Else display Main menu
Display Main Menu	Main Menu	Menu Selection	Output Selection	Selected State
Display Book search menu	Search book	Book ISBN/Name/Author's name/ book Id	Query database using user defined parameters	Display book info resulting from query or return to main menu
Display Book info	Book Menu	Menu Selection	If reserve Book request, update the data base	Reserve Confirmation, or return to display book search menu
Reserve Confirmation	Menu	Menu Selection	Output Selection	Display reserve books menu or return to display main menu
Display account info menu	Menu	Update	Output selection (Update)	Selected Page
Display Reserved Books	Get Reserved Menu list	No input	Query database for users reserved books list	Return to display account information on return request

Display Checkout books	Get checked out list	No input	Query database for users checked out books list	Return to display account information on return request
Display pending penalty/due amount	Get Due amount	No input	Query database for users due	Return to display account information on return request
Display Payment Menu	Menu	Menu Selection	Output Selection to banking system	Banking System
Banking System	Payment Validation	Users payment parameters	Validate users payment parameters	If success, update due amount. Else return to payment page
Update due amount	Update Due	Confirmation code from bank system	Update account database to reflect user's payment	Display Account Info Menu

5.1.2 Teachers Events and Actions table

Current State	Event	Input	Action	Next State
Display Home page	Login/Sign Up	Login – User ID and Password Sign Up – Fill up the registration page	Validate Login OR Store new user details in database	If login fails, loop to the current state. Else display Main menu
Display Main Menu	Menu	Menu Selection	Output Selection	Selected State

Display Account options Menu	Menu	Menu Selection	Output Selection	Selected State
Display Book search menu	Search book	Book ISBN/Name/Author’s name/ book Id	Query database using user defined parameters	Display book info resulting from query or return to main menu
Display Student activity search menu	Search student activity	Student ID	Query database using user defined parameters	Display student activity info resulting from query or return to main menu
Display Pending Reservation requests	Menu	Menu Selection	Approve/disapprove requests	if more requests loop to current state else display main menu
Display student account	Menu	Menu Selection	Update Student info	Loop to current state for more updates or display main menu

5.1.3 Parents’ Events and Actions table

Current State	Event	Input	Action	Next State
Display Home Page	Login/Sign Up	Login – User ID and Password Sign Up – Fill up the registration page	Validate Login OR Store new user details in database	If login fails, loop to the current state. Else display Main menu
Display Main Menu	Main Menu	Menu Selection	Output	Selected State

			Selection	
Display Book search menu	Search book	Book ISBN/Name/Author's name/ book Id	Query database using user defined parameters	Display book info resulting from query or return to main menu
Display Book info	Book Menu	Menu Selection	If reserve Book request, update the data base	Reserve Confirmation, or return to display book search menu
Display account info menu	Menu	Menu Selection	Output selection (Update)	Selected Page
Display Reserved Books	Get Reserved Menu list	No input	Query database for users reserved books list	Return to display account information on return request
Display Checkout books	Get checked out list	No input	Query database for users checked out books list	Return to display account information on return request
Display pending penalty/due amount	Get Due amount	No input	Query database for users due	Return to display account information on return request
Display Payment Menu	Menu	Menu Selection	Output Selection to banking system	Banking System
Banking System	Payment Validation	Users payment parameters	Validate users payment parameters	If success, update due amount. Else return to

				payment page
Update due amount	Update Due	Confirmation code from bank system	Update account database to reflect user’s payment	Display Account Info Menu

5.1.4 Faculty/Staff Events and Actions table

Current State	Event	Input	Action	Next State
Display Home Page	Login/Sign Up	Login – User ID and Password Sign Up – Fill up the registration page	Validate Login OR Store new user details in database	If login fails, loop to the current state. Else display Main menu
Display Main Menu	Main Menu	Menu Selection	Output Selection	Selected State
Display account info menu	Menu	Update	Output selection (Update)	Selected Page
Display student search menu	Menu	Student ID	Output selection	Selected Page
Display Book search menu	Search book	Book ISBN/Name/Author’s name/ book Id	Query database using user defined parameters	Display book info resulting from query or return to main menu
Display Book info	Book Menu	Menu Selection	If reserve Book request, update the data base	Return to book info or return to display book search menu
Display Reserved Books	Get Reserved Menu list	No input	Query database for	Return to display

			users reserved books list	account information on return request
Display Checkout books	Get checked out list	No input	Query database for all the checked out books list	Return to display account information on return request
Display Student activity search menu	Search student activity	Student ID	Query database using user defined parameters	Display student activity info resulting from query or return to main menu
Display Book search menu	Search Book	ISBN	Query database using user defined parameters and update book info	Go back to book search menu for more updates or display main menu

5.1.5 Admin Events and actions table

Current State	Event	Input	Action	Next State
Display Home Page	Login/Sign Up	Login – User ID and Password Sign Up – Fill up the registration page	Validate Login OR Store new user details in database	If login fails, loop to the current state. Else display Main menu
Display Main Menu	Main Menu	Menu Selection	Output Selection	Selected State
Display Account Management Menu	Menu	Menu Selection	Output Selection	Selected State
Display Add Account Menu	Add Account	Account Parameters	Query database to validate	If request valid, Display Change

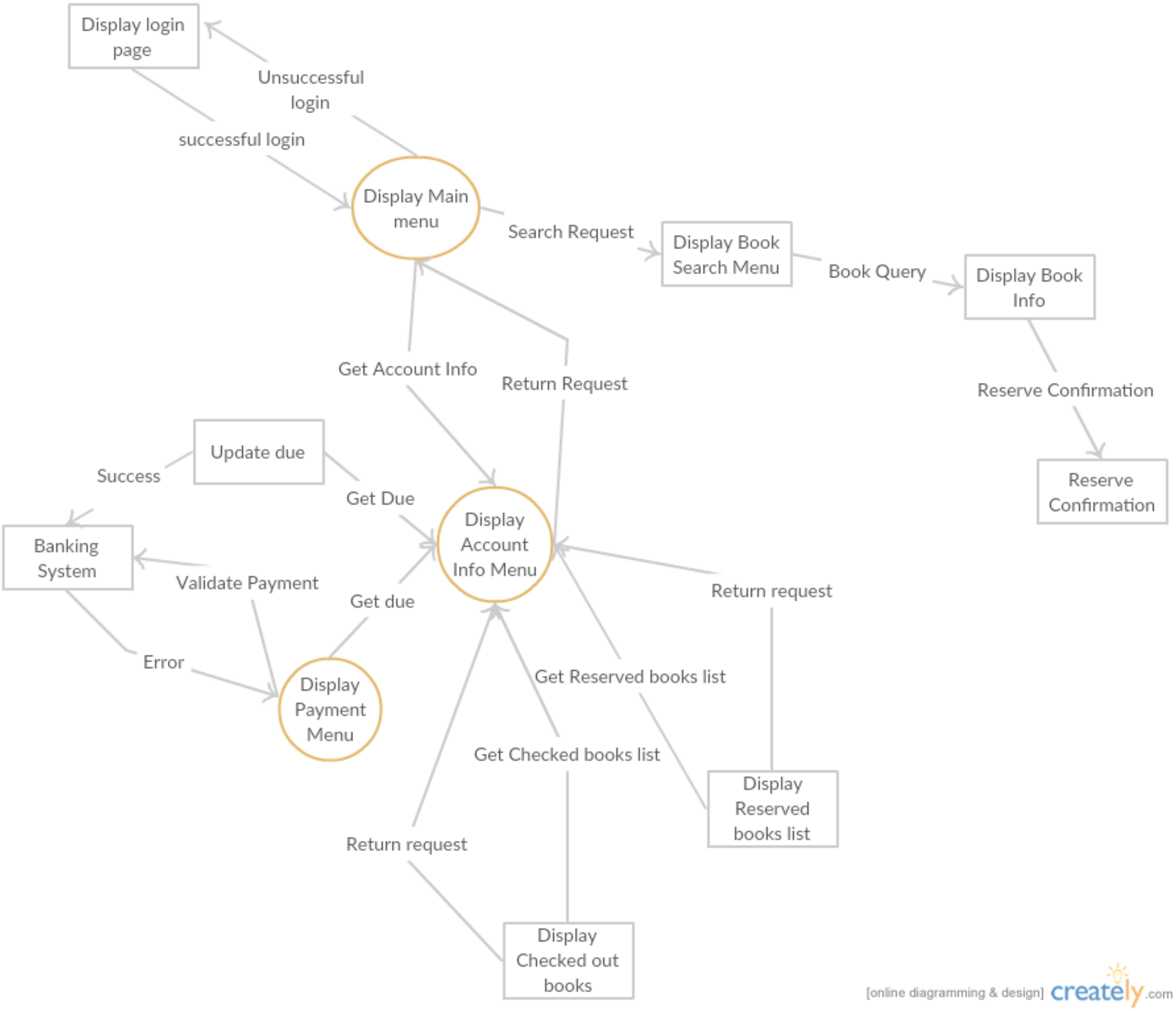
			request	Confirmation. Else loop to current state
Display Update Account Menu	Update Account	Account parameters	Query database to validate request	If request valid, Display Change Confirmation. Else loop to current state
Display Delete Account Menu	Delete Account	Account Parameters	Query database to validate request	If request valid, Display Change Confirmation. Else loop to current state
Display Account Change Confirmation	Confirm Account Change	User confirm	Update database to reflect change	Display Account Options Menu
Display Media Management Menu	Menu	Menu Selection	Output Selection	Selected Menu
Display Add Books Menu	Add Books	Books Parameters	Query database to validate request	If request valid, Display Change Confirmation. Else loop to current state
Display Update Books Menu	Update	Book Parameters	Query database to validate request	If request valid, Display Change Confirmation. Else loop to current state

Display Delete Books Menu	Delete	Book Parameters	Query database to validate request	If request valid, Display Change Confirmation. Else loop to current state

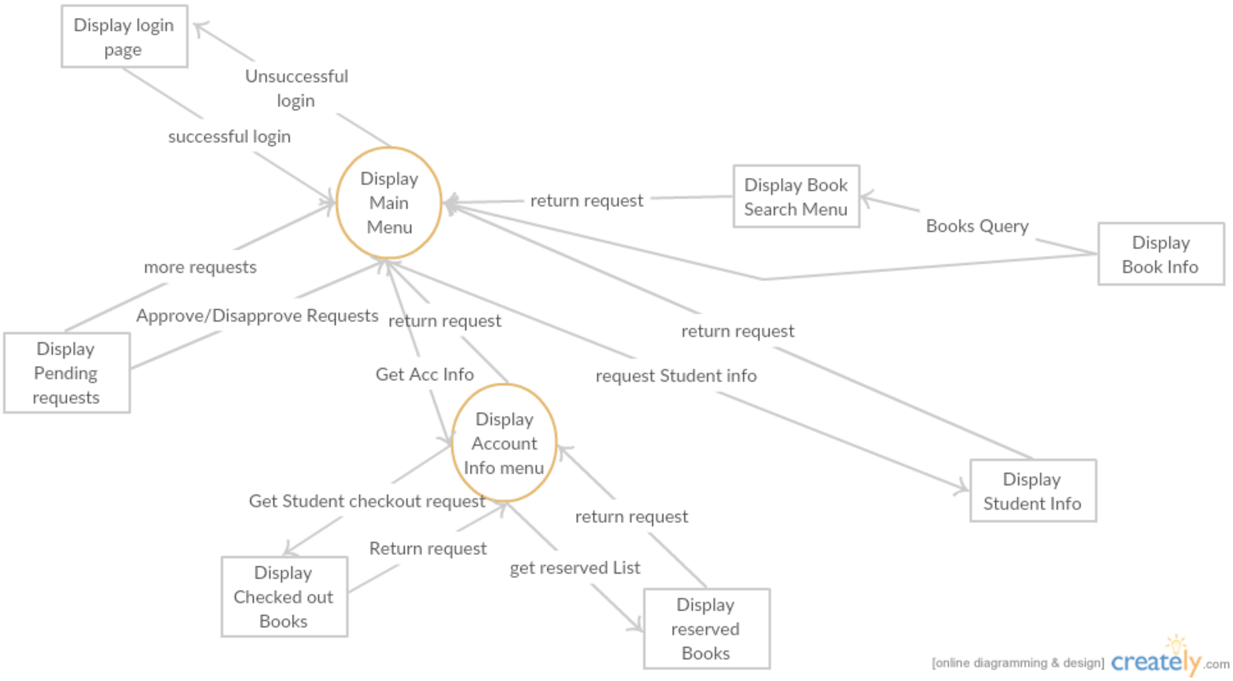
5.2 Statechart Diagram

5.2.1 Student State chart diagram

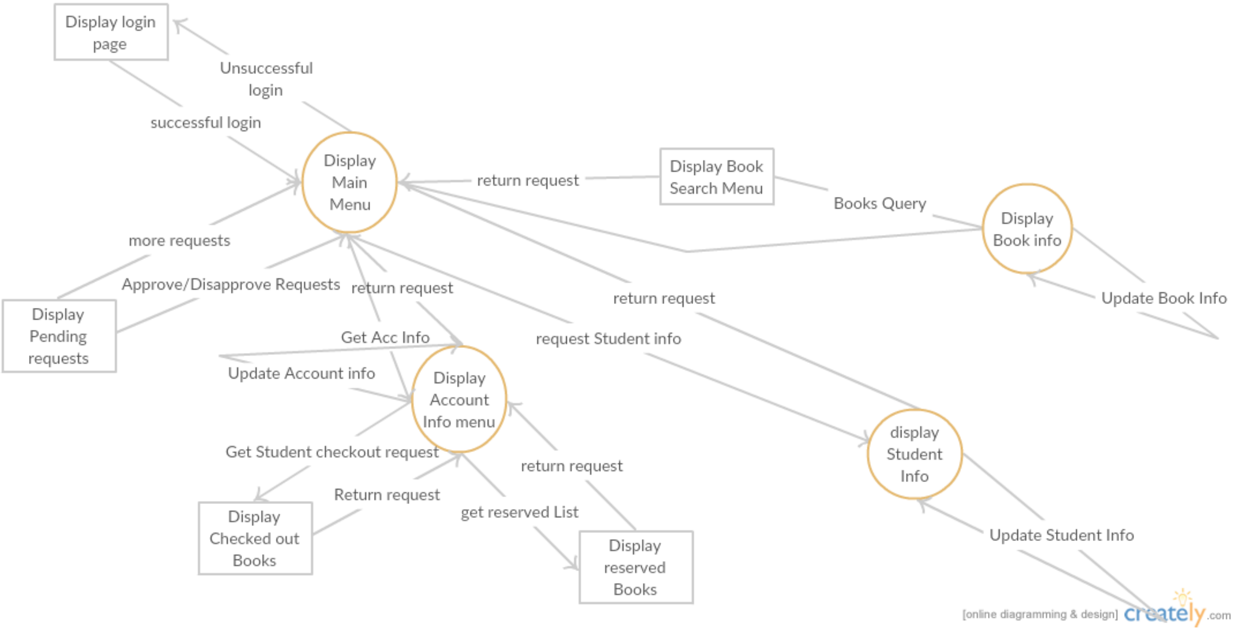
* All the circles and squares represent states



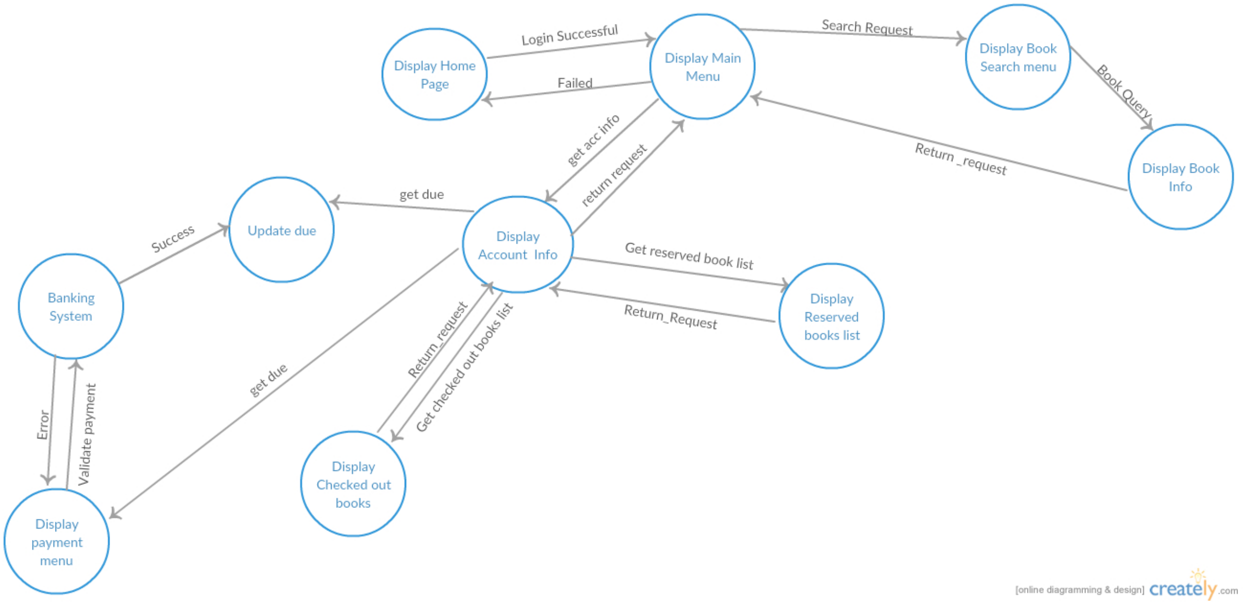
5.2.2 Teacher State chart diagram



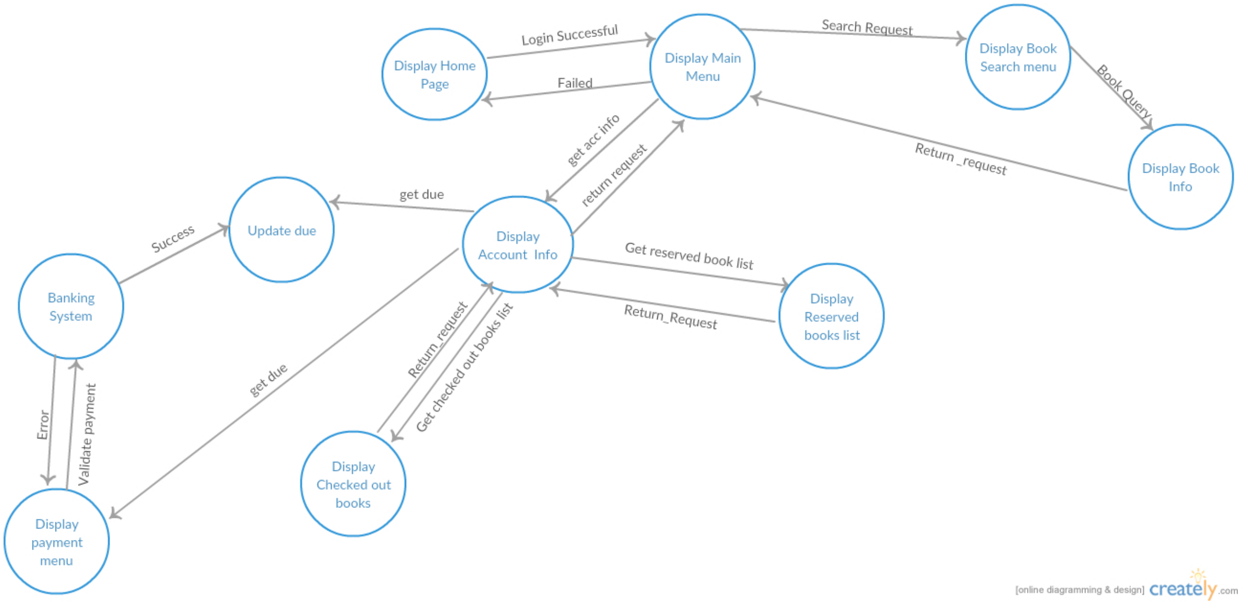
5.2.3 Faculty State chart diagram



5.2.4 Parent State chart diagram



5.2.5 Admin State chart diagram



6.Restrictions, limitations and Constraints

- Some of the users shall have their user ids and passwords generated by the admins before the installation of the system.
- Downtime of Google servers may affect the system as we are using google books API.
- The school's billing system must be properly integrated by the Admins.

6. 1 Assumptions and dependencies

- All the users must have sufficient knowledge of computers.
- All the computers in the school which use this should have the product installed correctly.

7.Validation Criteria

The approach to software validation is described.

7.1 Classes of tests

Testing process ensures that all the requirements have been tested. It also ensures that the given input produces actual results that agree with the required results.

Two major types of testing are performed. They are

1) White-Box Testing

It is also called as “Glass Box Testing”.

- All logical decisions can be checked
- Bugs prevailing in any part of the code can be fixed

2) Black Box Testing

This testing focuses on the functional requirements of the software. This is a complementary approach to white box testing that uncovers different class of errors.

- Interface errors
- Performance errors
- Initialization and termination errors

7.2 Expected software response

Test cases will be generated for all the modules and analysis will be done to check the results.

- Validation of username/password combination

- Check for response time
- Checking if the book information and availability status is correctly displayed or not
- Check for maximum number of books to be issued

7.3 Performance bounds

- time taken to search books, access information and view issue history should not be more than 5sec.
- System shall take as less time as possible.

.