COLLEGES

Diploma in Software Development

Yoobee College of creative innovation.

This sheet must be signed by the student and attached to the submitted assessment.

Course Title:	Software Development	**Course code:**	2202-Online-DSD
Student Name:	Ubiranilson Sabino Dantas Oliveira	**Student ID:**	270095370
Assessment No & Type:	Integrated Studio II Assessment 2		
Due Date:	23/11/2022	**Date Submitted:**	22/11/2022
Tutor's Name:	Arthur		
Assessment Weighting	100%		
Total Marks	15		

Student Declaration:

I declare that:

- I have read the Yoobee College policies and regulations on assessments and understand what plagiarism is.
- I am aware of the penalties for cheating and plagiarism as laid down by the Yoobee College • This is an original assessment and is entirely my own work.
- Where I have quoted or made use of the ideas of other writers, I have acknowledged the source.
- This assessment has been prepared exclusively for this course and has not been or will not be submitted as assessed work in any other course.
- If I am late in handing in this assessment without prior marks will be deducted, to a maximum of 50%.

Student signature:

Date: 22/11/2022

Contents

Section 1: Acknowledgement

We take this opportunity to thank everyone that has collaborated directly or indirectly with the completion of this project. Family, friends, users, tutors, and classmates you all are important to us, thanks for giving support and encouraging students during the journey of this course.

We also take this occasion to thank Yoobee College for all the contents provided guiding us to the way of learning outcomes that are important to our future. We would like to show our gratitude to all citizens that supported each other during the hard time of the last few years.

Section 1.1: Introduction

The present work approaches the case study to develop an information system with specific characteristics to manage a library. The book market has undergone major changes, many of which are due to the advancement of technology that allows readers to buy books over the internet or even read books digitally on a portable computer developed especially for this purpose. This makes the reader feel attracted to technology and stop going to a bookstore to buy books to practice traditional reading.

To attract readers to bookstores again, it is necessary to offer a quality service and guarantee good service. Making the best-selling books reach the reading public with the same speed that the internet offers, but for that to happen, bookstores need to be prepared. Therefore, the objective of this work is to carry out an analysis of the processes of a library, to see how they work to improve them with the help of an information system.

The information system of a library will guarantee easy management of data using the Online Library Management system which is computerise, automate, and conduct all other processes related to member information, book issues, and book returns. The computerisation of the library aids in many of its maintenance procedures.

Most of the manual work is minimised, which lessens the management team's workload and risk of data loss.

Section 2: Objectives

The project aims and objectives that will be achieved after completion of this project are discussed in this subchapter. The aims and objectives accordingly with business and users' requirements are the following:

- Administrator and members can access the system.
- The admin access can edit books.
- Administrator can add, edit, or delete data.
- System log message when book is overdue.
- Members can see catalogue with book availability.

Library Management System is an application which refers to library systems which is generally small or medium in size. It is used by librarian to manage the library using a computerized system where the administrators can add new books, user, and control all data from the library.

Books and student maintenance modules are also included in this system which would keep track of the students using the library and a detailed description about the books the library contains. With this computerized system there will be no loss of book record or member record which generally happens when a non-computerized system is used.

All these modules can help librarian to manage the library with more convenience and in a more efficient way as compared to library systems which are not computerized.

Section 2.1 Self-reflection

During this project a few challenges could be identified, as the age of the technology keeps growing so fast more and more different challenges coming along so, improvements and changes need to be made as they are necessary for business guarantee customers satisfaction.

The information age in digital support is happening however if librarians had and still have great concerns about physical space, maybe now they have one more thing to worry about: these data that have to be stored and the servers where they are hosted not only have to ensure a copy security, as well as maintaining the continuity of reading of the media and the easy access and quick response to several users at the same time.

This brings new challenges arise, technology survives in shorter periods than the information transmitted so there is a need to make the continuity of the life cycle of each IT solution compatible with the necessary migration between different storage and access platforms. Most libraries do not have the necessary equipment so that their users can take advantage of these technologies that are already part of our current way of being in life.

It is necessary to rethink the space and in the current concept of library, one cannot forget the easy access to the network, structured or wireless, they are our reality today and users no longer have only simple access to bibliographic references to now have the full text. The key concept for all this change is the transition induced by technologies such as: XML, Web services, service-oriented architecture embodied in its most comprehensive conception - Web 2.0 (integration of social computing as the central architecture of communication) which is greatly enhanced by the transformation of personal computing into mobile computing made possible with cloud architecture.

Faced with this new reality libraries currently have documents in various supports, which turns them into hybrid libraries, where traditional models of integrated management are no longer adequate. The discovery of new interfaces it will replace or complete the system's requirements helping the system to attend the requirements. The construction of this new model comes to change the legacy of monolithic architectures. We then observe that libraries have been undergoing major changes due to technology and in particular the new design of the Web environment. Libraries feel this pressure for change, it is a parallel path in the face of user demands.

This project worked well in terms of implement the main idea of how a library management system should work however the challenges along this project brought some issues and concerns about what customers are expecting when accessing the online system. The difficulties then were exposed and a deep analyse needed to be made to find out which issues are the most common and how can we deal with them.

For my personal learning outcomes, this project brought me reflection about how technology it will change the world we are living now and what's the future look like, where we will be able to go, and what we will be able to do. Alongside these thoughts, I personally have concerns about my young children and how they going to deal with all these facilities and different world of technology.

Working on this project as a member of a group is another challenge to be mentioned as we are all different from each other and kindness must be put into practice. Helping each other is essential for success in any group project. The experience acquired during this project can add in professional and personal terms, since the challenges and the problems encountered during the development of the project must have their decision-making together.

Section 3: Timeline and constraints

This project needs to be completed in 6 weeks which the tasks will be shared between the team members and each member has a particular task to be complete. In all projects constraints are present and need to be discussed to assure that the project will be successfully implemented.

For this project the 3 main constraints are: time, scope, and cost. These three constraints are known as the triple constraints or the project management triangle. They are all connected to each other which means if we must increase the scope of the project will likely require more time and money to finish the tasks.

Time constraint: The project's completion timetable, which details the due dates for each stage of the project as well as the date of rollout of the final deliverable, is referred to as the time constraint.

Scope constraint: The objectives, deliverables, features, and functions of a project are defined by its scope, along with the tasks necessary to achieve it.

Cost constraint: All the financial resources required to finish the project on schedule and within its set scope are included in the project cost, often known as the project's budget. Remember that cost includes not just the money spent on goods but also the price of labour, vendors, quality control, and other elements.

Section 3.1: Time constraints

Planning: This entails outlining the primary goal(s) of the project team, how it plans to accomplish the goal(s), and the tools and/or procedures it will use to do so.

Scheduling: The project management team must determine a realistic timetable for finishing each project phase.

Monitoring: The project team must examine how previous phases of the project performed, note trends and their effects on future, and then communicate their findings to all pertinent stakeholders. This step happens after the project has begun.

Control: The team must proceed appropriately in the control stage after presenting the findings of each project phase. To continue and repeat a successful outcome, the team must examine the elements that contributed to it when things are going well. If there has been a derailment, the team must determine how and why it happened so that it may be corrected for subsequent operations.

Section 3.2: Scope constraint

At the start of the project, provide clear documentation of the whole project scope, including all needs.

Create a procedure for managing changes so that, if someone makes a change proposal, there is a regulated process in place for how the change will be examined, authorised, or rejected, and, if necessary, implemented.

Give stakeholders frequent and clear updates on the scope.

Section 3.3: Cost constraint

Both fixed and variable costs, such as those for labour, materials, permits, and other project-related expenses, are included in the budget for a project. A few techniques for determining a project's cost include:

Historical information: Examining prices paid for comparable projects in the recent past estimating the rate of cost for labour and items.

Comparing historical data with current, pertinent variables

Vendor bid: Calculating the average price based on many competitive vendors bids

Effective cost control is essential to the project's success.

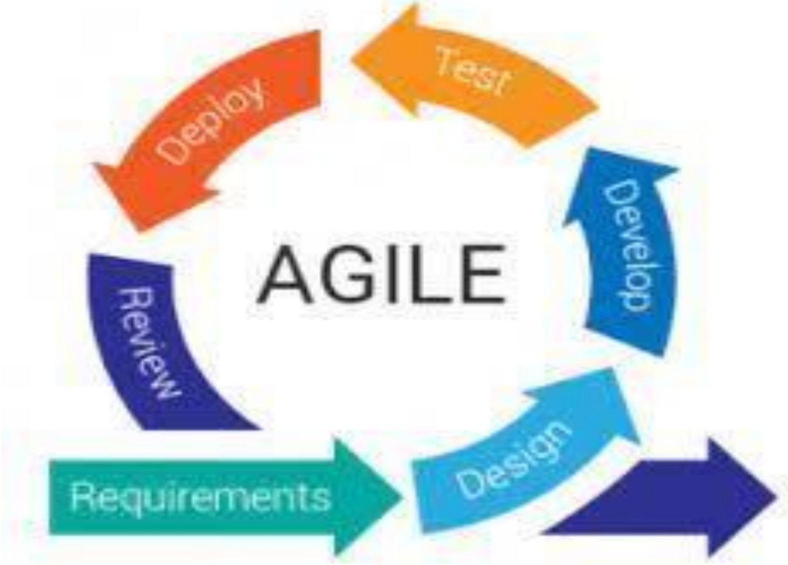
Section 3.4: Team Members and their responsibilities

In this section, we will discuss and analyse about the developing process of Library Management System including software requirement specification (SRS) and comparison between existing and proposed system.

The functional and non-functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process begins. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one. The project will be implemented and shared with two members of the team which they are Ubiranilson and Paulo, each member will have tasks to be completed. The main Idea is dividing the project into two parts which one member will be working on project details such as: objectives, constraints, software requirements and specification while the other member will be working on design, wireframe, prototyping and testing.

Section 4: SDLC (Software development life cycle)

The chosen SDLC model for this project is AGILE MODEL.



The Agile model, which divides the product into cycles, quickly produces a usable product and is regarded as a very realistic development strategy.

The methodology generates continuous releases, each of which contains minor, incremental improvements over the preceding version. The product is examined after each iteration.

This paradigm stresses communication because clients, developers, and testers collaborate on the project. However, because this approach strongly relies on consumer involvement, the project may move in the incorrect direction if the client is unsure of where they want to go.

Advantages of Agile model:

- Customer satisfaction by rapid, continuous delivery of useful software.
- People and interactions are emphasized rather than process and tools. Customers, developers, and testers constantly interact with each other.
- Working software is delivered frequently (weeks rather than months).
- Face-to-face conversation is the best form of communication.
- Close, daily cooperation between businesspeople and developers.
- Continuous attention to technical excellence and good design.
- Regular adaptation to changing circumstances.
- Even late changes in requirements are welcomed

Section 5: REQUIREMENTS ELICITATION

The main idea of the project library management system is operating the application efficiency and reduce costs. The automated system pretends complete all tasks involved in operations of the library system, the activities such as book purchasing, cataloguing, indexing, circulation recording and stock checking are managed by the software, this software minimizes the chances of errors as they don't require repetitive manual work.

The software for the library management system aids in cutting expenses for operations. Manually running a library requires a significant amount of work and paperwork. Using an automated method requires less staff and office supplies. Lower operational costs are the result of this. Both the user and the librarian benefit from the system's time savings. The user can search for books in the library with just one click. About questions on the availability of books, the librarian can respond with ease. The process of adding, removing, or editing the database is straightforward. It is simple to add new members or terminate existing memberships.

Within a few hours, the library's book stock can be checked and verified. Compared to the manual system, the automated system saves a substantial amount of time. The volumes are methodically arranged by author, title, and subject by the library management system software, making the library intelligent. Users may now easily and rapidly search for books thanks to this.

Section 5.1: Client's requirements

- Library member should be able to search books by their details such as: title, author, and category.
- Each book has their own ID and perhaps a rack location or something to make it easy to find physically.
- More than one copy of the book.
- System needs to show information about the person that has taken the book in case is not available.

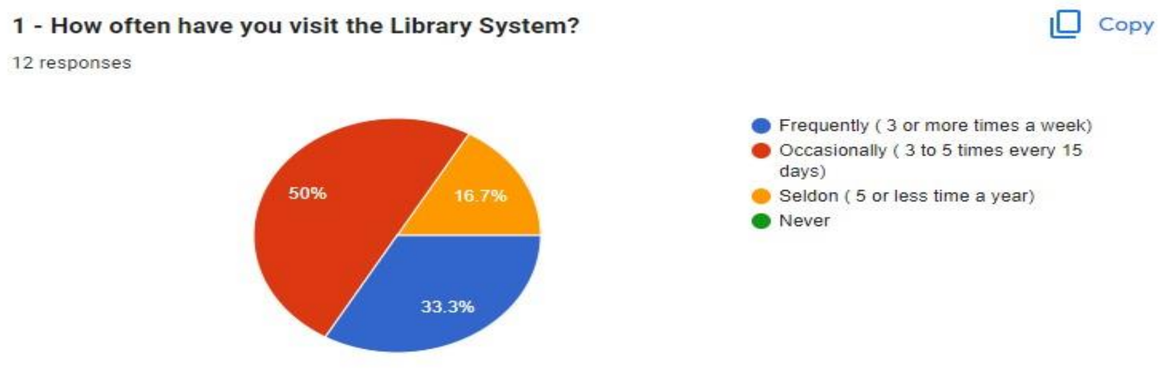
It is important the engagement between the client's requirements, the user's requirements, and what are the expectations from users. Finding out what type of users are engaged with the product helps to identify and make improvements when necessary, according to what the end users need and what they are looking for.

Section 5.2: Research and stakeholders

Additionally, it appears that today's users like services that can be accessed quickly and are constantly available. Users who are used to the Google toolbar and Google desktop gadgets being present everywhere, independent of the website or programme they are using, find it difficult to adapt to the interfaces of library systems and scholarly information sites. For instance, when a user is working through a course management system, viewing their Web based workspace like their Facebook or Connote accounts, or perusing the selection of an online bookstore, the process of scholarly discovery and delivery is not yet sufficiently integrated with other user spaces to allow users to access such services when they need them.

When the user experience offered by current library systems is contrasted with that of well-known online services, they fall short. But when it comes to other factors, like the calibre and applicability of their collections, the extensive metadata they provide for precise searching, the services they can customize for their users, and the control over the overall workflow, lingo, and aesthetics of their application, libraries are better able to meet user needs. Libraries are more likely to provide discovery and delivery services that will best meet the users' expectations and needs by utilizing their special attributes and enhancing the user experience.

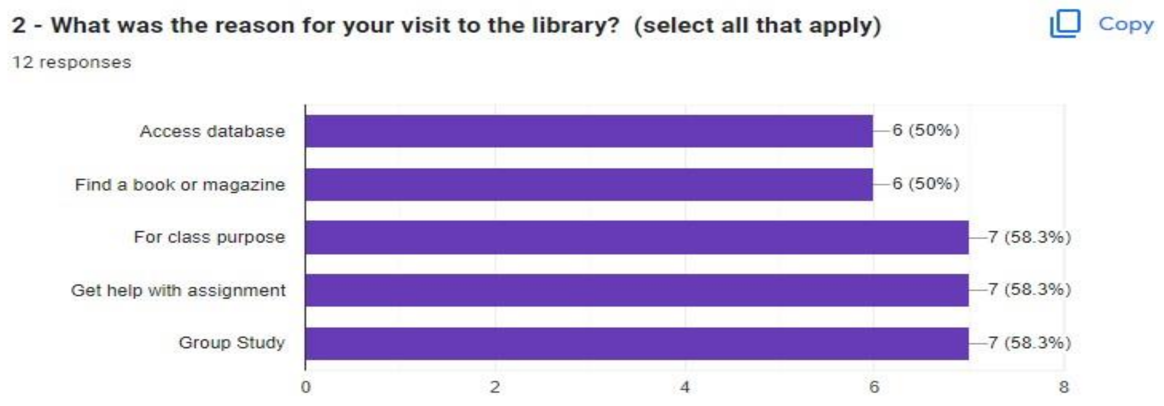
It goes without saying that the user should have access to the most comprehensive and pertinent research data feasible. The Primo system fulfils this criterion in accordance with the library's requirements by collecting data from numerous resources regardless of the type of resource, the format of the resource's catalogue, or the location of the resource. Furthermore, the Primo system normalises the harvested data (makes the data conform to one set of rules) and converts it into a unified format that is designed to fully support the task at hand—a quick, efficient, and friendly search process. Data elements may originate from heterogeneous repositories. Today's technology makes it possible to set up such an infrastructure, and the Primo system follows other pioneers like Google in doing so.



1 - How often have you visit the Library System?

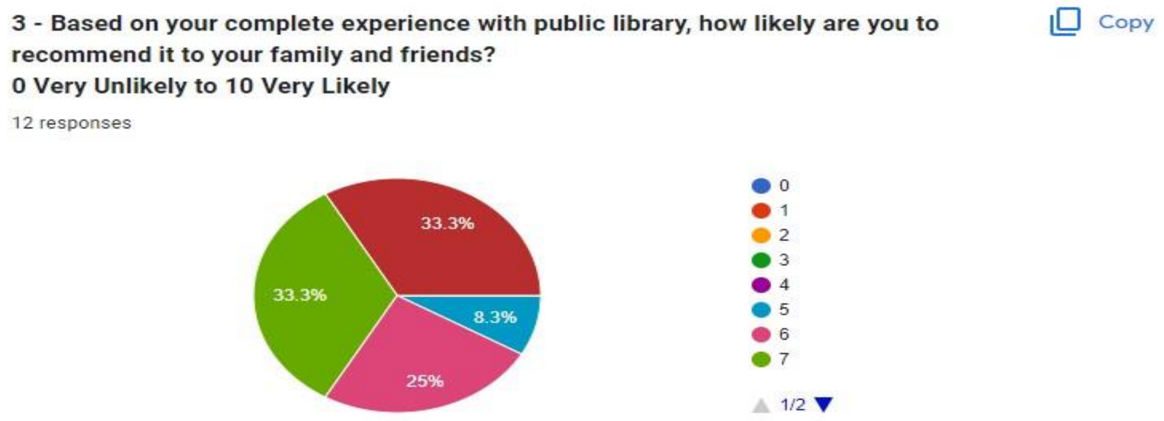
12 responses

Copy
50%
16.7%
33.3%
● Frequently (3 or more times a week)
● Occasionally (3 to 5 times every 15 days)
● Seldon (5 or less time a year)
● Never



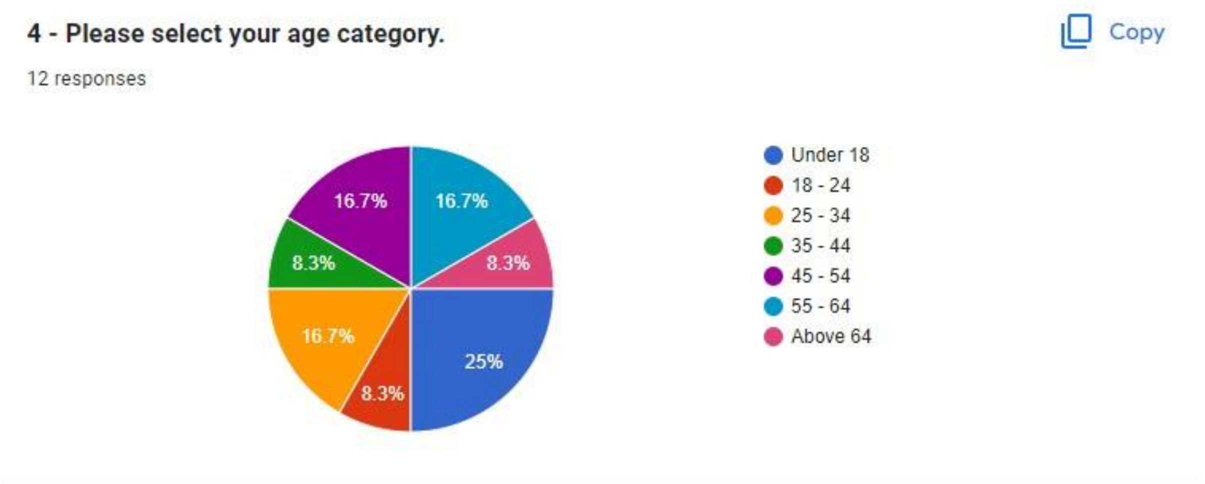
2 - What was the reason for your visit to the library? (select all that apply)

12 responses



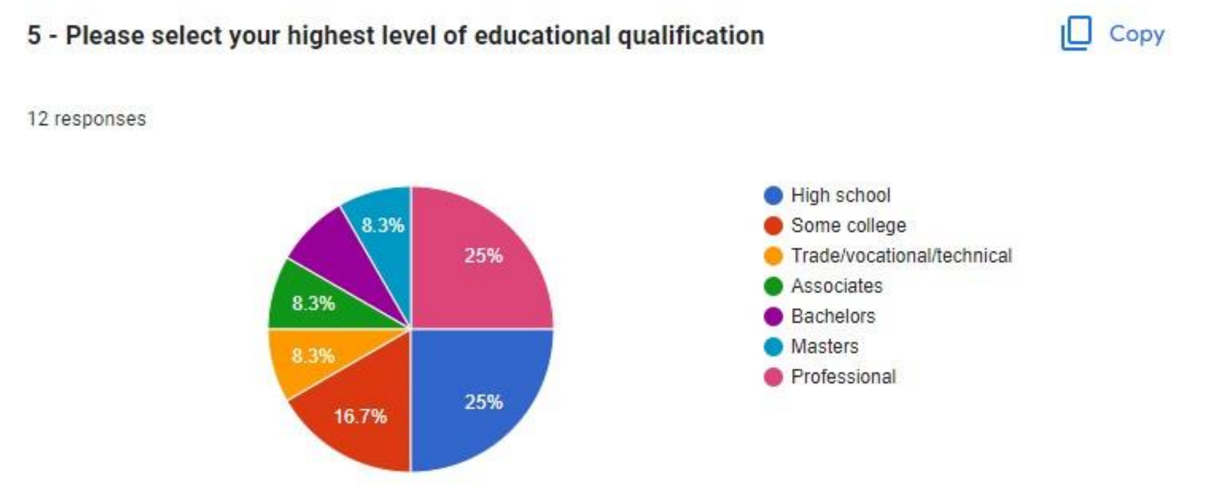
3 - Based on your complete experience with public library, how likely are you to recommend it to your family and friends?
0 Very Unlikely to 10 Very Likely

12 responses



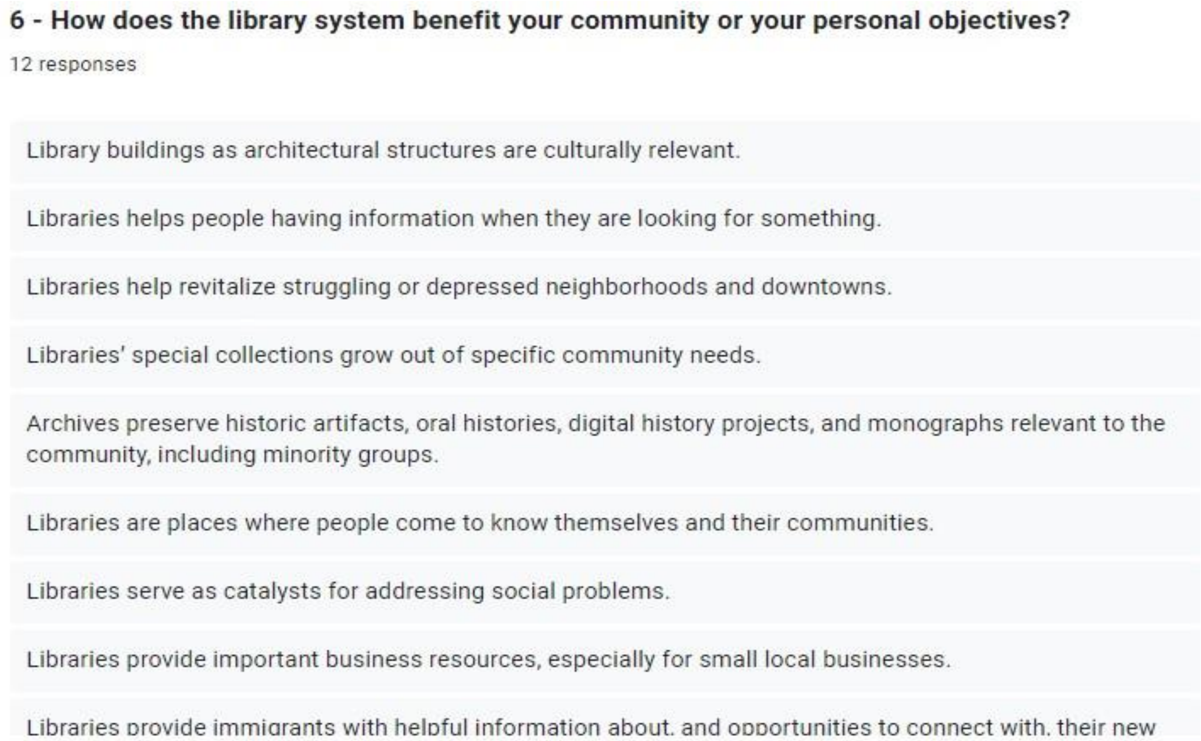
4 - Please select your age category.

12 responses



5 - Please select your highest level of educational qualification

12 responses



6 - How does the library system benefit your community or your personal objectives?

12 responses

Library buildings as architectural structures are culturally relevant.

Libraries helps people having information when they are looking for something.

Libraries help revitalize struggling or depressed neighborhoods and downtowns.

Libraries' special collections grow out of specific community needs.

Archives preserve historic artifacts, oral histories, digital history projects, and monographs relevant to the community, including minority groups.

Libraries are places where people come to know themselves and their communities.

Libraries serve as catalysts for addressing social problems.

Libraries provide important business resources, especially for small local businesses.

Libraries provide immigrants with helpful information about, and opportunities to connect with, their new

Section 6: Software requirements and specification

PRODUCT DESCRIPTION:

Library Management System is a computerized system which helps user(librarian) to manage the library daily activity in electronic format. It reduces the risk of paperwork such as file lost, file damaged and time consuming.

It can help user to manage the transaction or record more effectively and time-saving.

PROBLEM STATEMENT:

The problem occurred before having computerized system includes:

- File lost:

When computerized system is not implemented file is always lost because of human environment. Sometimes due to some human error there may be a loss of records.

- File damaged:

When a computerized system is not their file is always lost due some accident like spilling of water by some member on file accidentally. Besides some natural disaster like floods or fires may also damage the files.

- Difficult to search record:

When there is no computerized system there is always a difficulty in searching of records if the records are large in number.

- Space consuming:

After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.

- Cost consuming:

As there is no computerized system the to add each record paper will be needed which will increase the cost for the management of library.

Section 6.1: System Objectives

- Improvement in control and performance:

The system is developed to cope up with the current issues and problems of library the system can add user, validate user and is also bug free.

- Save cost:

  After computerized system is implemented less human force will be required to maintain the library thus reducing the overall cost.

- Save time

Librarian can search record by using few clicks of mouse and few search keywords thus saving his valuable time.

- Option of online Notice board

Librarian will be able to provide a detailed description of workshops going in the college as well as in nearby colleges

- Lecture Notes

Teacher has a facility to upload lectures notes in a pdf file having size not more than 10mb

Section 6.2: System Requirements and Non-Functional Requirements

PRODUCT REQUIREMENT

When a library management system will be implemented librarian and user will easily access library as searching and book transaction will be very faster.

RELIABILITY REQUIREMENT

The system should accurately perform member registration, member validation report generation, book transaction and search.

USABILITY REQUIREMENT

The system is designed for a user-friendly environment so that student and staff of library can perform the various tasks easily and in an effective way.

ORGANIZATIONAL REQUIREMENT IMPLEMENTATION REQUIREMNTS

In implementing whole system, it uses C++ computer language and VS code as a platform side scripting language which will be used for database connectivity.

DELIVERY REQUIREMENTS

The whole system is expected to be delivered in six weeks of time with a weekly evaluation by the project guide.

FUNCTIONAL REQUIREMENTS NORMAL USER AND
LOGIN

Description of feature

This feature used by the user to login into system. They are required to enter user id and password before they are allowed to enter the system. The user id and password will be verified and if invalid id is their user is allowed to not enter the system.

Functional requirements

- user password is provided when they register.
- The system must only allow user with valid id or password to enter the system.
- The system performs authorization process which decides what user level can access to.
- The user must be able to logout after they finished using system.

REGISTER NEW USER

Description of feature

This feature can be performed by all users to register new user to create account.

Functional requirements:

System must be able to verify information.

System must be able to delete information if information is wrong.

REGISTER NEW BOOK

Description of feature

This feature allows to add new books to the library.

Functional requirements.

System must be able to verify information.

System must be able to enter number of copies into table.

System must be able to not allow two books having same book id.

DESCRIPTION OF FEATURE

This feature is found in book maintenance part. we can search book based on book id, book name, publication or by author name.

System must be able to search the database based on select search type.

System must be able to filter book based on keyword entered.

System must be able to show the filtered book in table view.

Functional requirements.

System should be able to add detailed information about events.

System should be able to display information on notice board available in the homepage of site.

Section 6.3: Software and Hardware Requirements

This section describes the software and hardware requirements of the system

SOFTWARE REQUIREMENTS

Operating system- Windows 10/11 is used as the operating system as it is stable and supports more features and is more user friendly.

Development tools and Programming language C++ is used to write the whole code and develop application with visual Studio Code as an SDK.

HARDWARE REQUIREMENTS

Intel core i5 2 generation is used as a processor because it is fast than other processors an provide reliable and stable and we can run our pc for long time. By using this processor, we can keep on developing our project without any worries.

Ram 4GB is used as it will provide fast reading and writing capabilities and will in turn support in processing.

Existing System:

In the past Libraries were managed manually and it required a lot of time to record or to retrieve the details. The employees who must record the details must perform their job very carefully. Even a small mistake would create a lot of problems. Security of information is very poor and report generations of all the information is a very tough task.

Maintenance of Library catalogue and arrangement of the books to the catalogue is very complex task. In addition to its maintenance of member details, issue dates and return dates etc.

All the operations must be performed in perfect manner for the maintenance of the library without any degradation which may finally result in the failure of the entire system.

Proposed System:

To solve the inconveniences as mentioned in the existing system, an application called library information system which is an online system developed to achieve the following features:

The students will register their details through this system.

Individually each member will have his account through which he can access the information he needs.

Book details like authors, number of copies totally maintained by library, present available number of books, reference books, non-reference books etc. all this information can be made handy.

Regarding the members designation, number of books was issued.

Issue dates and returns of each member is maintained separately and fine charged if there is any delay in returning the book.

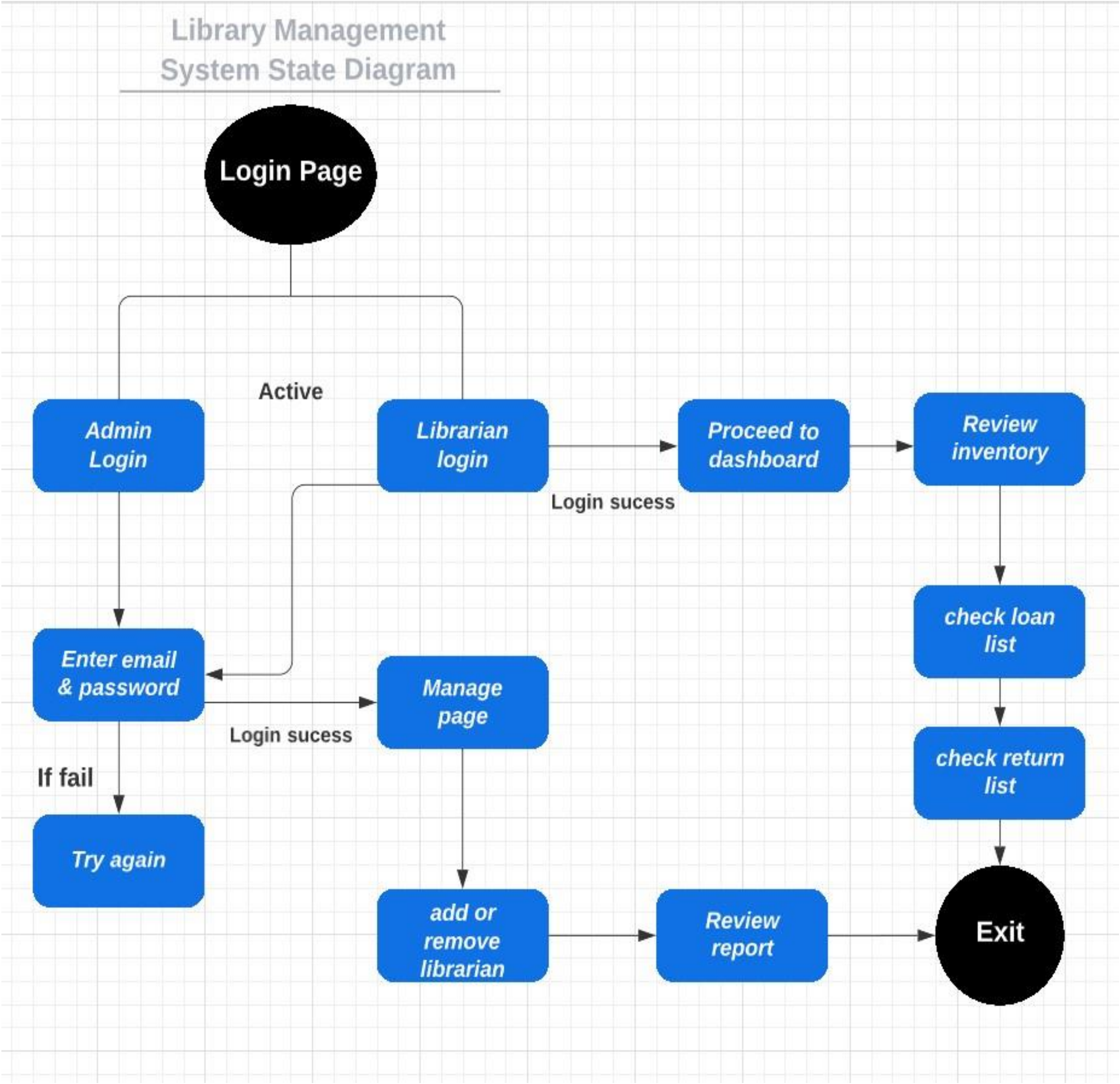
Administrator can add, update the books.

Time consuming is low, gives accurate results, reliability can be improved with the help of security.

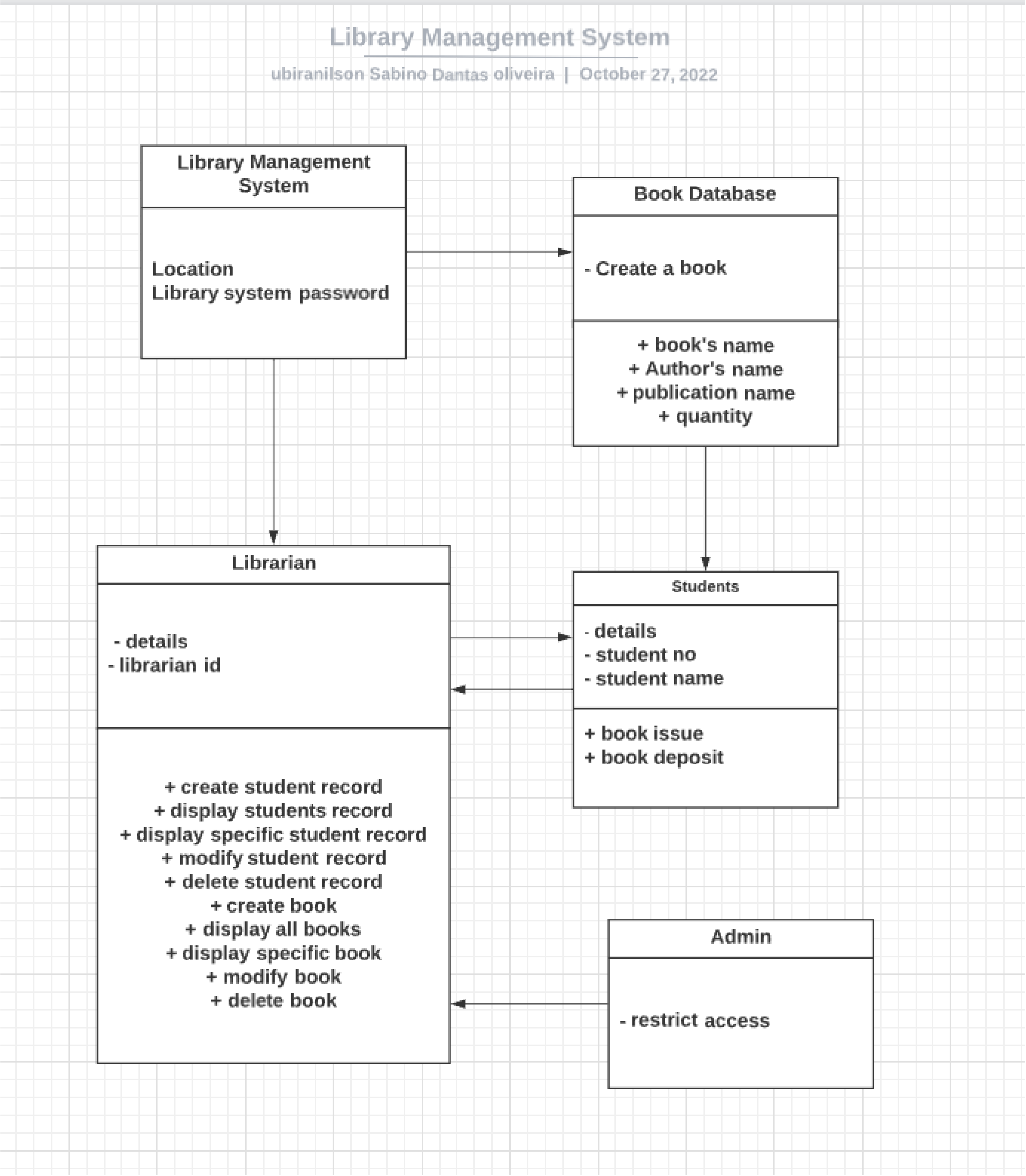
Section 7: Design and tools used on this project:

The design phase will be developed and tested using Figma for sketches, wireframes and prototyping, testing usability also will be provided from users. For the testing scenario survey monkey or google form with question that will help us to gathering users' requirements and suggestions that we can implement and improve UX. User experience and interface will be provided by a prototype where people can look at the application and leave their thoughts about the system. For development process C++ programming language will be used to develop the application, Visual Studio Code is the IDE for supporting and help usability of C++.

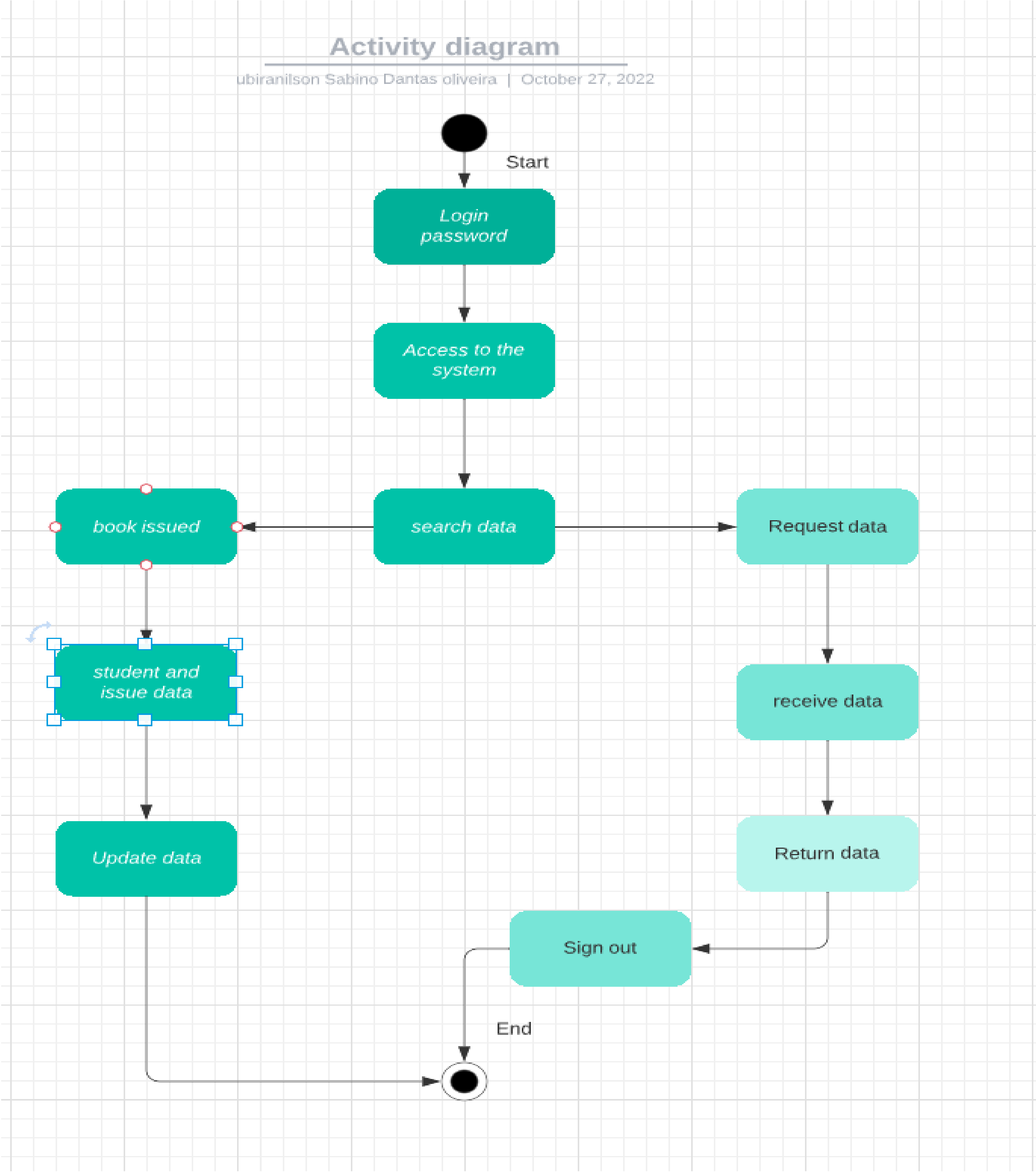
Library management System state diagram



Class diagram



Activity Diagram



Section 8: Design of the project

UI (user interface)

UI designer works with a product's interface. This means that these professionals work with the image of a web page, application, or software and in relation to how the user feels when using the application.

Clearly, this aspect of a product's overall design is critical, as it should be accessible, attractive, and easy to use. The fact that it is attractive, and its image are only aspects related to the product, nothing more. UI designers turn a product into something that consumers can use and enjoy. The purpose of UX design is to create a great experience, but the focus is only on the interface.

Interactions with interfaces are also included in this concept. As mentioned earlier, it is not only about the visual aspect, but also about interactions with the product, regardless of whether it is a “simple” interaction, such as, for example, clicking a button.

User interface (UI) design includes just about everything that can be seen and done by the user, from colour palettes to the size of buttons and sound. Designers establish a hierarchy that allows users to easily navigate the product and helps them achieve the ultimate goals of the software or web page.

Section 8.1: UX (user experience)

UX design is a complicated concept that has yet to be defined. The term includes various applications and ideas and synthesizes different environments, disciplines, skills, and professions in just one function.

Basically, UX design tries to make the user experience positive. The UX designer considers some elements, such as:

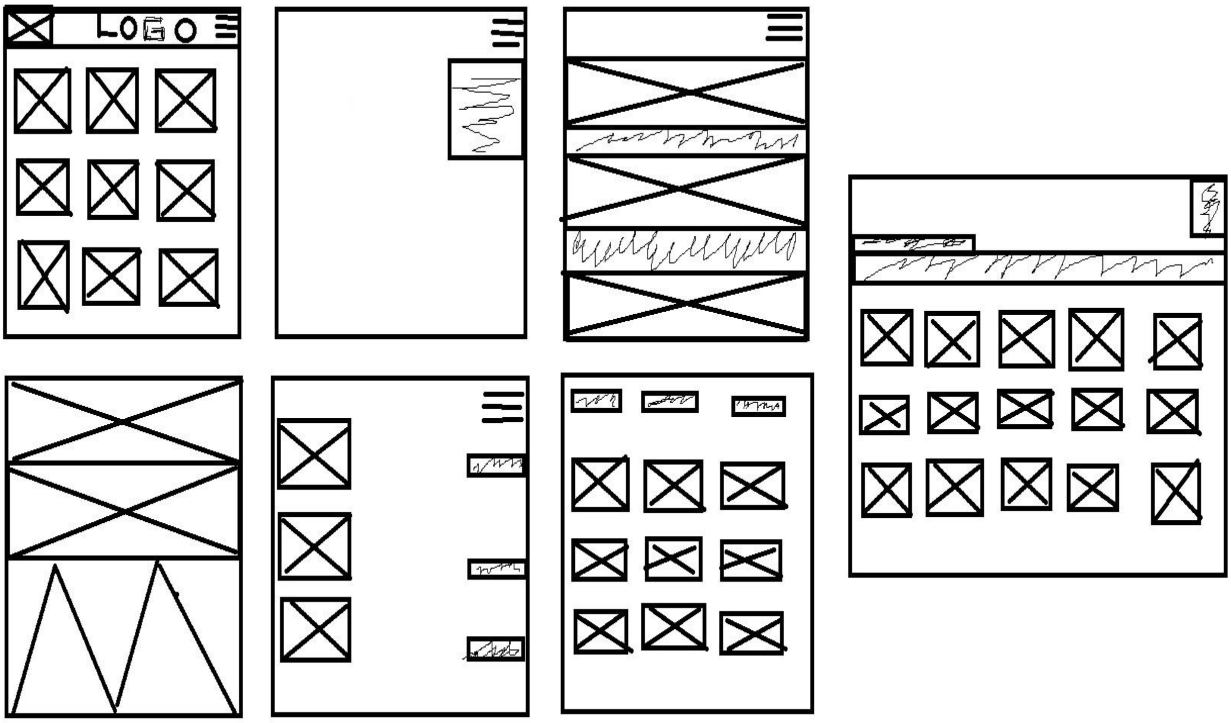
- The way the user interacts with the product
- Software performance
- How the software works
- the visual aspect
- How the product is complemented and how it fits within the overall brand

UX designers aim to meet the consumer's needs and put into practice the purpose of the product, whether it's a website, an application or anything else. Designers constantly analyse the market to understand what users want and what their needs are, always aiming to create a great experience through designs, considering not only usability, but also interactions and behaviours. They want users to feel a certain way.

Designers also look for ways to differentiate their products (in terms of experience) from those already on the market. For this they create identification profiles of their user base, develop functional diagrams, wireframes, and prototypes. From that gathering users' ideas and their experience using the product the developers can make improvements and provides users the best experience when using the application.

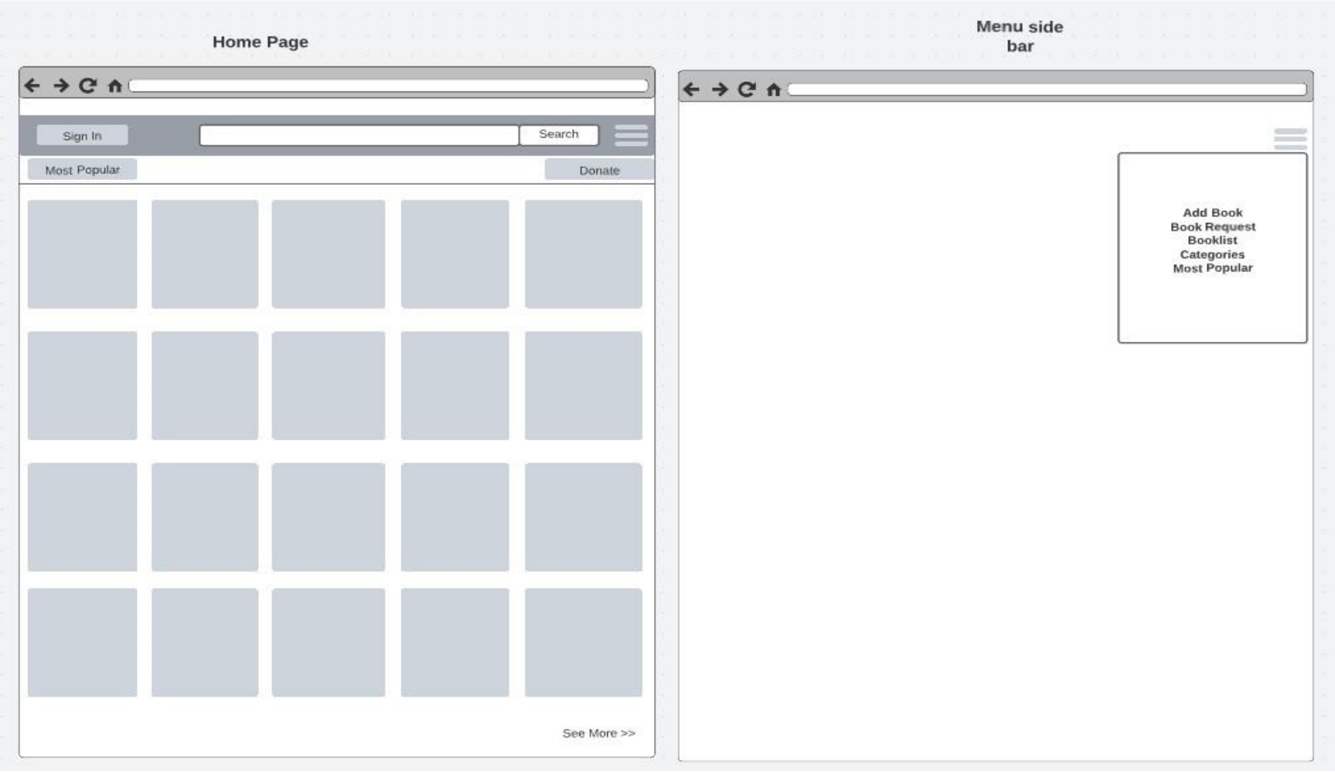
Section 8.2: Sketches

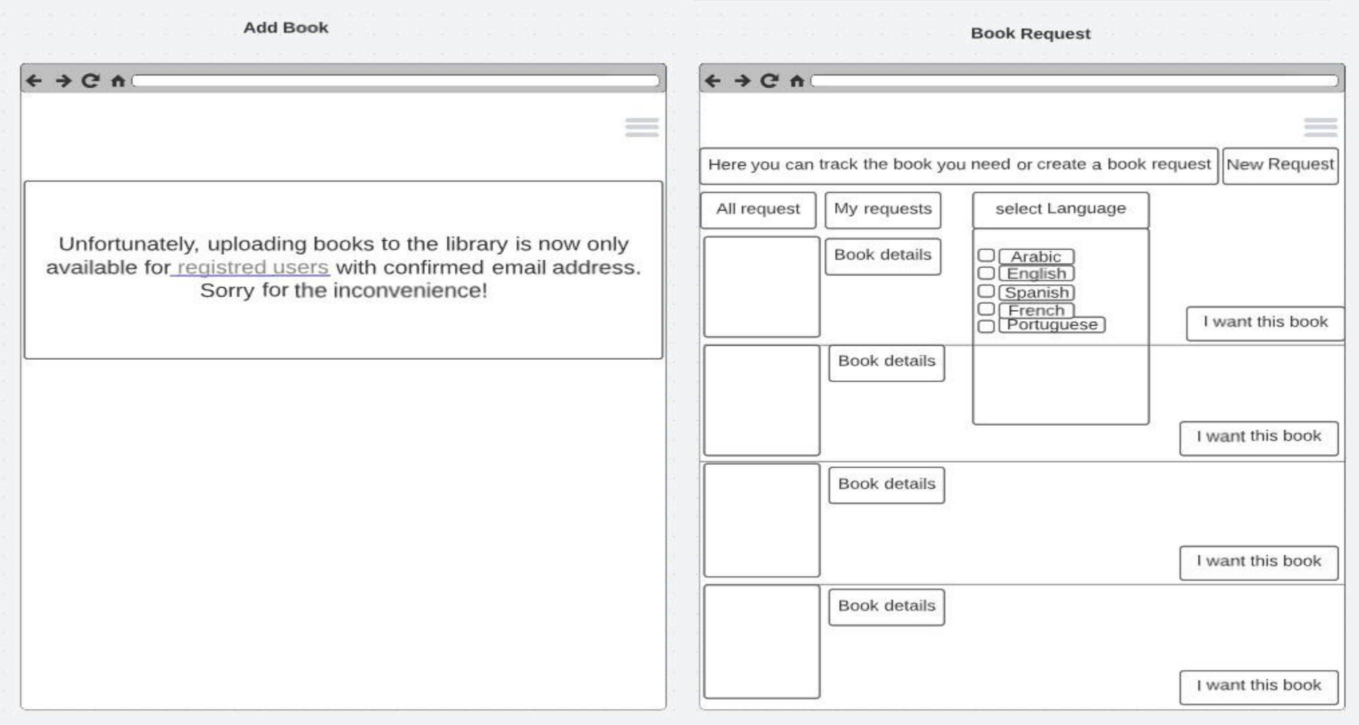
This is the easiest and better way to visualize the ideas about the product. This can be drawn by hand on a piece of paper, on a whiteboard, or in a digital tool. It is important and very useful during brainstorming sessions because it can help the team member visualize a broad range of design solution before deciding which one is the best option to go with.



Section 8.3: Wireframes

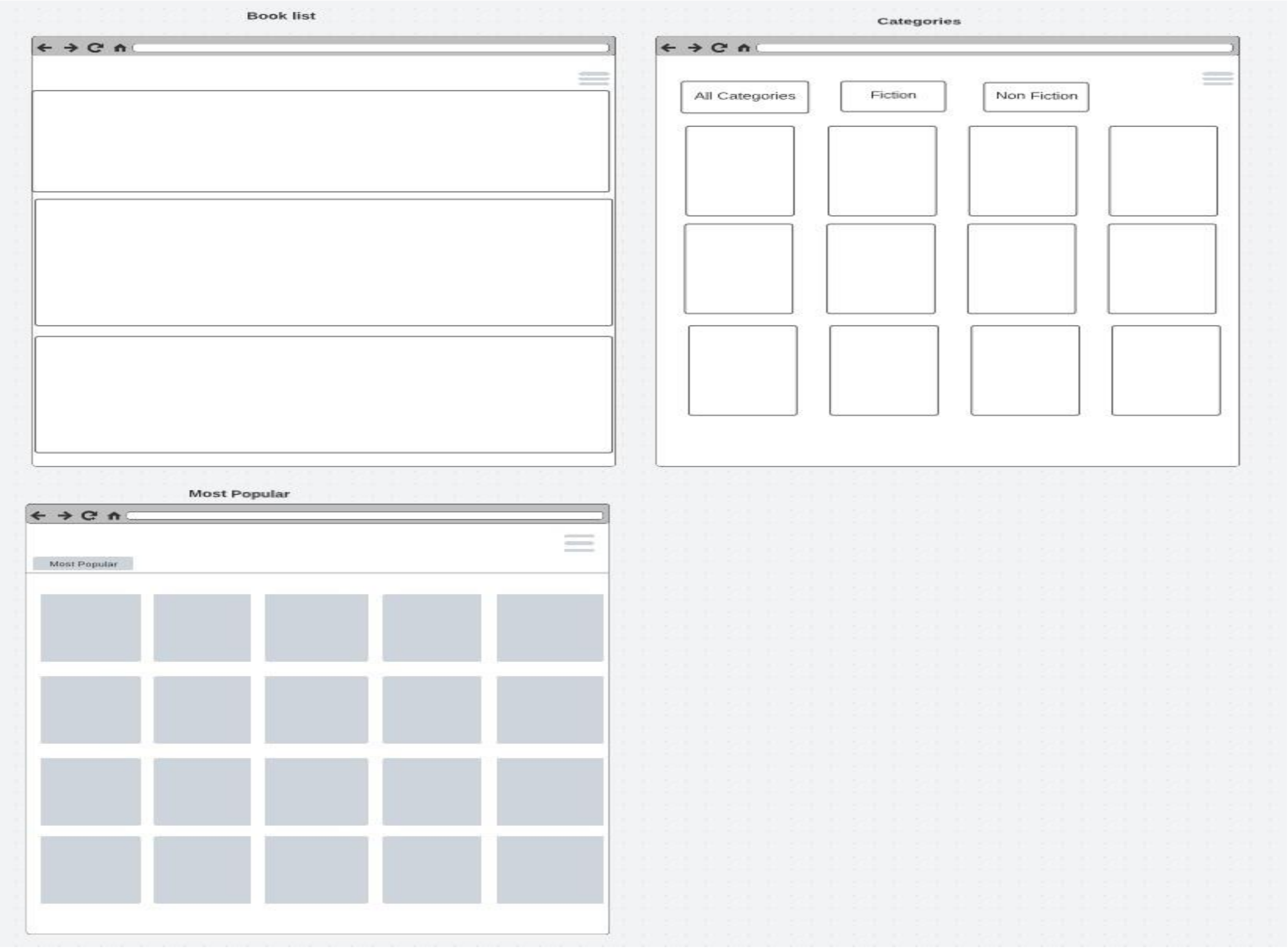
The wireframe is a tool that helps designers to visualize the basic structure of the future interface of the page, this is including the key elements and how they will fit together. The wireframes are usually used as a foundation for designers work on before prototyping.





Unfortunately, uploading books to the library is now only available for registred users with confirmed email address. Sorry for the inconvenience!

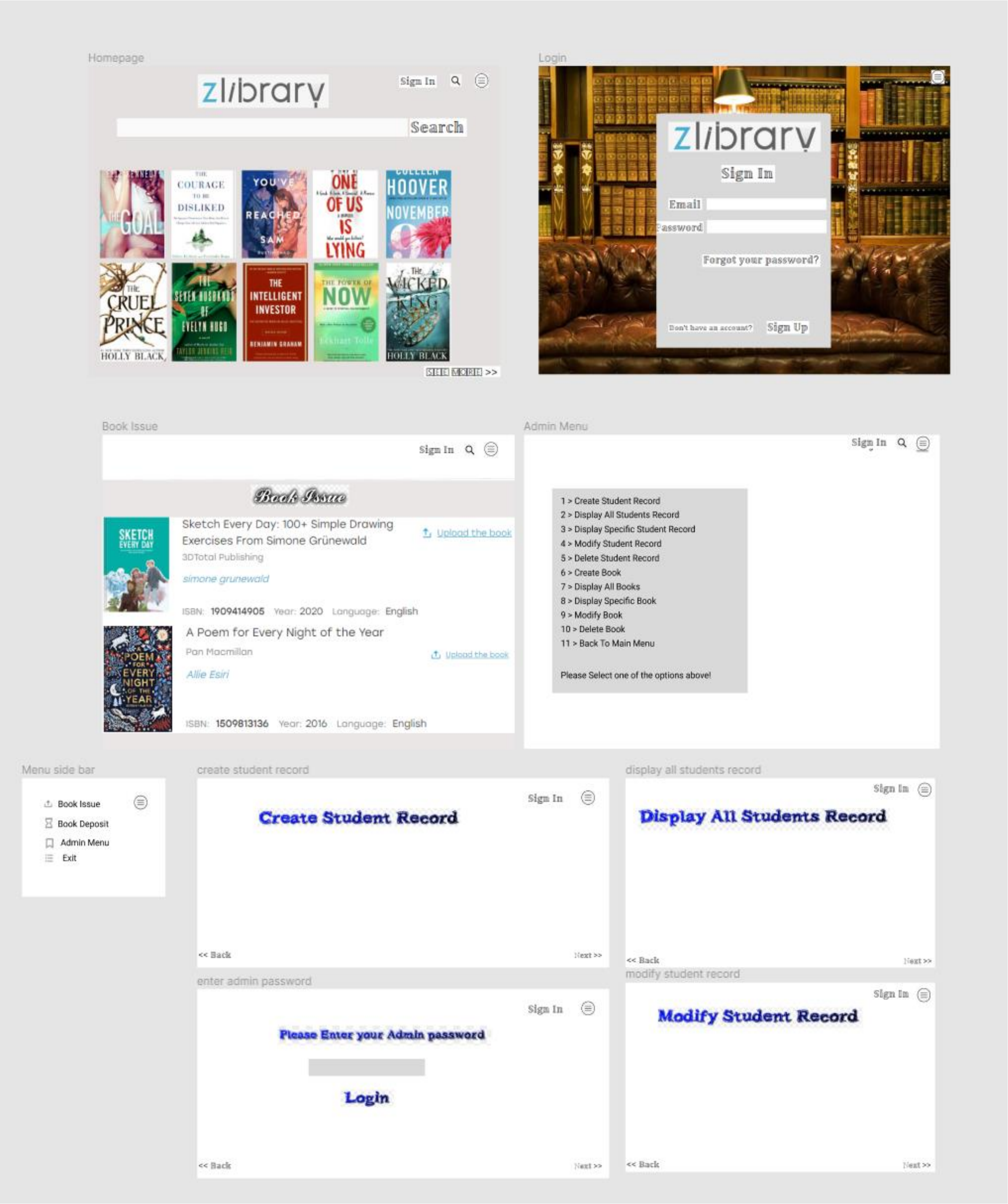
Here you can track the book you need or create a book request



Wireframe is the layout of the application; this wireframe shows what interface elements will exist on the pages and how they will fit into the application. The aim of the wireframe is providing a visual understanding of how the pages will work, it is also be used to create the main and secondary navigation ensuring the final design and structure for the application meets user requirements and expectations.

Section 8.4: Hi – Fi Prototype

If the wireframes are more about structure and visual hierarchy, prototypes are more about the actual interaction experience between the interface and the feeling of the application. We can assume that the prototype is a simulation of the product and may be low-fidelity (clickable wireframes) to hight-fidelity (code prototypes).

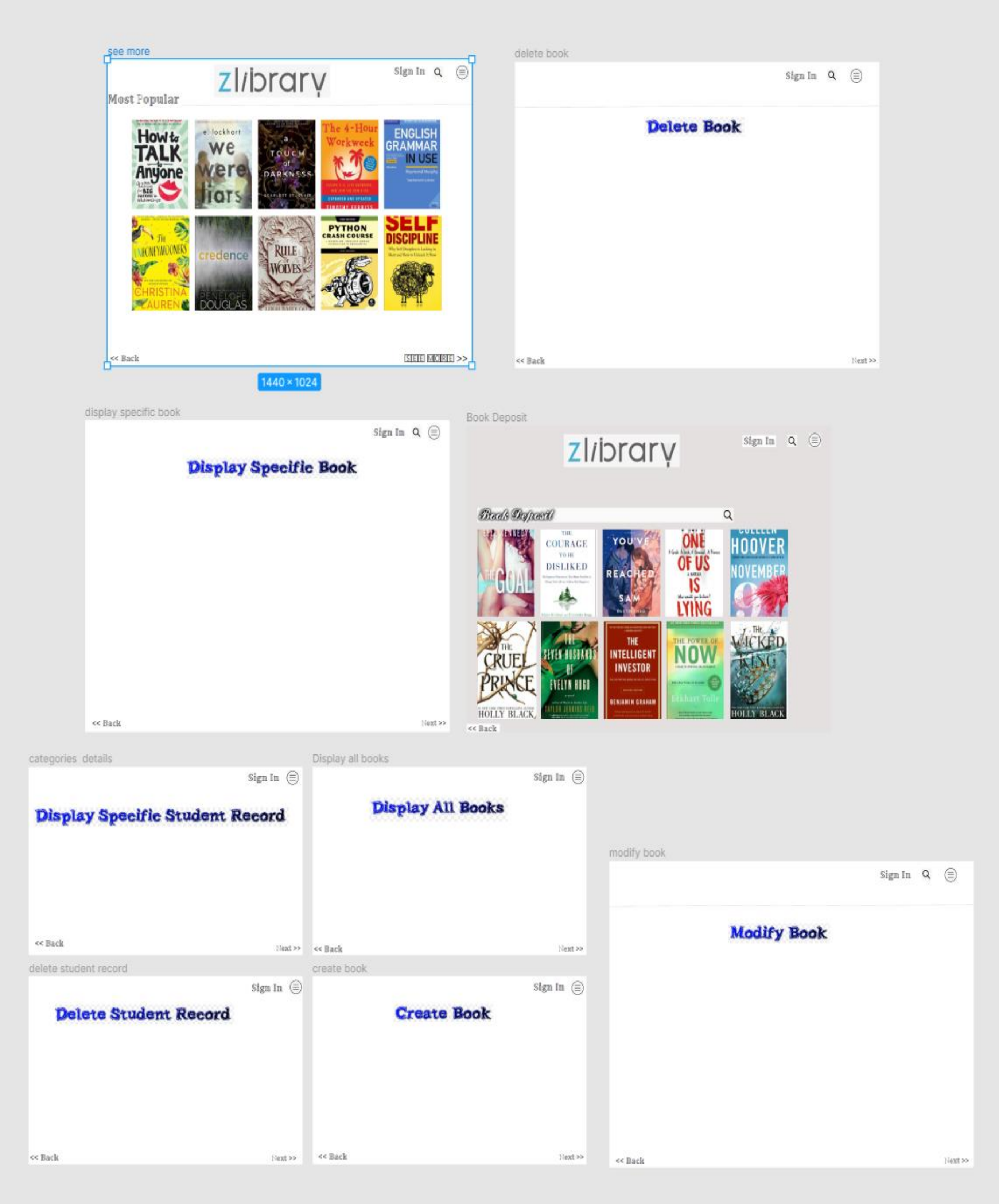


Sketch Every Day: 100+ Simple Drawing Exercises From Simone Grünewald
3DTotal Publishing
simone grunewald
ISBN: 1909414905 Year: 2020 Language: English

A Poem for Every Night of the Year
Pan Macmillan
Allie Esiri
ISBN: 1509813136 Year: 2016 Language: English

1 > Create Student Record
2 > Display All Students Record
3 > Display Specific Student Record
4 > Modify Student Record
5 > Delete Student Record
6 > Create Book
7 > Display All Books
8 > Display Specific Book
9 > Modify Book
10 > Delete Book
11 > Back To Main Menu

Please Select one of the options above!


see more
zlibrary
Sign In
Most Popular
<< Back
>>
1440 × 1024
delete book
Sign In
Delete Book
<< Back
Next >>
display specific book
Sign In
Display Specific Book
<< Back
Next >>
Book Deposit
zlibrary
Sign In
Book Deposit
<< Back
categories details
Sign In
Display Specific Student Record
<< Back
Next >>
Display all books
Sign In
Display All Books
<< Back
Next >>
modify book
Sign In
Modify Book
<< Back
Next >>
delete student record
Sign In
Delete Student Record
<< Back
Next >>
create book
Sign In
Create Book
<< Back
Next >>

Section 8.5: Testing

Testing phase is important because after this phase the project will be out into the real life for real user interact with it. This phase gives you the chance to see if your project reaches out the user’s requirements and business requirements, also it gives you a chance to make the last improvements before launch the end. The customers feedback even the negative one is very welcome in the testing phase; it gives you a chance to fixing the problems identified by the user and make improvements looking at the comments and suggestions by the customers feedback. In the Design Thinking process, the solution is tested by consumers in a real-world environment. During testing, the final user interacts with the prototype without being given any explicit instructions. It's an opportunity for the design thinker to watch how customers interact with a product and get their opinions on various features. The Design Thinking process is complete if the user is happy with the prototype. If the user is still not satisfied, you re-start the process while considering their comments.

The test cases for the Library Management System are an application that explains how the system works. Software testing is critical and important part that is involved in the overall development of the application. Quality assurance is the review of the software application and checks for the correctness, reliability, completeness, and maintainability.

Section 9: Usability testing

Usability testing is a technique used to evaluate a product or service. In the case of websites and software, it is used to verify that the system meets the user's real needs. If well applied, the result found will clearly show the difficulties and positive points of the consumers' experience.

Through usability testing, it is possible to identify failures or other problematic processes — such as navigation difficulties, slowness, or crashes. By identifying problems and making corrections possible, it largely contributes to delivering a better UX (User Experience), since navigation will be fluid and dynamic in the final product.

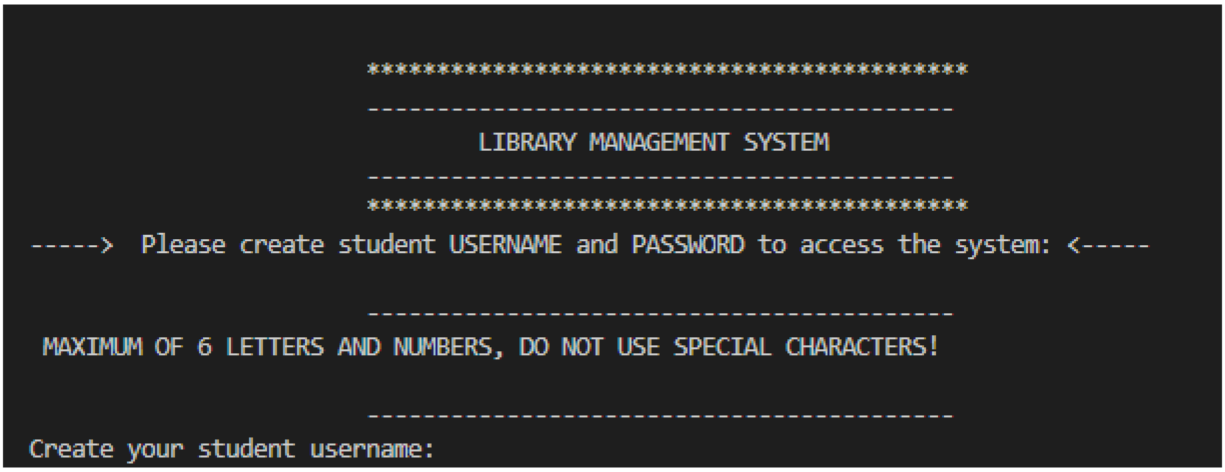
For start the usability testing into practice it is necessary to simulate routine actions of the website or software, to analyse the behaviour of users while using the system. So, the first step is to define the main objective of the test and the metrics that must be considered during the analysis: what you want to test and how the hypotheses will be validated.

Usability testing is a tool designed to ensure the consistency of a product by getting feedback directly from the end user. It’s also a way to prove the solidity of an interface by putting it to use in common situations. Allowing then, constant, and targeted improvements. In other words, it is a test that seeks to understand how the system behaves daily, in the naturalness of use, and if it meets the requirements thought and established.

It is important to note that this test must be done with real users (and not testers or designers), with a profile like the persona of the project in question. The intention is to receive deeper and more real insights than what the designers could have achieved in designing the system. It is through these tests that designers can decide about optimizing and cleaning the interface for the best user experience.

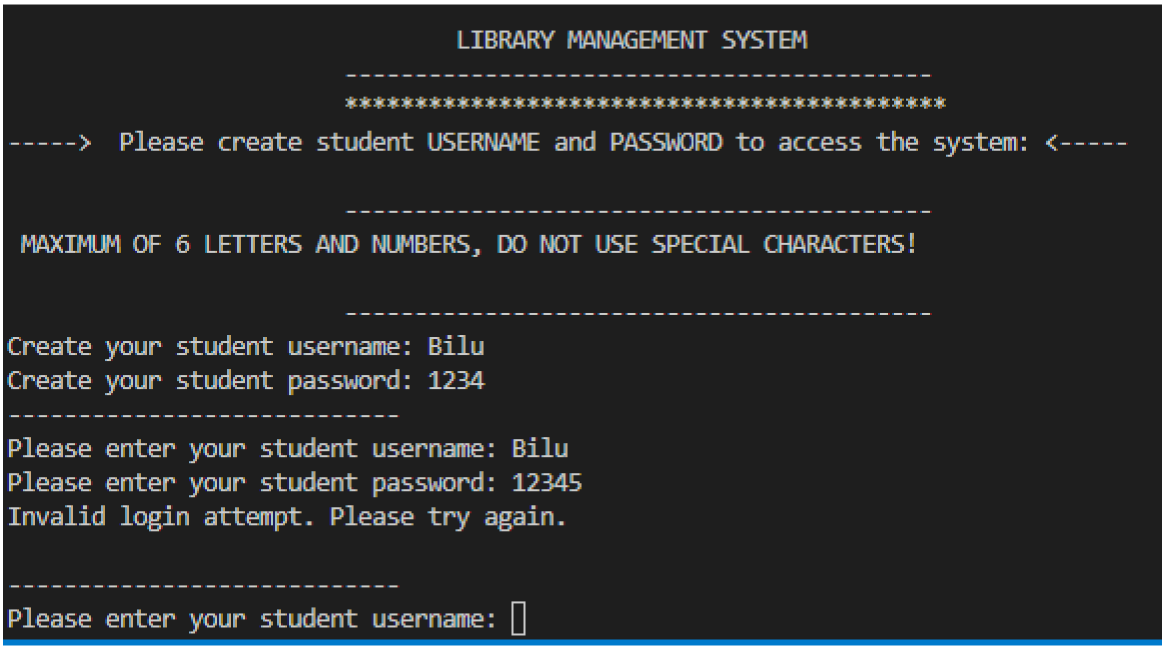
Section 9.1 Project Functionality testing

It is necessary to create student's username and password before accessing the system.



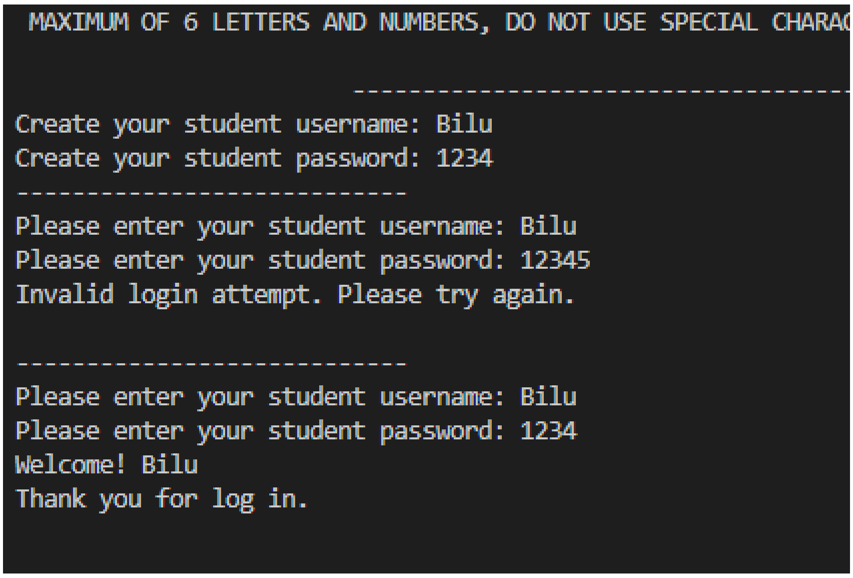
```
                        *******************************************
                        ------------------------------------------
                                LIBRARY MANAGEMENT SYSTEM
                        ------------------------------------------
                        *******************************************
----->  Please create student USERNAME and PASSWORD to access the system: <-----

                        ------------------------------------------
 MAXIMUM OF 6 LETTERS AND NUMBERS, DO NOT USE SPECIAL CHARACTERS!

                        ------------------------------------------
Create your student username:
```

If username and password does not match students won't be able to access the system

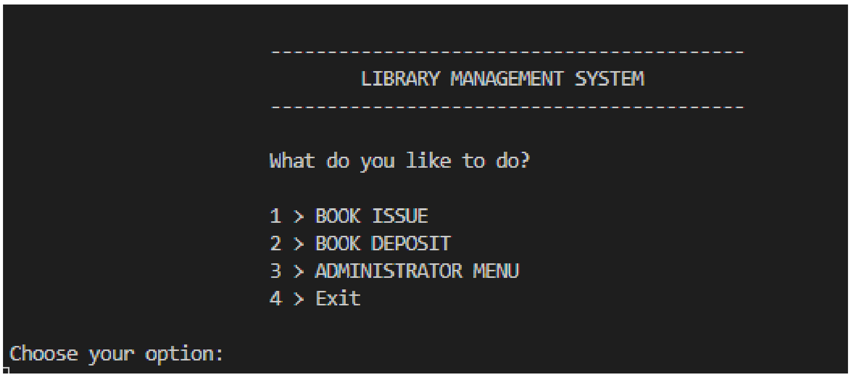


```
                                LIBRARY MANAGEMENT SYSTEM
                        ------------------------------------------
                        *******************************************
----->  Please create student USERNAME and PASSWORD to access the system: <-----

                        ------------------------------------------
 MAXIMUM OF 6 LETTERS AND NUMBERS, DO NOT USE SPECIAL CHARACTERS!

                        ------------------------------------------
Create your student username: Bilu
Create your student password: 1234
----------------------------
Please enter your student username: Bilu
Please enter your student password: 12345
Invalid login attempt. Please try again.

----------------------------
Please enter your student username: 
```

After students insert their correct username and password, they can access the system!

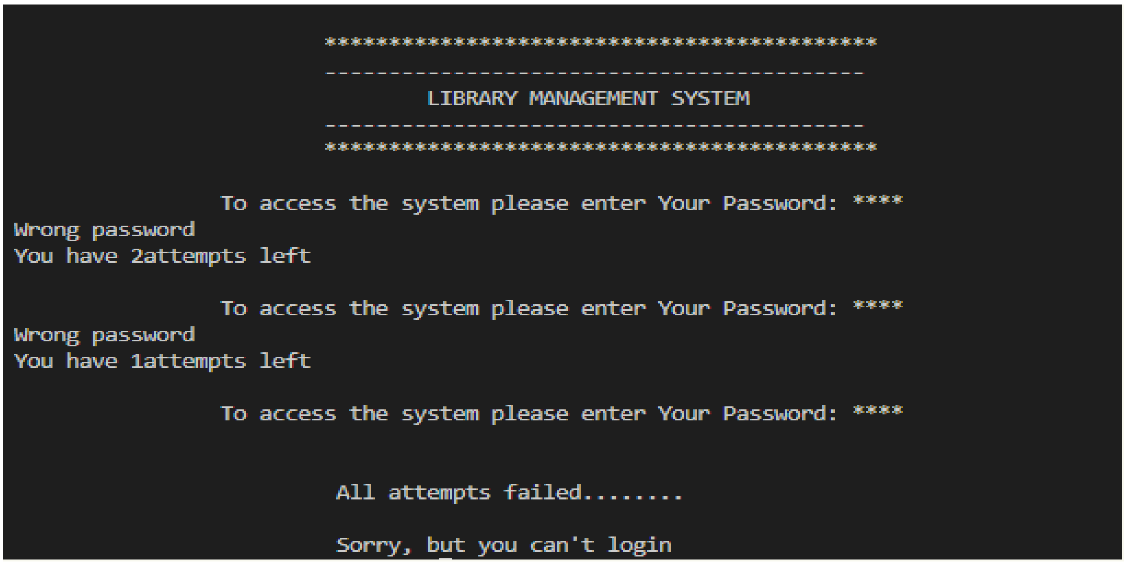


```
 MAXIMUM OF 6 LETTERS AND NUMBERS, DO NOT USE SPECIAL CHARAC

                        -----------------------------------
Create your student username: Bilu
Create your student password: 1234
----------------------------
Please enter your student username: Bilu
Please enter your student password: 12345
Invalid login attempt. Please try again.

----------------------------
Please enter your student username: Bilu
Please enter your student password: 1234
Welcome! Bilu
Thank you for log in.
```



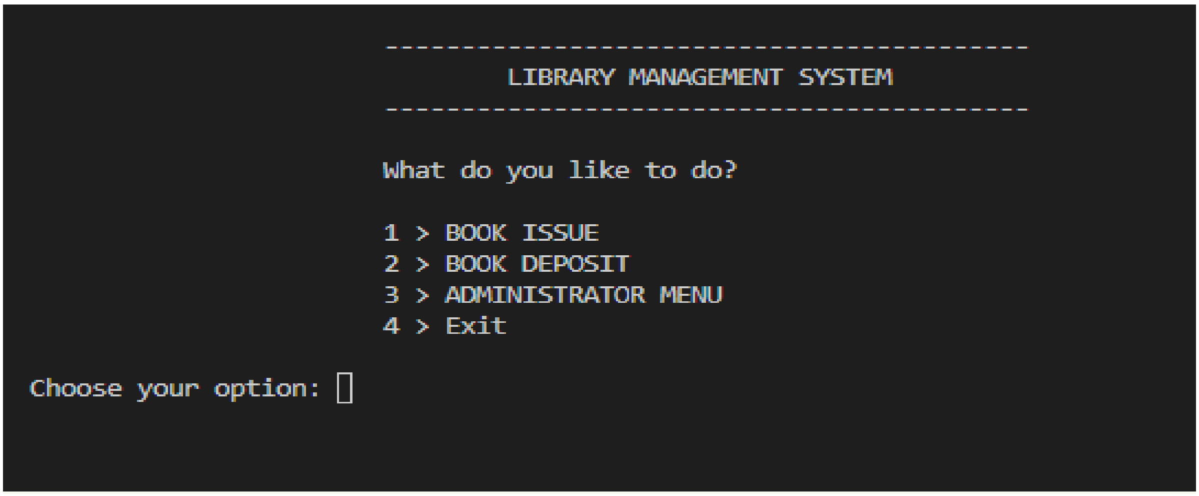
```
                       -----------------------------------------
                               LIBRARY MANAGEMENT SYSTEM
                       -----------------------------------------

                       What do you like to do?

                       1 > BOOK ISSUE
                       2 > BOOK DEPOSIT
                       3 > ADMINISTRATOR MENU
                       4 > Exit

Choose your option:
```

Testing for user's login into the system, trying different characters and reaching the limit of attempts.



```
                         *******************************************
                         ------------------------------------------
                                 LIBRARY MANAGEMENT SYSTEM
                         ------------------------------------------
                         *******************************************

                To access the system please enter Your Password: ****
Wrong password
You have 2attempts left

                To access the system please enter Your Password: ****
Wrong password
You have 1attempts left

                To access the system please enter Your Password: ****


                          All attempts failed........

                          Sorry, but you can't login
```

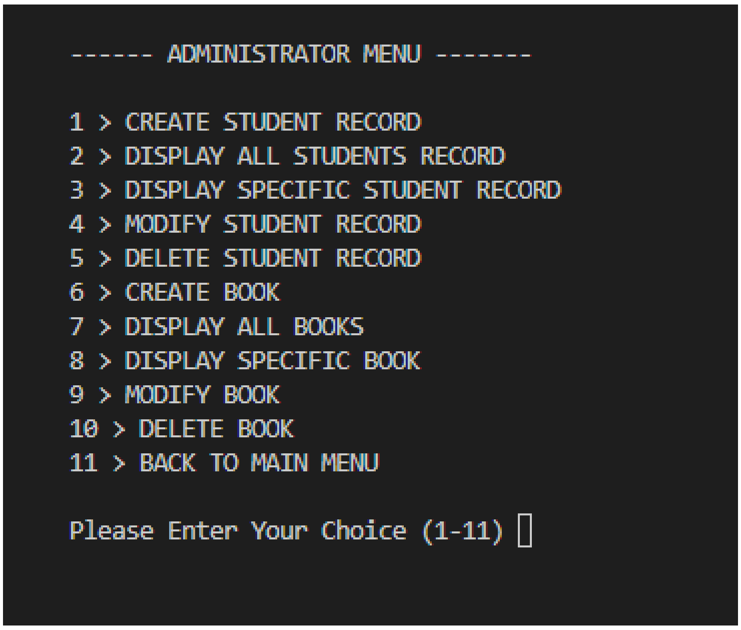
Login using the right password which correspond to “0000”



```
                         -----------------------------------------
                                 LIBRARY MANAGEMENT SYSTEM
                         -----------------------------------------

                         What do you like to do?

                         1 > BOOK ISSUE
                         2 > BOOK DEPOSIT
                         3 > ADMINISTRATOR MENU
                         4 > Exit

 Choose your option:
```

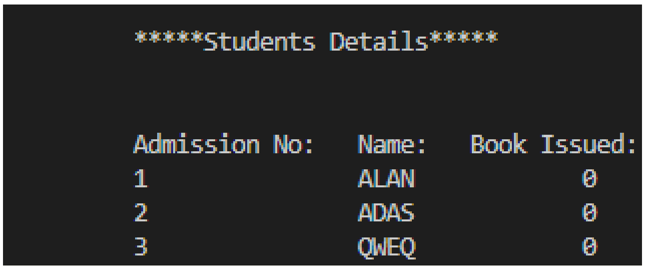
Testing the ADMIN option number 3: The admin access contains specific access permitted only by administrators and their password.

Admin MENU:



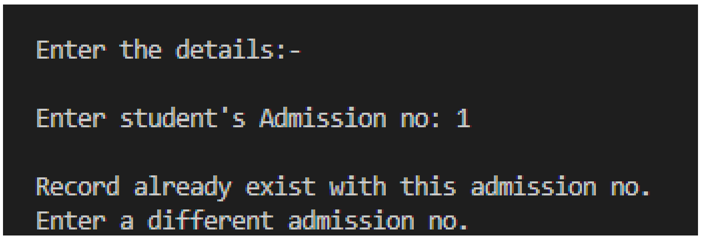
```
    ------ ADMINISTRATOR MENU -------

    1 > CREATE STUDENT RECORD
    2 > DISPLAY ALL STUDENTS RECORD
    3 > DISPLAY SPECIFIC STUDENT RECORD
    4 > MODIFY STUDENT RECORD
    5 > DELETE STUDENT RECORD
    6 > CREATE BOOK
    7 > DISPLAY ALL BOOKS
    8 > DISPLAY SPECIFIC BOOK
    9 > MODIFY BOOK
    10 > DELETE BOOK
    11 > BACK TO MAIN MENU

    Please Enter Your Choice (1-11)
```

Adding new data for students’ option (1) and displaying the students data option (2):



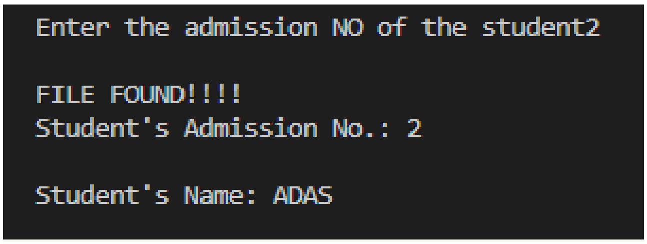
```
        *****Students Details*****


        Admission No:   Name:   Book Issued:
        1               ALAN               0
        2               ADAS               0
        3               QWEQ               0
```

Testing students record that already exist in the system:

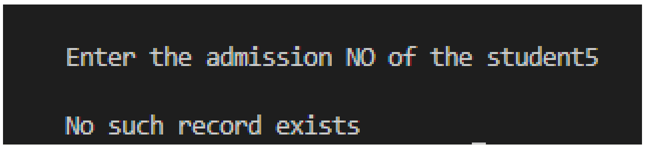


```
Enter the details:-

Enter student's Admission no: 1

Record already exist with this admission no.
Enter a different admission no.
```

Testing option (3) displaying a specific student record. Comparing existent student and non-existent student:

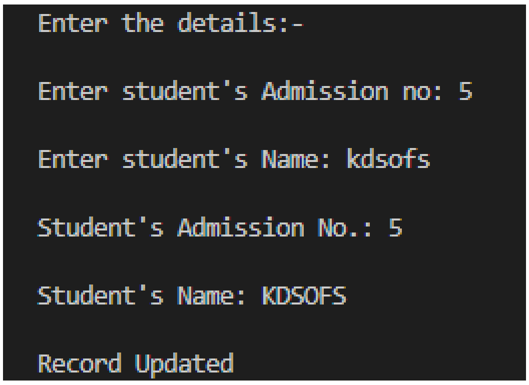


```
Enter the admission NO of the student2

FILE FOUND!!!!
Student's Admission No.: 2

Student's Name: ADAS
```



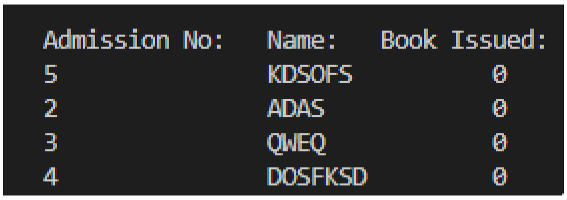
```
  Enter the admission NO of the student5

  No such record exists
```

Testing updating student admission 1 and change to a new name and number:



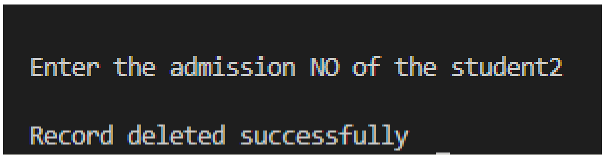
```
Enter the details:-

Enter student's Admission no: 5

Enter student's Name: kdsofs

Student's Admission No.: 5

Student's Name: KDSOFS

Record Updated
```

Displaying a new student list after updating the existing one:

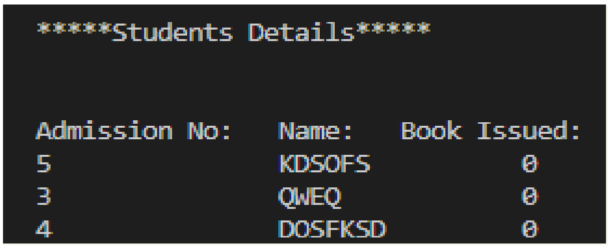


```
Admission No:    Name:   Book Issued:
5                KDSOFS         0
2                ADAS           0
3                QWEQ           0
4                DOSFKSD        0
```

Testing deleting student record. The number of the student data is 2:

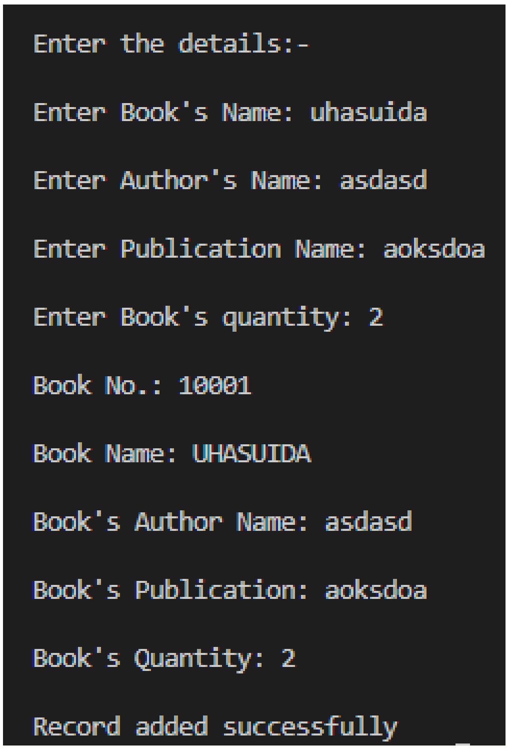


```
Enter the admission NO of the student2

Record deleted successfully _
```

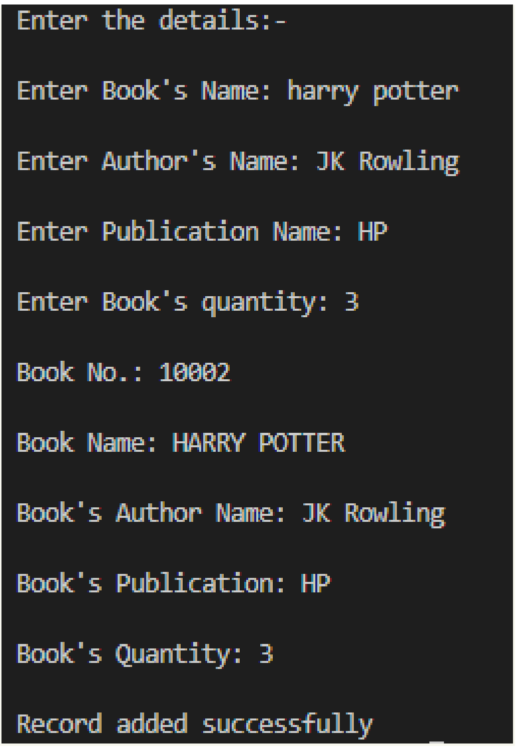


```
*****Students Details*****


Admission No:    Name:    Book Issued:
5                KDSOFS          0
3                QWEQ            0
4                DOSFKSD         0
```

Testing adding two new book’s data:

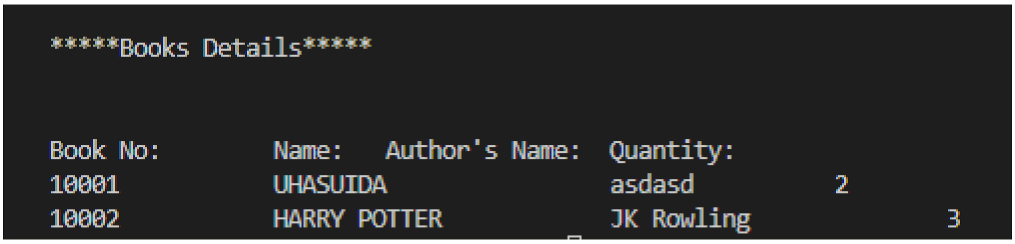


```
Enter the details:-

Enter Book's Name: uhasuida

Enter Author's Name: asdasd

Enter Publication Name: aoksdoa

Enter Book's quantity: 2

Book No.: 10001

Book Name: UHASUIDA

Book's Author Name: asdasd

Book's Publication: aoksdoa

Book's Quantity: 2

Record added successfully
```



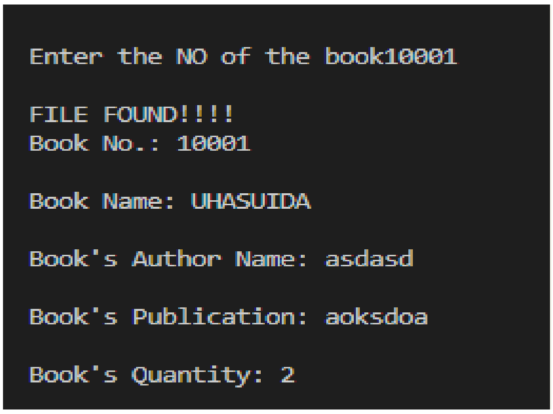
```
Enter the details:-

Enter Book's Name: harry potter

Enter Author's Name: JK Rowling

Enter Publication Name: HP

Enter Book's quantity: 3

Book No.: 10002

Book Name: HARRY POTTER

Book's Author Name: JK Rowling

Book's Publication: HP

Book's Quantity: 3

Record added successfully
```

Displaying the new books created:



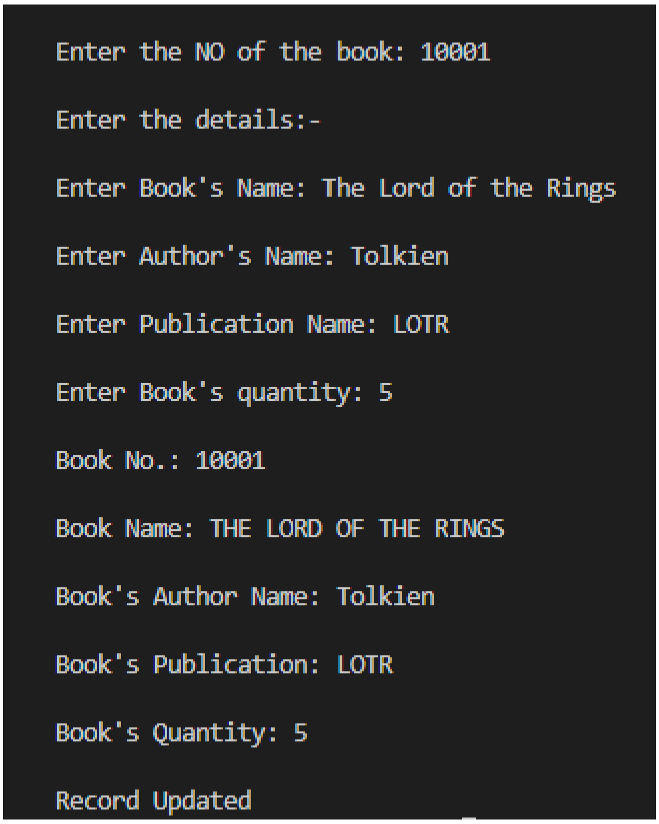
```
*****Books Details*****


Book No:        Name:   Author's Name:  Quantity:
10001           UHASUIDA                asdasd          2
10002           HARRY POTTER            JK Rowling              3
```

Displaying a specific book requested:

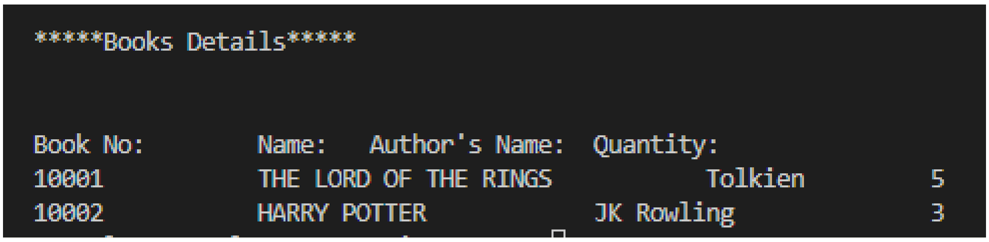


```
Enter the NO of the book10001

FILE FOUND!!!!
Book No.: 10001

Book Name: UHASUIDA

Book's Author Name: asdasd

Book's Publication: aoksdoa

Book's Quantity: 2
```

Updating book number 10001



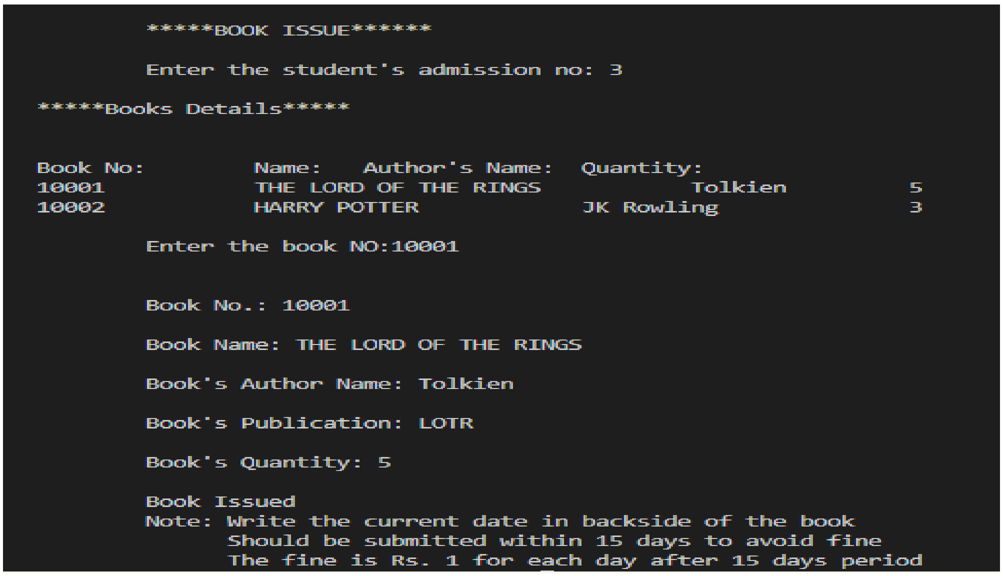
```
Enter the NO of the book: 10001

Enter the details:-

Enter Book's Name: The Lord of the Rings

Enter Author's Name: Tolkien

Enter Publication Name: LOTR

Enter Book's quantity: 5

Book No.: 10001

Book Name: THE LORD OF THE RINGS

Book's Author Name: Tolkien

Book's Publication: LOTR

Book's Quantity: 5

Record Updated
```



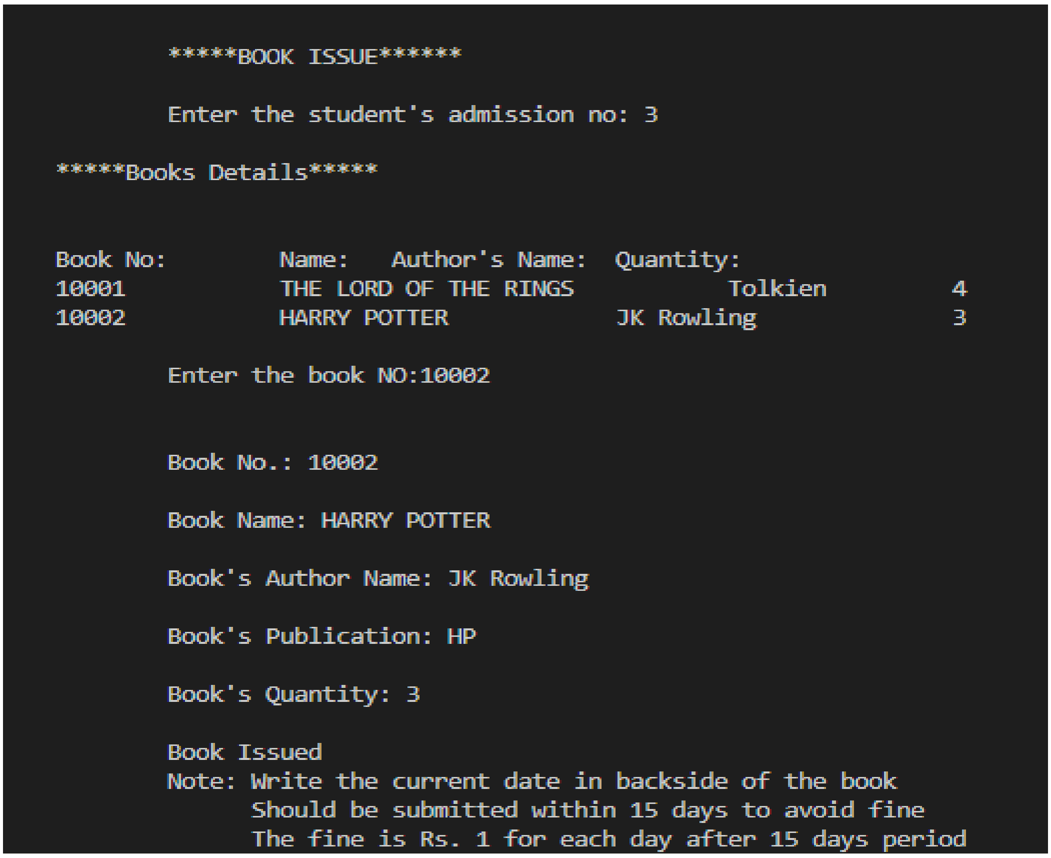
```
*****Books Details*****


Book No:        Name:   Author's Name:  Quantity:
10001           THE LORD OF THE RINGS           Tolkien         5
10002           HARRY POTTER            JK Rowling              3
```

Testing user interface

Book issue:

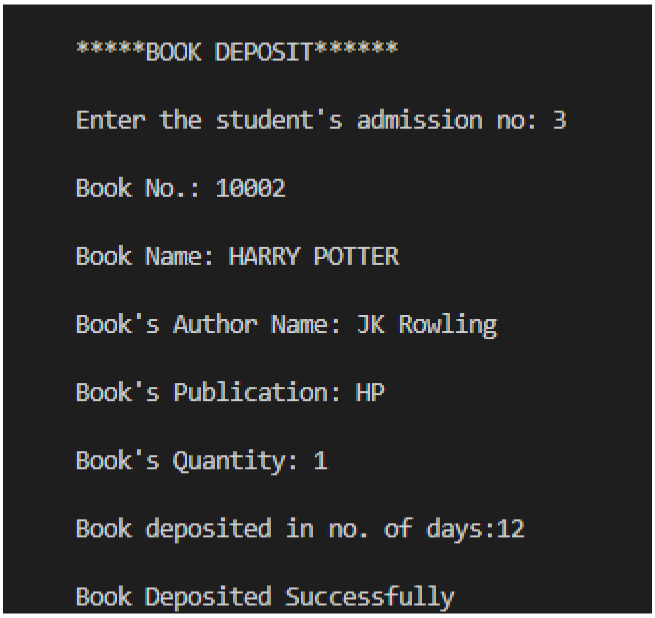


```
        *****BOOK ISSUE******

        Enter the student's admission no: 3

  *****Books Details*****


  Book No:        Name:   Author's Name:  Quantity:
  10001           THE LORD OF THE RINGS           Tolkien           5
  10002           HARRY POTTER            JK Rowling                3

        Enter the book NO:10001


        Book No.: 10001

        Book Name: THE LORD OF THE RINGS

        Book's Author Name: Tolkien

        Book's Publication: LOTR

        Book's Quantity: 5

        Book Issued
        Note: Write the current date in backside of the book
              Should be submitted within 15 days to avoid fine
              The fine is Rs. 1 for each day after 15 days period
```



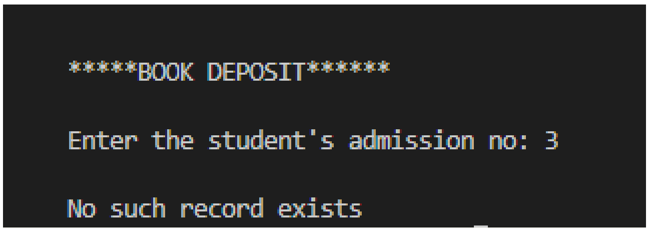
```
        *****BOOK ISSUE******

        Enter the student's admission no: 3

  *****Books Details*****


  Book No:        Name:   Author's Name:  Quantity:
  10001           THE LORD OF THE RINGS           Tolkien           4
  10002           HARRY POTTER            JK Rowling                3

        Enter the book NO:10002


        Book No.: 10002

        Book Name: HARRY POTTER

        Book's Author Name: JK Rowling

        Book's Publication: HP

        Book's Quantity: 3

        Book Issued
        Note: Write the current date in backside of the book
              Should be submitted within 15 days to avoid fine
              The fine is Rs. 1 for each day after 15 days period
```

Testing book return:



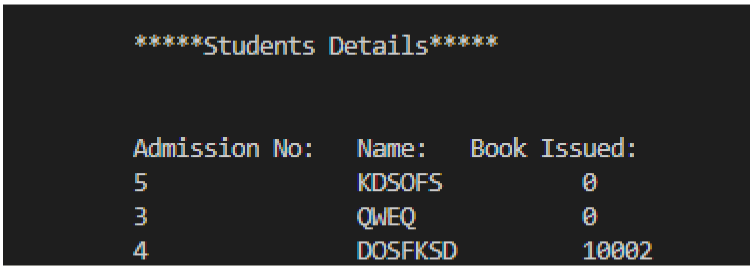
```
    *****BOOK DEPOSIT******

    Enter the student's admission no: 3

    Book No.: 10002

    Book Name: HARRY POTTER

    Book's Author Name: JK Rowling

    Book's Publication: HP

    Book's Quantity: 1

    Book deposited in no. of days:12

    Book Deposited Successfully
```

Trying make the request on a different student ID:



```
    *****BOOK DEPOSIT******

    Enter the student's admission no: 3

    No such record exists
```

Testing displaying the student’s information on admin menu with the updated books.

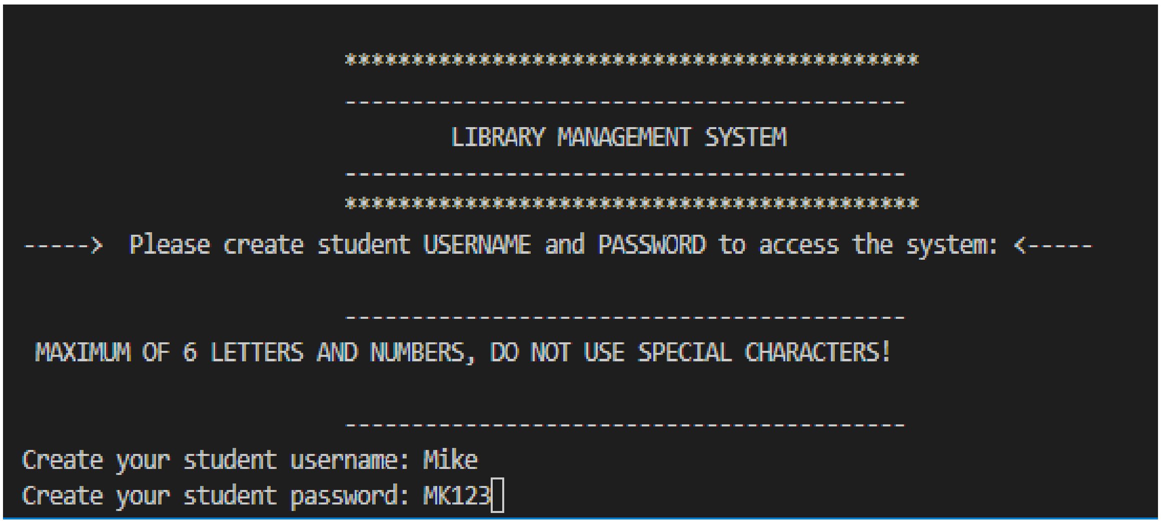


```
        *****Students Details*****


        Admission No:    Name:    Book Issued:
        5                KDSOFS         0
        3                QWEQ           0
        4                DOSFKSD        10002
```

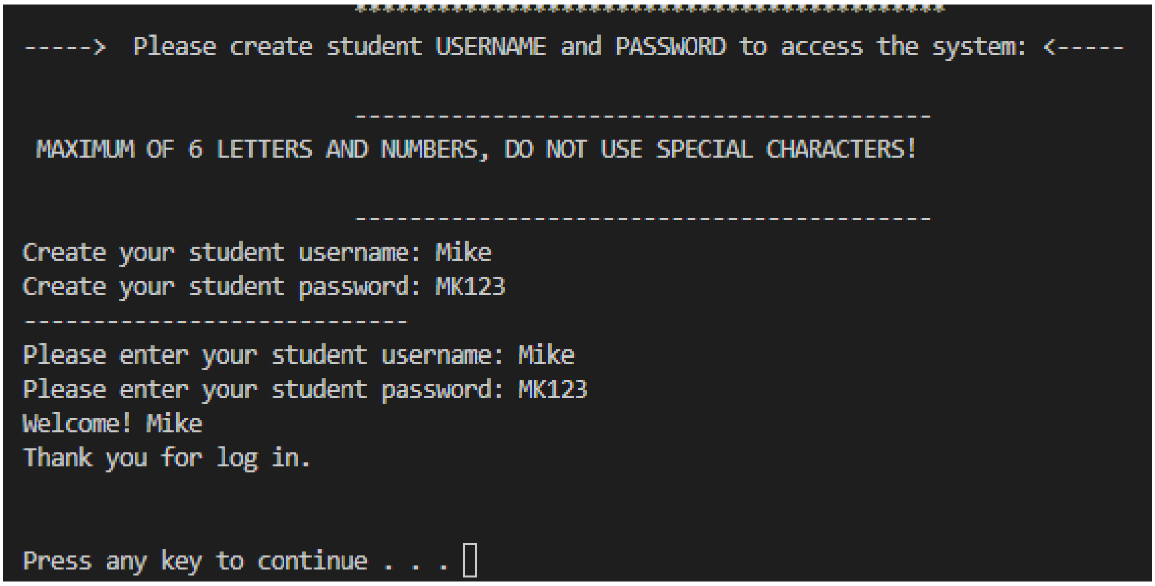
Section 9.2: User documentation

There are two main types of users that will be accessing the library management system, they are: The admin, which is the responsible for adding, deleting, updating or input information on the system. The data can be from the students or books, The other user is the student or librarian responsible to manage the information about the books selected.

The step by step consists of creating username and password to logging into the system as the first process \ step. If the password requested has been inputted wrong 3 times the system will shut down to make sure anyone that shouldn't is using the system. To make sure students or users can access the system the created username and password need to be matching.

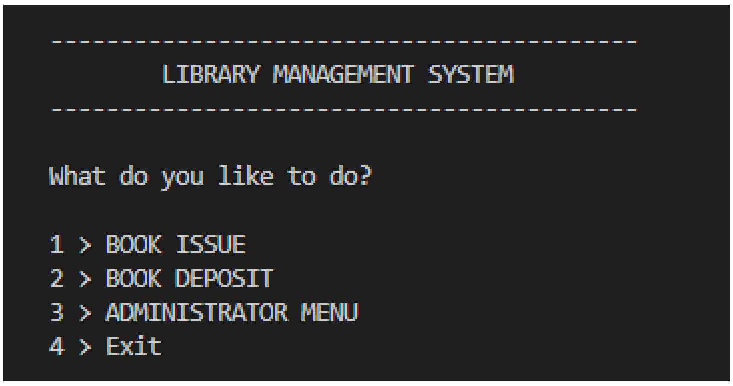


```
                    *****************************************
                    ----------------------------------------
                           LIBRARY MANAGEMENT SYSTEM
                    ----------------------------------------
                    *****************************************
---->  Please create student USERNAME and PASSWORD to access the system: <-----

                    ----------------------------------------
 MAXIMUM OF 6 LETTERS AND NUMBERS, DO NOT USE SPECIAL CHARACTERS!

                    ----------------------------------------
Create your student username: Mike
Create your student password: MK123
```



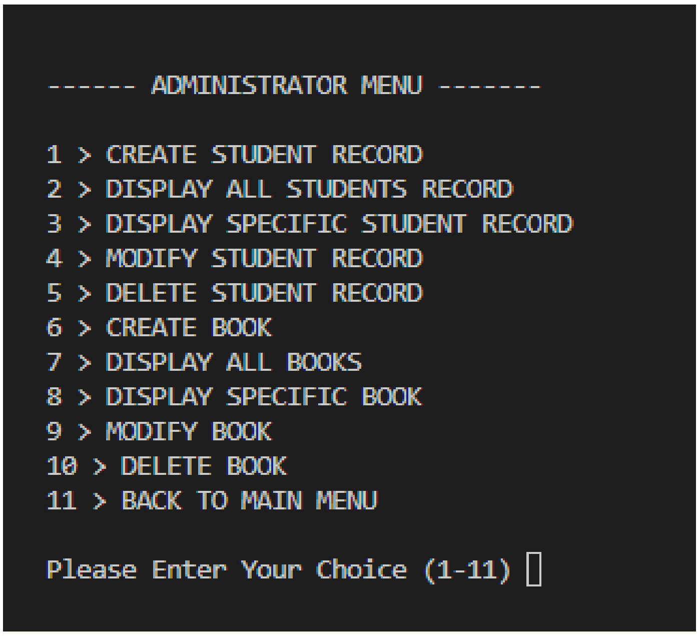
```
                    *****************************************
---->  Please create student USERNAME and PASSWORD to access the system: <-----

                    ----------------------------------------
 MAXIMUM OF 6 LETTERS AND NUMBERS, DO NOT USE SPECIAL CHARACTERS!

                    ----------------------------------------
Create your student username: Mike
Create your student password: MK123
---------------------------
Please enter your student username: Mike
Please enter your student password: MK123
Welcome! Mike
Thank you for log in.


Press any key to continue . . .
```

Second step is choosing an option from the menu or login as an administrator of the system. Note that the data need to be inserted before the you look for any information, the librarian option will be failed if there is no data inserted into the system. For testers we suggest login as administrator and add the data.



```
  -----------------------------------------
         LIBRARY MANAGEMENT SYSTEM
  -----------------------------------------

  What do you like to do?

  1 > BOOK ISSUE
  2 > BOOK DEPOSIT
  3 > ADMINISTRATOR MENU
  4 > Exit
```

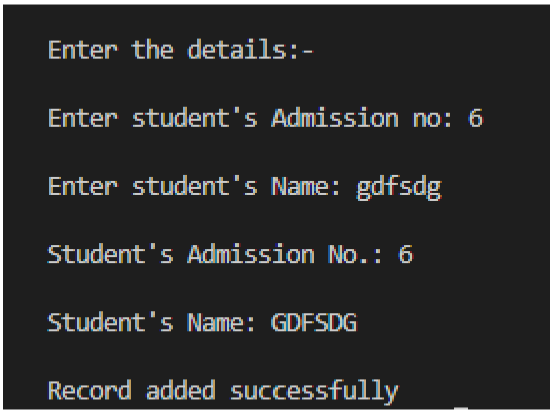
The librarian has the option 1 to lend books (if the quantity allows) to registered students. And can use the second option to issue information about the returned books. The admin has total access to the database of students and books, in this case he can choose to add, remove, or update information about one or all subjects.



```
  ------ ADMINISTRATOR MENU -------

  1 > CREATE STUDENT RECORD
  2 > DISPLAY ALL STUDENTS RECORD
  3 > DISPLAY SPECIFIC STUDENT RECORD
  4 > MODIFY STUDENT RECORD
  5 > DELETE STUDENT RECORD
  6 > CREATE BOOK
  7 > DISPLAY ALL BOOKS
  8 > DISPLAY SPECIFIC BOOK
  9 > MODIFY BOOK
  10 > DELETE BOOK
  11 > BACK TO MAIN MENU

  Please Enter Your Choice (1-11)
```

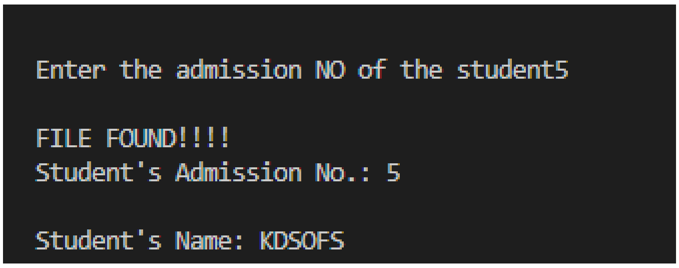
The options 1 to 5 regards to students and 6 to 10 to books.

To add a student, the admin should enter a non-existent id and then his name. For books, he must enter the name, the author, and the press name.



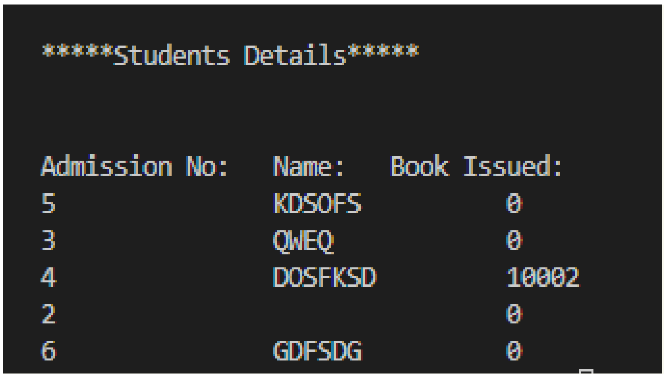
```
Enter the details:-

Enter student's Admission no: 6

Enter student's Name: gdfsdg

Student's Admission No.: 6

Student's Name: GDFSDG

Record added successfully
```

To remove and/or update, the admin must enter an existent and valid id, in this case the system will display the information about the student/book.



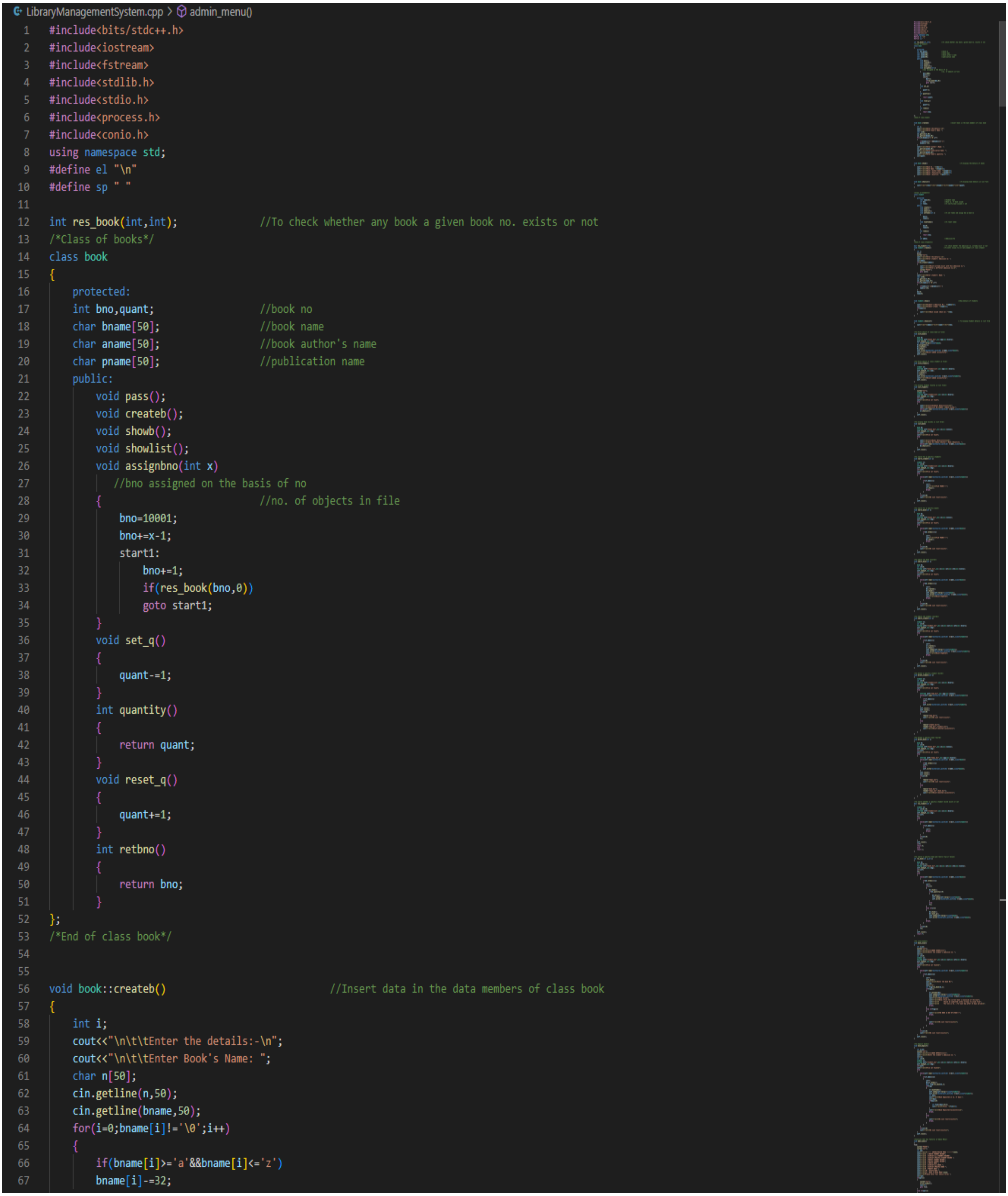
```
Enter the admission NO of the student5

FILE FOUND!!!!
Student's Admission No.: 5

Student's Name: KDSOFS
```

Choosing 2 or 7, will display all information about current student/books in the system, in this case all insertions, removal or updates will be displayed as up to date.

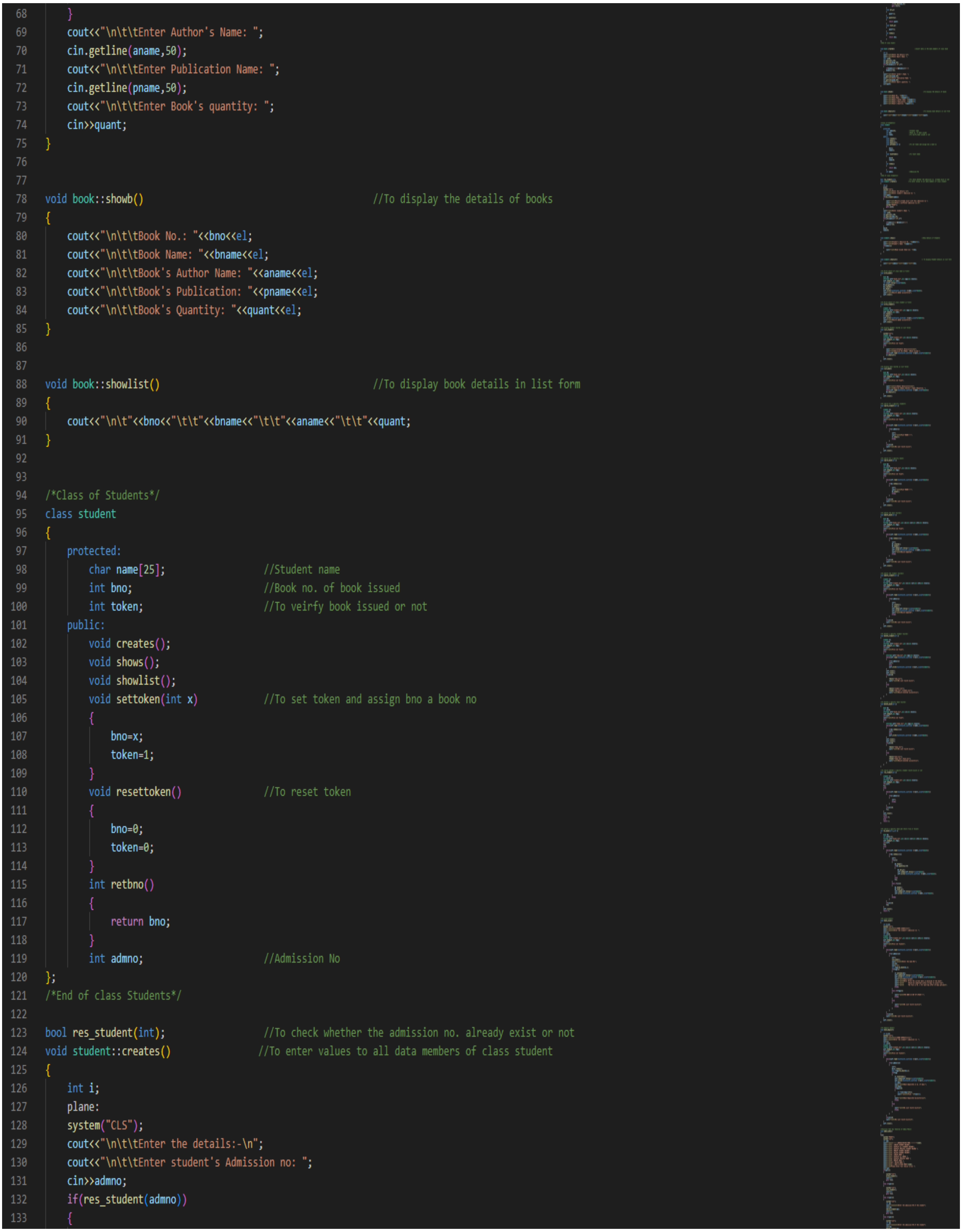


```
*****Students Details*****


Admission No:    Name:   Book Issued:
5                KDSOFS          0
3                QWEQ            0
4                DOSFKSD         10002
2                                0
6                GDFSDG          0
```

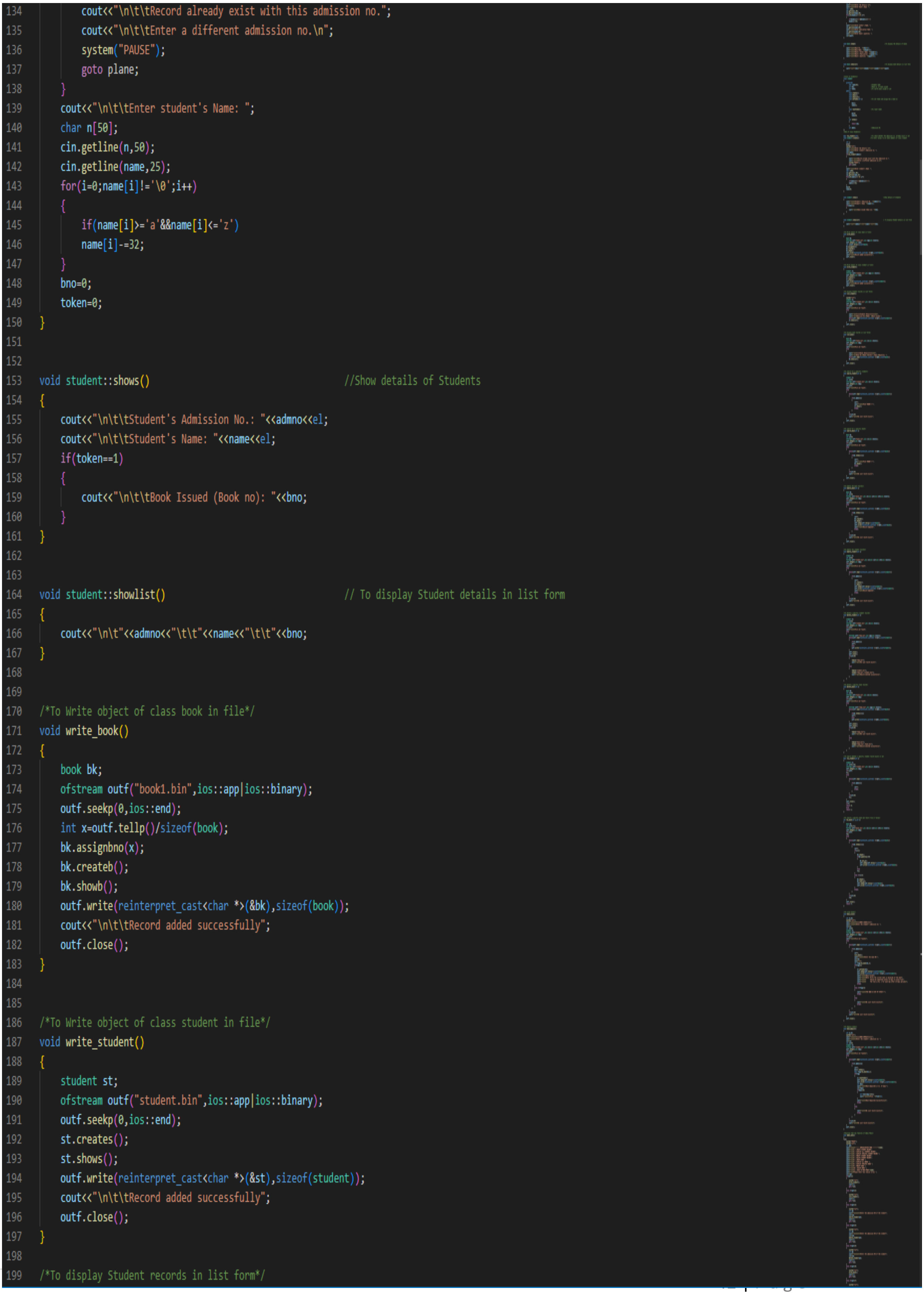
Section 9.3: Library Management System Application



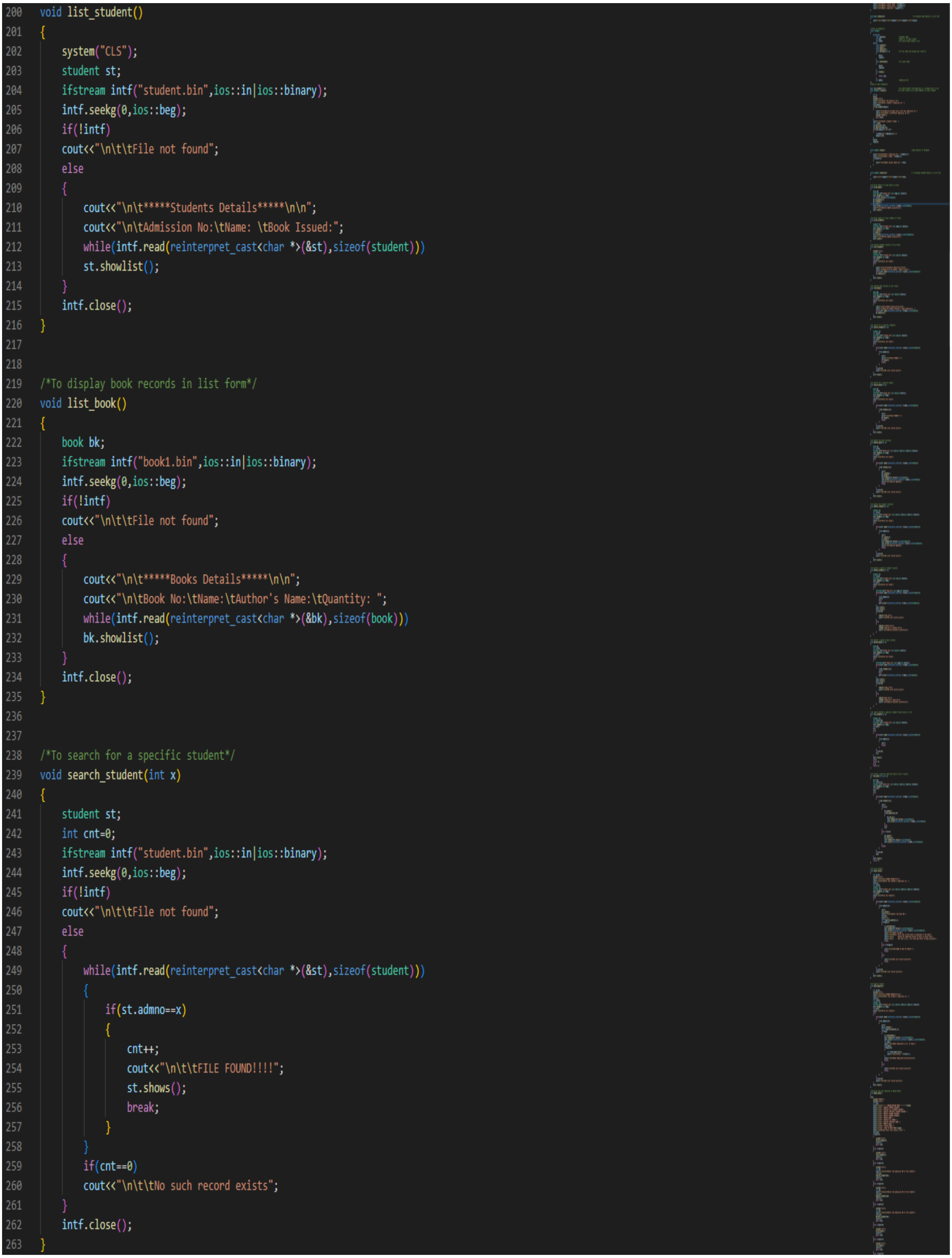
```cpp
#include<bits/stdc++.h>
#include<iostream>
#include<fstream>
#include<stdlib.h>
#include<stdio.h>
#include<process.h>
#include<conio.h>
using namespace std;
#define el "\n"
#define sp " "

int res_book(int,int);              //To check whether any book a given book no. exists or not
/*Class of books*/
class book
{
    protected:
    int bno,quant;                  //book no
    char bname[50];                 //book name
    char aname[50];                 //book author's name
    char pname[50];                 //publication name
    public:
        void pass();
        void createb();
        void showb();
        void showlist();
        void assignbno(int x)
           //bno assigned on the basis of no
        {                           //no. of objects in file
            bno=10001;
            bno+=x-1;
            start1:
                bno+=1;
                if(res_book(bno,0))
                goto start1;
        }
        void set_q()
        {
            quant-=1;
        }
        int quantity()
        {
            return quant;
        }
        void reset_q()
        {
            quant+=1;
        }
        int retbno()
        {
            return bno;
        }
};
/*End of class book*/


void book::createb()                            //Insert data in the data members of class book
{
    int i;
    cout<<"\n\t\tEnter the details:-\n";
    cout<<"\n\t\tEnter Book's Name: ";
    char n[50];
    cin.getline(n,50);
    cin.getline(bname,50);
    for(i=0;bname[i]!='\0';i++)
    {
        if(bname[i]>='a'&&bname[i]<='z')
        bname[i]-=32;
```



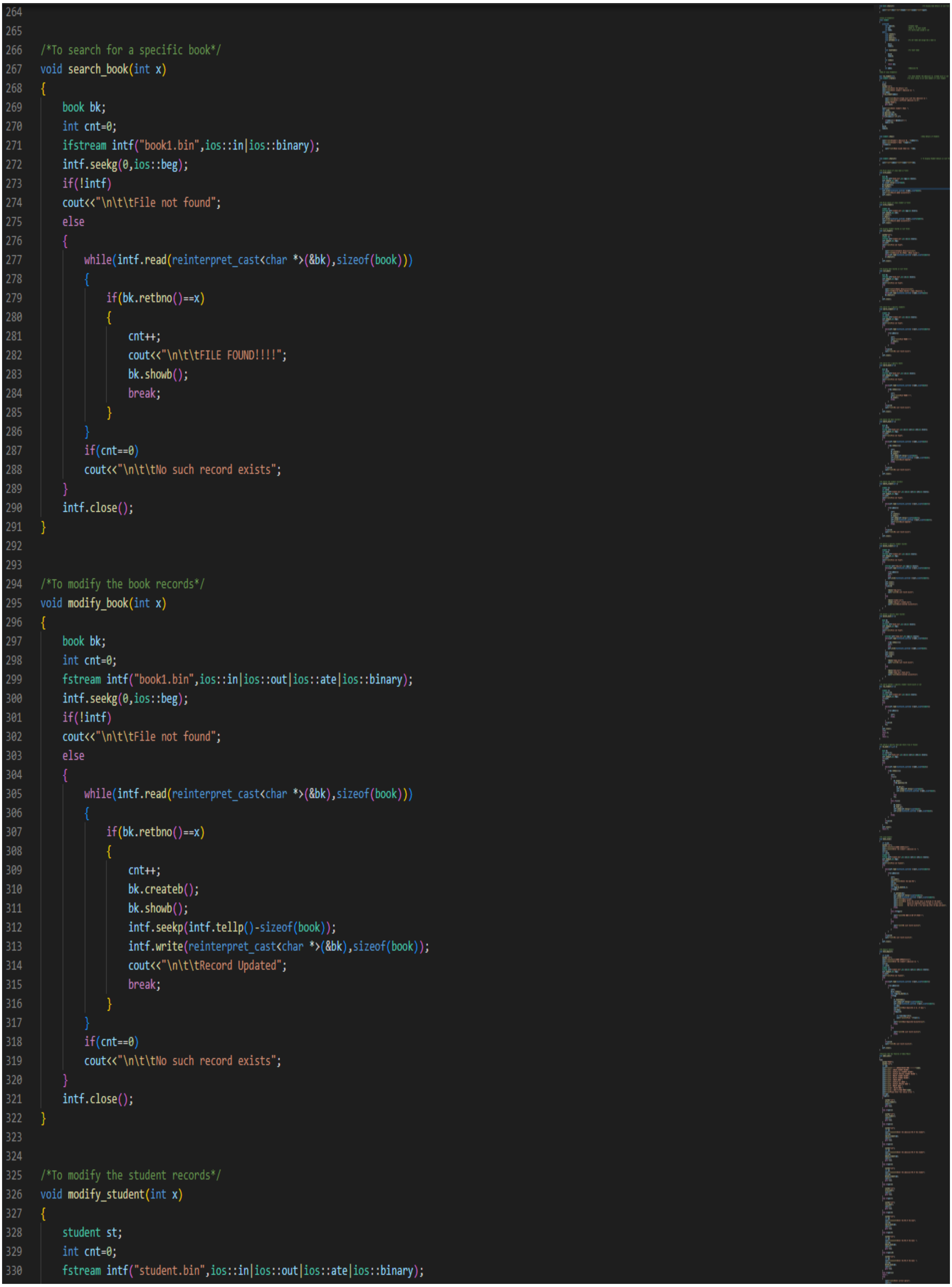
```
    }
    cout<<"\n\t\tEnter Author's Name: ";
    cin.getline(aname,50);
    cout<<"\n\t\tEnter Publication Name: ";
    cin.getline(pname,50);
    cout<<"\n\t\tEnter Book's quantity: ";
    cin>>quant;
}


void book::showb()                                      //To display the details of books
{
    cout<<"\n\t\tBook No.: "<<bno<<el;
    cout<<"\n\t\tBook Name: "<<bname<<el;
    cout<<"\n\t\tBook's Author Name: "<<aname<<el;
    cout<<"\n\t\tBook's Publication: "<<pname<<el;
    cout<<"\n\t\tBook's Quantity: "<<quant<<el;
}


void book::showlist()                                   //To display book details in list form
{
    cout<<"\n\t"<<bno<<"\t\t"<<bname<<"\t\t"<<aname<<"\t\t"<<quant;
}


/*Class of Students*/
class student
{
    protected:
        char name[25];                  //Student name
        int bno;                        //Book no. of book issued
        int token;                      //To veirfy book issued or not
    public:
        void creates();
        void shows();
        void showlist();
        void settoken(int x)            //To set token and assign bno a book no
        {
            bno=x;
            token=1;
        }
        void resettoken()               //To reset token
        {
            bno=0;
            token=0;
        }
        int retbno()
        {
            return bno;
        }
        int admno;                      //Admission No
};
/*End of class Students*/

bool res_student(int);                  //To check whether the admission no. already exist or not
void student::creates()                //To enter values to all data members of class student
{
    int i;
    plane:
    system("CLS");
    cout<<"\n\t\tEnter the details:-\n";
    cout<<"\n\t\tEnter student's Admission no: ";
    cin>>admno;
    if(res_student(admno))
    {
```



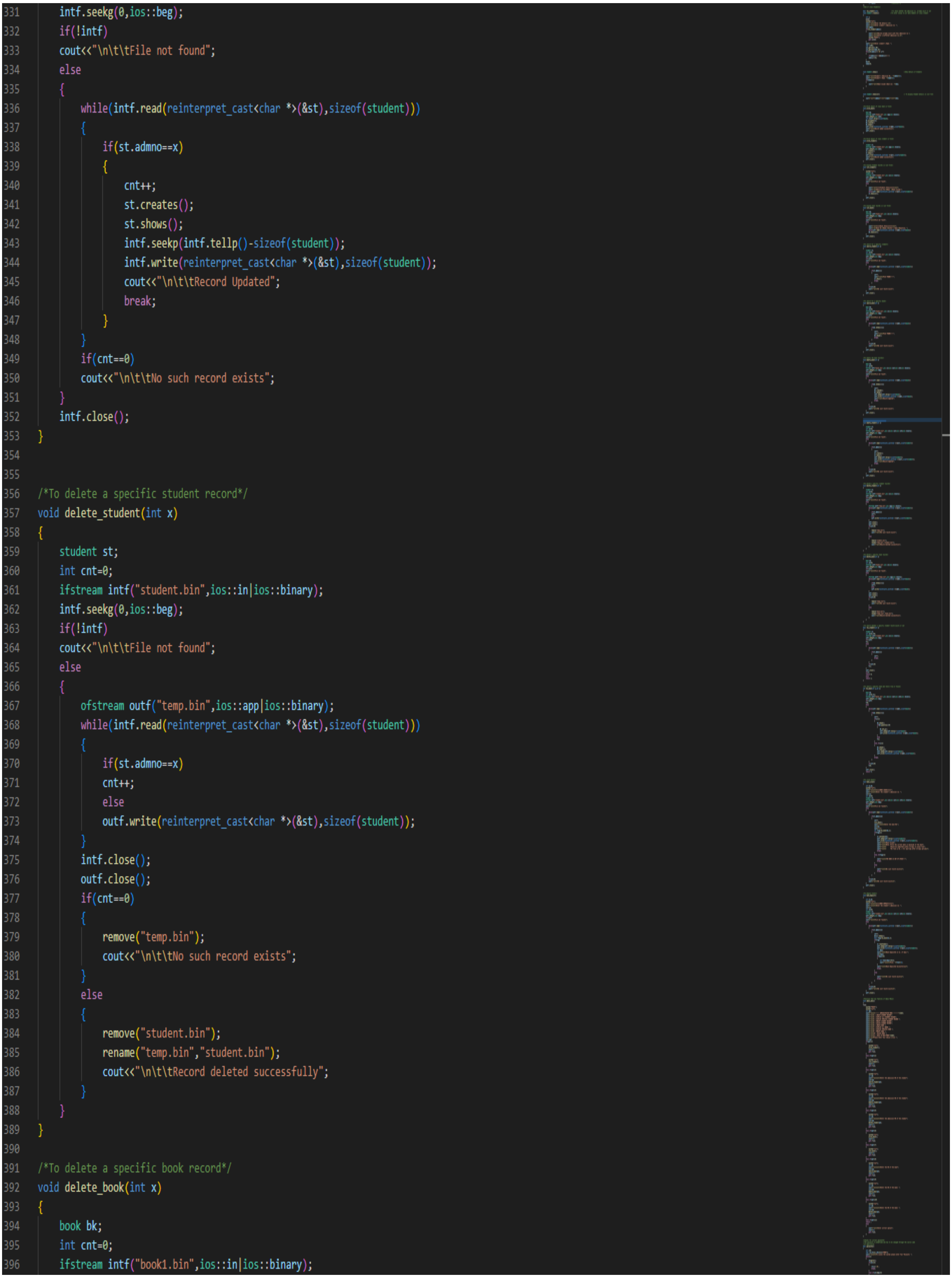
```
        cout<<"\n\t\tRecord already exist with this admission no.";
        cout<<"\n\t\tEnter a different admission no.\n";
        system("PAUSE");
        goto plane;
    }
    cout<<"\n\t\tEnter student's Name: ";
    char n[50];
    cin.getline(n,50);
    cin.getline(name,25);
    for(i=0;name[i]!='\0';i++)
    {
        if(name[i]>='a'&&name[i]<='z')
        name[i]-=32;
    }
    bno=0;
    token=0;
}


void student::shows()                               //Show details of Students
{
    cout<<"\n\t\tStudent's Admission No.: "<<admno<<el;
    cout<<"\n\t\tStudent's Name: "<<name<<el;
    if(token==1)
    {
        cout<<"\n\t\tBook Issued (Book no): "<<bno;
    }
}


void student::showlist()                            // To display Student details in list form
{
    cout<<"\n\t"<<admno<<"\t\t"<<name<<"\t\t"<<bno;
}


/*To Write object of class book in file*/
void write_book()
{
    book bk;
    ofstream outf("book1.bin",ios::app|ios::binary);
    outf.seekp(0,ios::end);
    int x=outf.tellp()/sizeof(book);
    bk.assignbno(x);
    bk.createb();
    bk.showb();
    outf.write(reinterpret_cast<char *>(&bk),sizeof(book));
    cout<<"\n\t\tRecord added successfully";
    outf.close();
}


/*To Write object of class student in file*/
void write_student()
{
    student st;
    ofstream outf("student.bin",ios::app|ios::binary);
    outf.seekp(0,ios::end);
    st.creates();
    st.shows();
    outf.write(reinterpret_cast<char *>(&st),sizeof(student));
    cout<<"\n\t\tRecord added successfully";
    outf.close();
}

/*To display Student records in list form*/
```



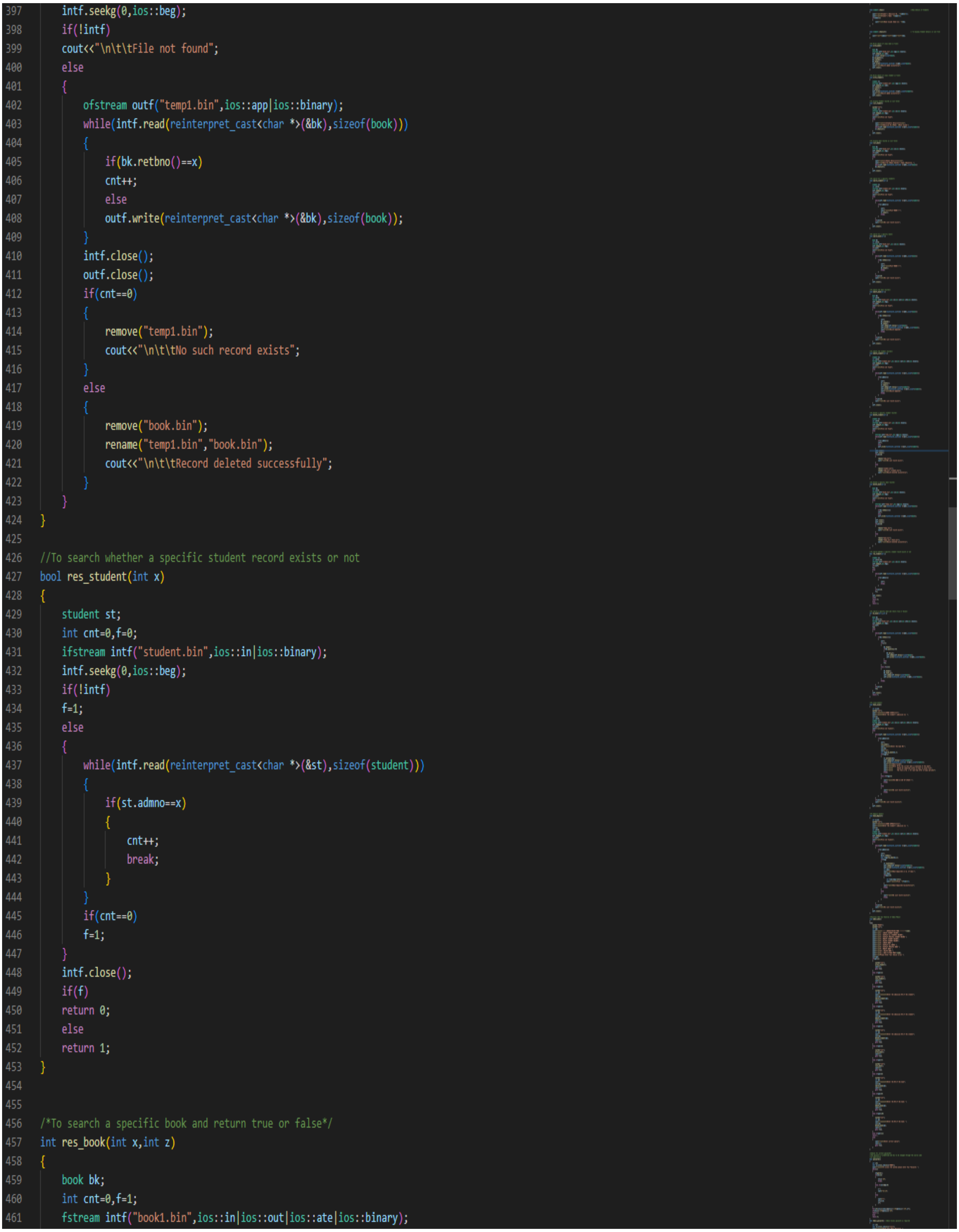
```cpp
void list_student()
{
    system("CLS");
    student st;
    ifstream intf("student.bin",ios::in|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        cout<<"\n\t*****Students Details*****\n\n";
        cout<<"\n\tAdmission No:\tName: \tBook Issued:";
        while(intf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        st.showlist();
    }
    intf.close();
}


/*To display book records in list form*/
void list_book()
{
    book bk;
    ifstream intf("book1.bin",ios::in|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        cout<<"\n\t*****Books Details*****\n\n";
        cout<<"\n\tBook No:\tName:\tAuthor's Name:\tQuantity: ";
        while(intf.read(reinterpret_cast<char *>(&bk),sizeof(book)))
        bk.showlist();
    }
    intf.close();
}


/*To search for a specific student*/
void search_student(int x)
{
    student st;
    int cnt=0;
    ifstream intf("student.bin",ios::in|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        while(intf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        {
            if(st.admno==x)
            {
                cnt++;
                cout<<"\n\t\tFILE FOUND!!!!";
                st.shows();
                break;
            }
        }
        if(cnt==0)
        cout<<"\n\t\tNo such record exists";
    }
    intf.close();
}
```



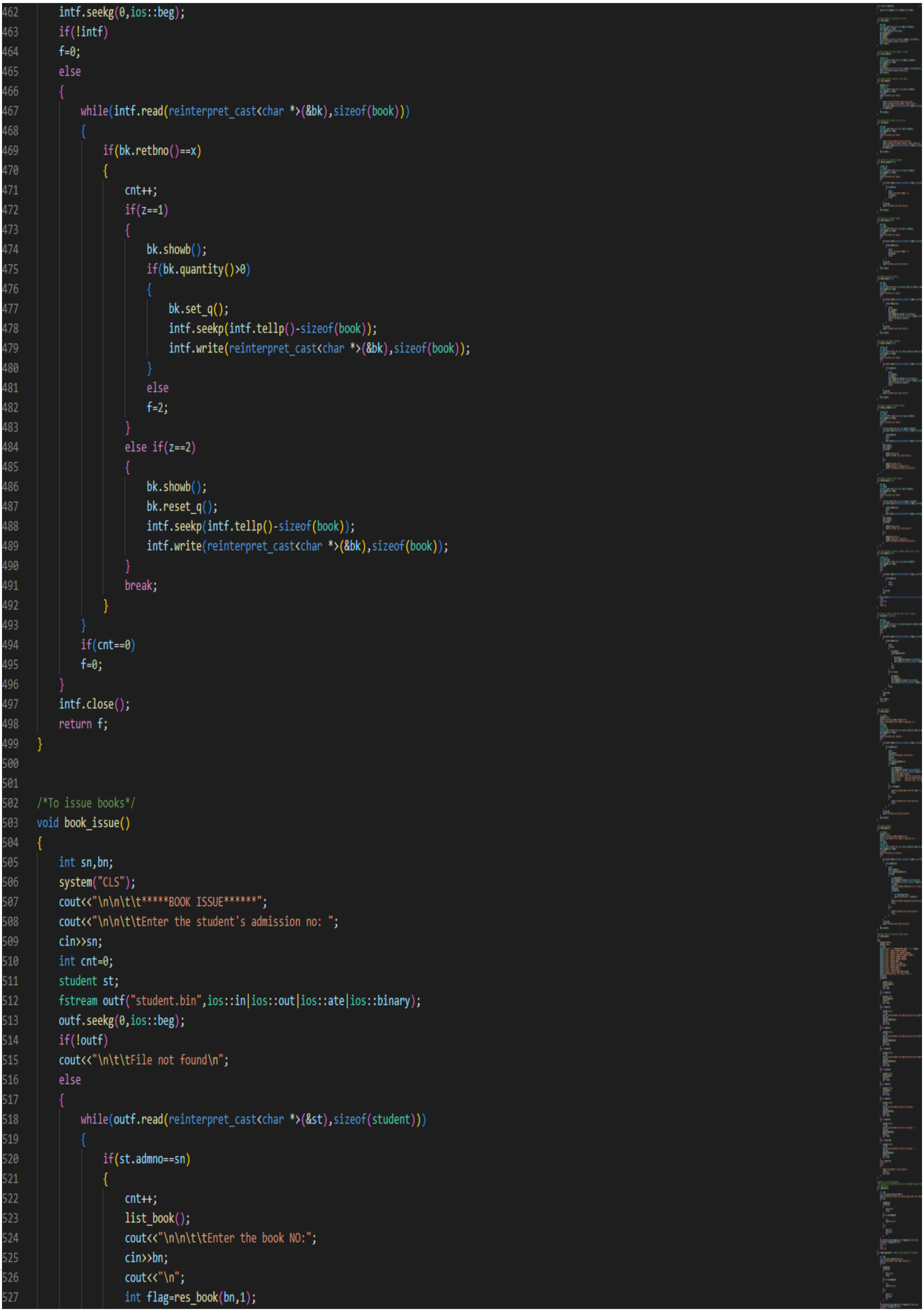
```
/*To search for a specific book*/
void search_book(int x)
{
    book bk;
    int cnt=0;
    ifstream intf("book1.bin",ios::in|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        while(intf.read(reinterpret_cast<char *>(&bk),sizeof(book)))
        {
            if(bk.retbno()==x)
            {
                cnt++;
                cout<<"\n\t\tFILE FOUND!!!!";
                bk.showb();
                break;
            }
        }
        if(cnt==0)
        cout<<"\n\t\tNo such record exists";
    }
    intf.close();
}


/*To modify the book records*/
void modify_book(int x)
{
    book bk;
    int cnt=0;
    fstream intf("book1.bin",ios::in|ios::out|ios::ate|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        while(intf.read(reinterpret_cast<char *>(&bk),sizeof(book)))
        {
            if(bk.retbno()==x)
            {
                cnt++;
                bk.createb();
                bk.showb();
                intf.seekp(intf.tellp()-sizeof(book));
                intf.write(reinterpret_cast<char *>(&bk),sizeof(book));
                cout<<"\n\t\tRecord Updated";
                break;
            }
        }
        if(cnt==0)
        cout<<"\n\t\tNo such record exists";
    }
    intf.close();
}


/*To modify the student records*/
void modify_student(int x)
{
    student st;
    int cnt=0;
    fstream intf("student.bin",ios::in|ios::out|ios::ate|ios::binary);
```



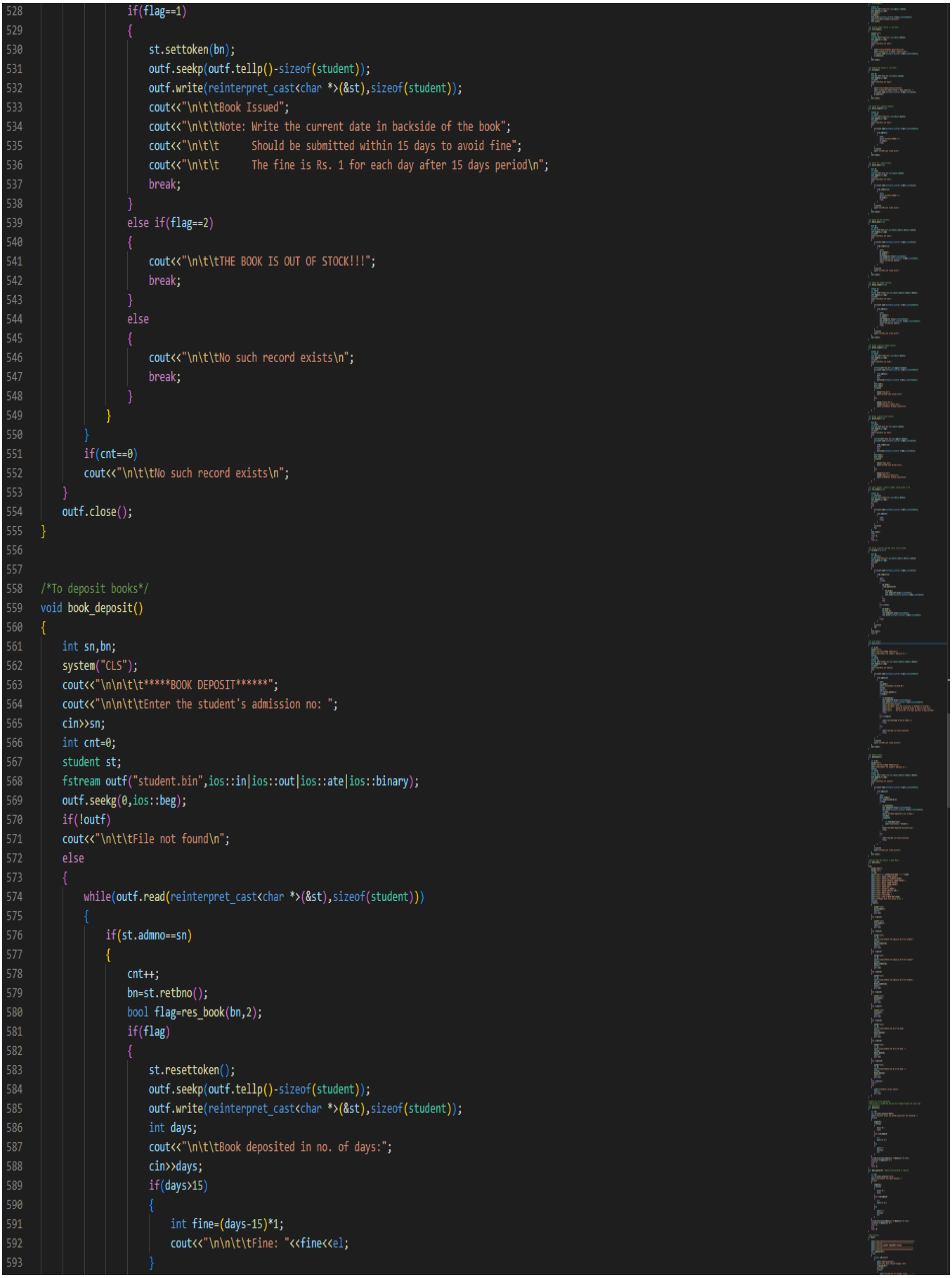
```
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        while(intf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        {
            if(st.admno==x)
            {
                cnt++;
                st.creates();
                st.shows();
                intf.seekp(intf.tellp()-sizeof(student));
                intf.write(reinterpret_cast<char *>(&st),sizeof(student));
                cout<<"\n\t\tRecord Updated";
                break;
            }
        }
        if(cnt==0)
        cout<<"\n\t\tNo such record exists";
    }
    intf.close();
}


/*To delete a specific student record*/
void delete_student(int x)
{
    student st;
    int cnt=0;
    ifstream intf("student.bin",ios::in|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        ofstream outf("temp.bin",ios::app|ios::binary);
        while(intf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        {
            if(st.admno==x)
            cnt++;
            else
            outf.write(reinterpret_cast<char *>(&st),sizeof(student));
        }
        intf.close();
        outf.close();
        if(cnt==0)
        {
            remove("temp.bin");
            cout<<"\n\t\tNo such record exists";
        }
        else
        {
            remove("student.bin");
            rename("temp.bin","student.bin");
            cout<<"\n\t\tRecord deleted successfully";
        }
    }
}

/*To delete a specific book record*/
void delete_book(int x)
{
    book bk;
    int cnt=0;
    ifstream intf("book1.bin",ios::in|ios::binary);
```



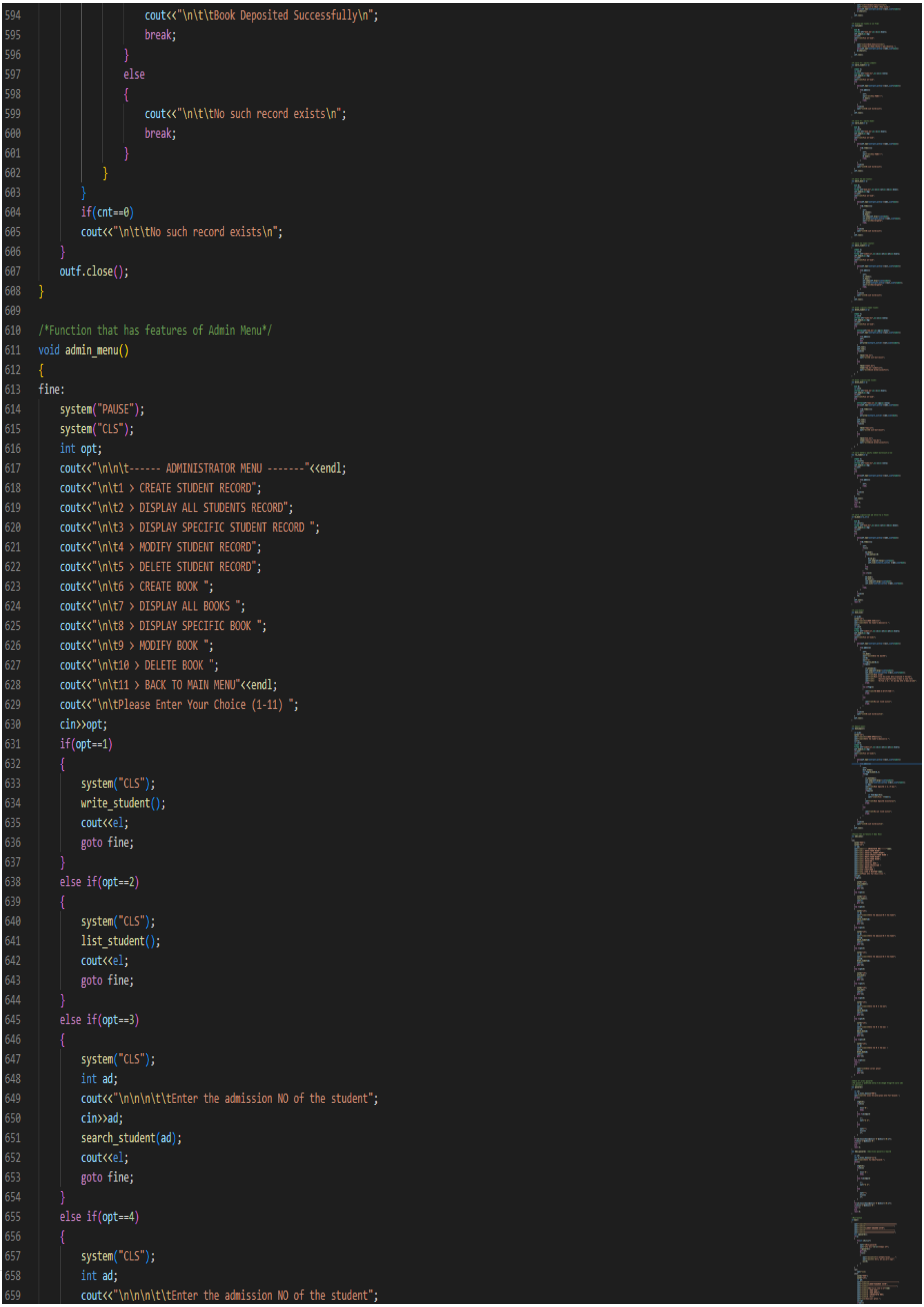
```
    intf.seekg(0,ios::beg);
    if(!intf)
    cout<<"\n\t\tFile not found";
    else
    {
        ofstream outf("temp1.bin",ios::app|ios::binary);
        while(intf.read(reinterpret_cast<char *>(&bk),sizeof(book)))
        {
            if(bk.retbno()==x)
            cnt++;
            else
            outf.write(reinterpret_cast<char *>(&bk),sizeof(book));
        }
        intf.close();
        outf.close();
        if(cnt==0)
        {
            remove("temp1.bin");
            cout<<"\n\t\tNo such record exists";
        }
        else
        {
            remove("book.bin");
            rename("temp1.bin","book.bin");
            cout<<"\n\t\tRecord deleted successfully";
        }
    }
}

//To search whether a specific student record exists or not
bool res_student(int x)
{
    student st;
    int cnt=0,f=0;
    ifstream intf("student.bin",ios::in|ios::binary);
    intf.seekg(0,ios::beg);
    if(!intf)
    f=1;
    else
    {
        while(intf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        {
            if(st.admno==x)
            {
                cnt++;
                break;
            }
        }
        if(cnt==0)
        f=1;
    }
    intf.close();
    if(f)
    return 0;
    else
    return 1;
}


/*To search a specific book and return true or false*/
int res_book(int x,int z)
{
    book bk;
    int cnt=0,f=1;
    fstream intf("book1.bin",ios::in|ios::out|ios::ate|ios::binary);
```



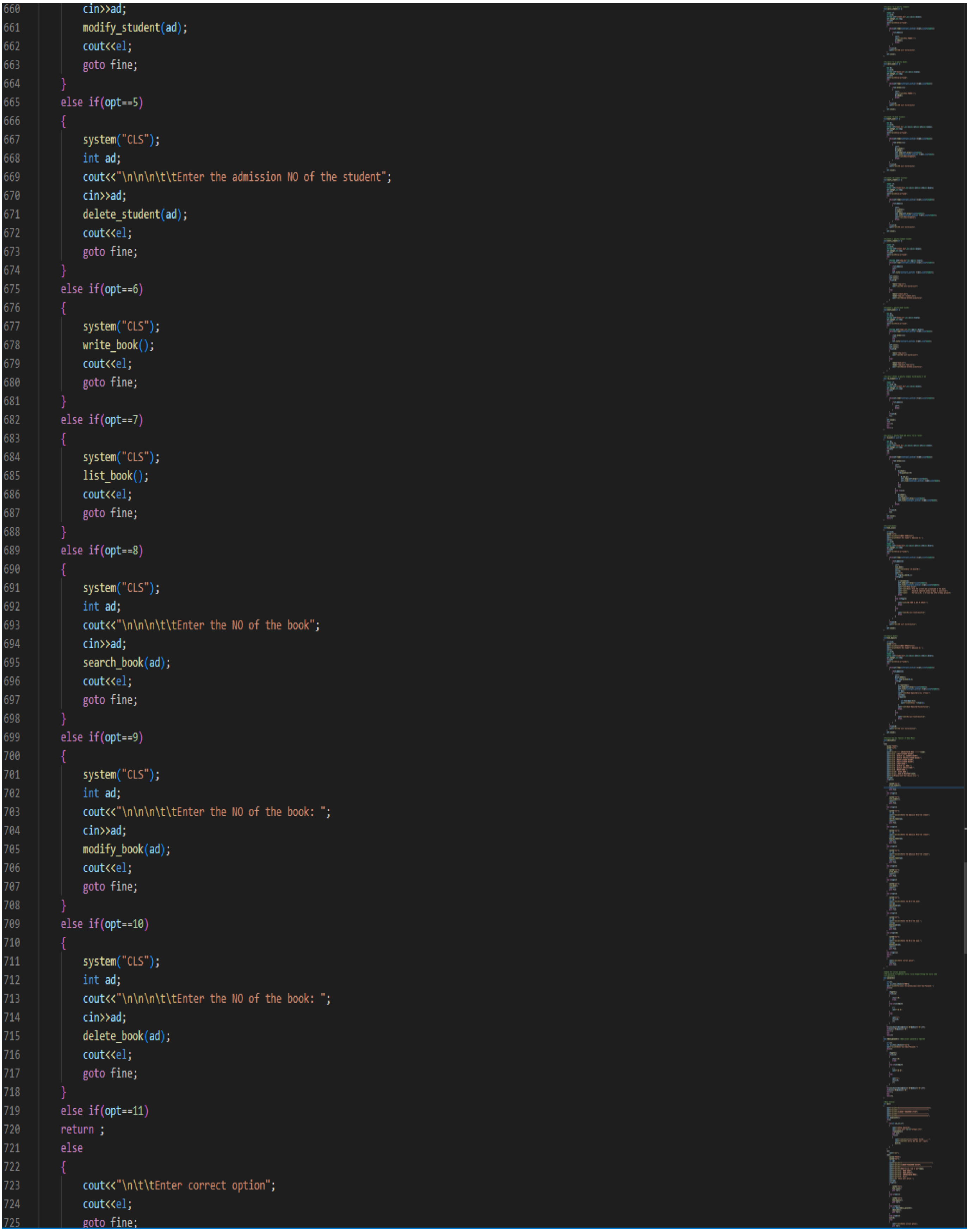
```
    intf.seekg(0,ios::beg);
    if(!intf)
    f=0;
    else
    {
        while(intf.read(reinterpret_cast<char *>(&bk),sizeof(book)))
        {
            if(bk.retbno()==x)
            {
                cnt++;
                if(z==1)
                {
                    bk.showb();
                    if(bk.quantity()>0)
                    {
                        bk.set_q();
                        intf.seekp(intf.tellp()-sizeof(book));
                        intf.write(reinterpret_cast<char *>(&bk),sizeof(book));
                    }
                    else
                    f=2;
                }
                else if(z==2)
                {
                    bk.showb();
                    bk.reset_q();
                    intf.seekp(intf.tellp()-sizeof(book));
                    intf.write(reinterpret_cast<char *>(&bk),sizeof(book));
                }
                break;
            }
        }
        if(cnt==0)
        f=0;
    }
    intf.close();
    return f;
}


/*To issue books*/
void book_issue()
{
    int sn,bn;
    system("CLS");
    cout<<"\n\n\t\t*****BOOK ISSUE******";
    cout<<"\n\n\t\tEnter the student's admission no: ";
    cin>>sn;
    int cnt=0;
    student st;
    fstream outf("student.bin",ios::in|ios::out|ios::ate|ios::binary);
    outf.seekg(0,ios::beg);
    if(!outf)
    cout<<"\n\t\tFile not found\n";
    else
    {
        while(outf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        {
            if(st.admno==sn)
            {
                cnt++;
                list_book();
                cout<<"\n\n\t\tEnter the book NO:";
                cin>>bn;
                cout<<"\n";
                int flag=res_book(bn,1);
```



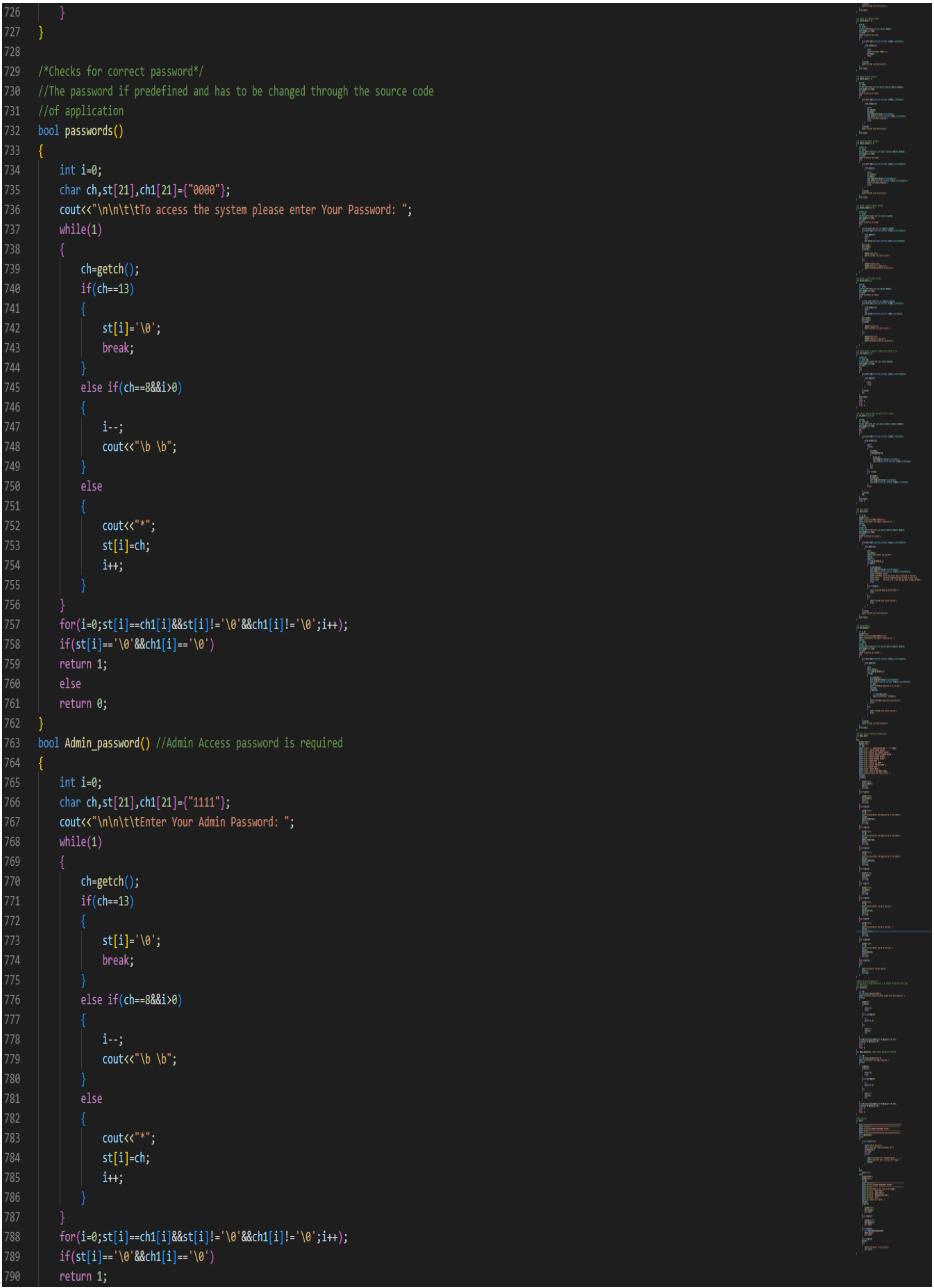
```
                if(flag==1)
                {
                    st.settoken(bn);
                    outf.seekp(outf.tellp()-sizeof(student));
                    outf.write(reinterpret_cast<char *>(&st),sizeof(student));
                    cout<<"\n\t\tBook Issued";
                    cout<<"\n\t\tNote: Write the current date in backside of the book";
                    cout<<"\n\t\t      Should be submitted within 15 days to avoid fine";
                    cout<<"\n\t\t      The fine is Rs. 1 for each day after 15 days period\n";
                    break;
                }
                else if(flag==2)
                {
                    cout<<"\n\t\tTHE BOOK IS OUT OF STOCK!!!";
                    break;
                }
                else
                {
                    cout<<"\n\t\tNo such record exists\n";
                    break;
                }
            }
        }
        if(cnt==0)
        cout<<"\n\t\tNo such record exists\n";
    }
    outf.close();
}


/*To deposit books*/
void book_deposit()
{
    int sn,bn;
    system("CLS");
    cout<<"\n\n\t\t*****BOOK DEPOSIT******";
    cout<<"\n\n\t\tEnter the student's admission no: ";
    cin>>sn;
    int cnt=0;
    student st;
    fstream outf("student.bin",ios::in|ios::out|ios::ate|ios::binary);
    outf.seekg(0,ios::beg);
    if(!outf)
    cout<<"\n\t\tFile not found\n";
    else
    {
        while(outf.read(reinterpret_cast<char *>(&st),sizeof(student)))
        {
            if(st.admno==sn)
            {
                cnt++;
                bn=st.retbno();
                bool flag=res_book(bn,2);
                if(flag)
                {
                    st.resettoken();
                    outf.seekp(outf.tellp()-sizeof(student));
                    outf.write(reinterpret_cast<char *>(&st),sizeof(student));
                    int days;
                    cout<<"\n\t\tBook deposited in no. of days:";
                    cin>>days;
                    if(days>15)
                    {
                        int fine=(days-15)*1;
                        cout<<"\n\n\t\tFine: "<<fine<<el;
                    }
```



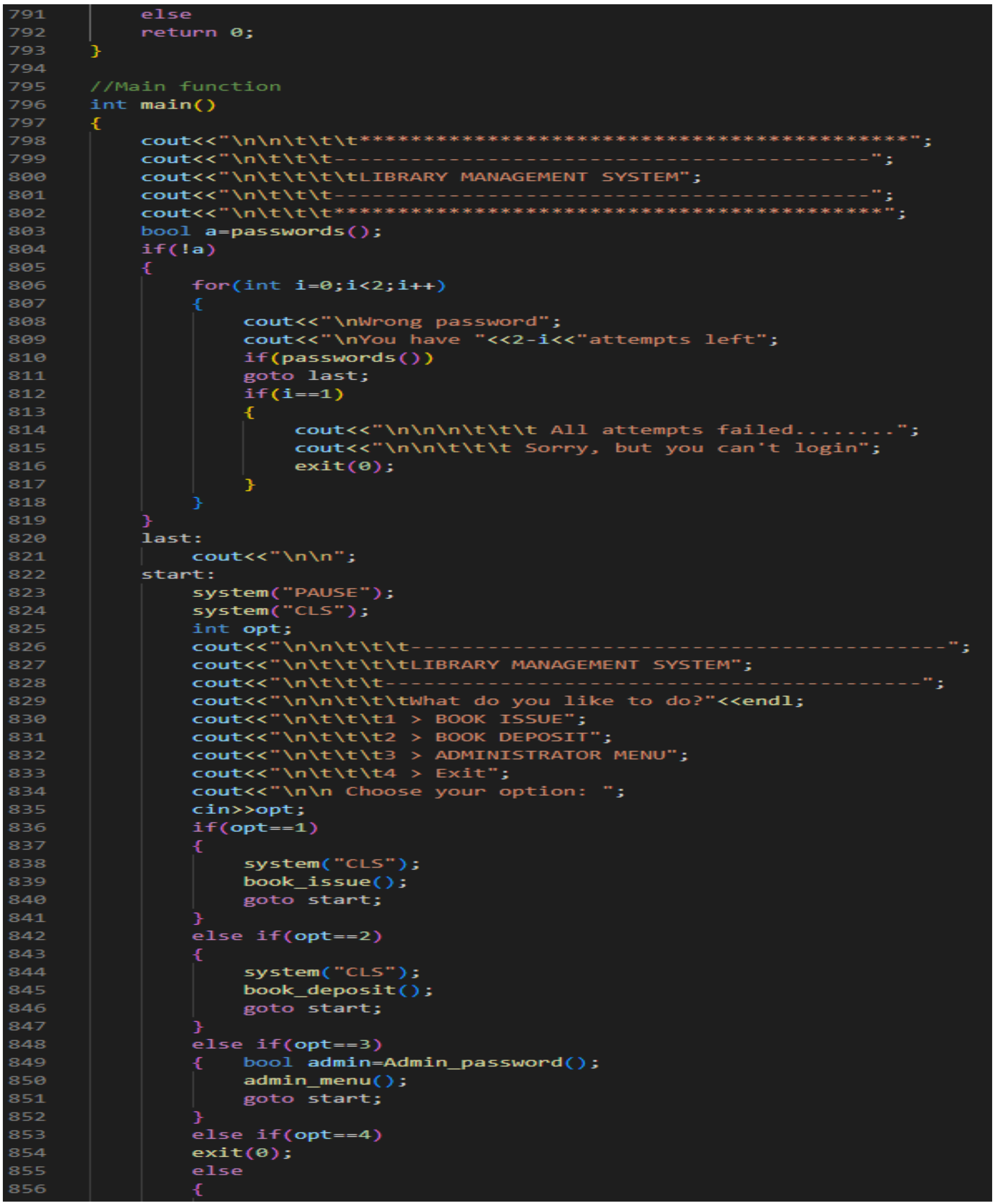
```
                    cout<<"\n\t\tBook Deposited Successfully\n";
                    break;
                }
                else
                {
                    cout<<"\n\t\tNo such record exists\n";
                    break;
                }
            }
        }
        if(cnt==0)
        cout<<"\n\t\tNo such record exists\n";
    }
    outf.close();
}

/*Function that has features of Admin Menu*/
void admin_menu()
{
fine:
    system("PAUSE");
    system("CLS");
    int opt;
    cout<<"\n\n\t------ ADMINISTRATOR MENU -------"<<endl;
    cout<<"\n\t1 > CREATE STUDENT RECORD";
    cout<<"\n\t2 > DISPLAY ALL STUDENTS RECORD";
    cout<<"\n\t3 > DISPLAY SPECIFIC STUDENT RECORD ";
    cout<<"\n\t4 > MODIFY STUDENT RECORD";
    cout<<"\n\t5 > DELETE STUDENT RECORD";
    cout<<"\n\t6 > CREATE BOOK ";
    cout<<"\n\t7 > DISPLAY ALL BOOKS ";
    cout<<"\n\t8 > DISPLAY SPECIFIC BOOK ";
    cout<<"\n\t9 > MODIFY BOOK ";
    cout<<"\n\t10 > DELETE BOOK ";
    cout<<"\n\t11 > BACK TO MAIN MENU"<<endl;
    cout<<"\n\tPlease Enter Your Choice (1-11) ";
    cin>>opt;
    if(opt==1)
    {
        system("CLS");
        write_student();
        cout<<el;
        goto fine;
    }
    else if(opt==2)
    {
        system("CLS");
        list_student();
        cout<<el;
        goto fine;
    }
    else if(opt==3)
    {
        system("CLS");
        int ad;
        cout<<"\n\n\n\t\tEnter the admission NO of the student";
        cin>>ad;
        search_student(ad);
        cout<<el;
        goto fine;
    }
    else if(opt==4)
    {
        system("CLS");
        int ad;
        cout<<"\n\n\n\t\tEnter the admission NO of the student";
```



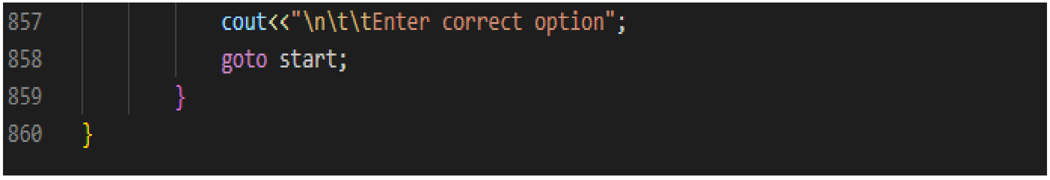
```
        cin>>ad;
        modify_student(ad);
        cout<<el;
        goto fine;
    }
    else if(opt==5)
    {
        system("CLS");
        int ad;
        cout<<"\n\n\n\t\tEnter the admission NO of the student";
        cin>>ad;
        delete_student(ad);
        cout<<el;
        goto fine;
    }
    else if(opt==6)
    {
        system("CLS");
        write_book();
        cout<<el;
        goto fine;
    }
    else if(opt==7)
    {
        system("CLS");
        list_book();
        cout<<el;
        goto fine;
    }
    else if(opt==8)
    {
        system("CLS");
        int ad;
        cout<<"\n\n\n\t\tEnter the NO of the book";
        cin>>ad;
        search_book(ad);
        cout<<el;
        goto fine;
    }
    else if(opt==9)
    {
        system("CLS");
        int ad;
        cout<<"\n\n\n\t\tEnter the NO of the book: ";
        cin>>ad;
        modify_book(ad);
        cout<<el;
        goto fine;
    }
    else if(opt==10)
    {
        system("CLS");
        int ad;
        cout<<"\n\n\n\t\tEnter the NO of the book: ";
        cin>>ad;
        delete_book(ad);
        cout<<el;
        goto fine;
    }
    else if(opt==11)
    return ;
    else
    {
        cout<<"\n\t\tEnter correct option";
        cout<<el;
        goto fine;
```



```
    }
}

/*Checks for correct password*/
//The password if predefined and has to be changed through the source code
//of application
bool passwords()
{
    int i=0;
    char ch,st[21],ch1[21]={"0000"};
    cout<<"\n\n\t\tTo access the system please enter Your Password: ";
    while(1)
    {
        ch=getch();
        if(ch==13)
        {
            st[i]='\0';
            break;
        }
        else if(ch==8&&i>0)
        {
            i--;
            cout<<"\b \b";
        }
        else
        {
            cout<<"*";
            st[i]=ch;
            i++;
        }
    }
    for(i=0;st[i]==ch1[i]&&st[i]!='\0'&&ch1[i]!='\0';i++);
    if(st[i]=='\0'&&ch1[i]=='\0')
    return 1;
    else
    return 0;
}
bool Admin_password() //Admin Access password is required
{
    int i=0;
    char ch,st[21],ch1[21]={"1111"};
    cout<<"\n\n\t\tEnter Your Admin Password: ";
    while(1)
    {
        ch=getch();
        if(ch==13)
        {
            st[i]='\0';
            break;
        }
        else if(ch==8&&i>0)
        {
            i--;
            cout<<"\b \b";
        }
        else
        {
            cout<<"*";
            st[i]=ch;
            i++;
        }
    }
    for(i=0;st[i]==ch1[i]&&st[i]!='\0'&&ch1[i]!='\0';i++);
    if(st[i]=='\0'&&ch1[i]=='\0')
    return 1;
```



```
    else
    return 0;
}

//Main function
int main()
{
    cout<<"\n\n\t\t\t*******************************************";
    cout<<"\n\t\t\t------------------------------------------";
    cout<<"\n\t\t\t\t\tLIBRARY MANAGEMENT SYSTEM";
    cout<<"\n\t\t\t------------------------------------------";
    cout<<"\n\t\t\t*******************************************";
    bool a=passwords();
    if(!a)
    {
        for(int i=0;i<2;i++)
        {
            cout<<"\nWrong password";
            cout<<"\nYou have "<<2-i<<"attempts left";
            if(passwords())
            goto last;
            if(i==1)
            {
                cout<<"\n\n\n\t\t\t All attempts failed........";
                cout<<"\n\n\t\t\t Sorry, but you can't login";
                exit(0);
            }
        }
    }
    last:
        cout<<"\n\n";
    start:
        system("PAUSE");
        system("CLS");
        int opt;
        cout<<"\n\n\t\t\t------------------------------------------";
        cout<<"\n\t\t\t\tLIBRARY MANAGEMENT SYSTEM";
        cout<<"\n\t\t\t------------------------------------------";
        cout<<"\n\n\t\t\tWhat do you like to do?"<<endl;
        cout<<"\n\t\t\t1 > BOOK ISSUE";
        cout<<"\n\t\t\t2 > BOOK DEPOSIT";
        cout<<"\n\t\t\t3 > ADMINISTRATOR MENU";
        cout<<"\n\t\t\t4 > Exit";
        cout<<"\n\n Choose your option: ";
        cin>>opt;
        if(opt==1)
        {
            system("CLS");
            book_issue();
            goto start;
        }
        else if(opt==2)
        {
            system("CLS");
            book_deposit();
            goto start;
        }
        else if(opt==3)
        {   bool admin=Admin_password();
            admin_menu();
            goto start;
        }
        else if(opt==4)
        exit(0);
        else
        {
```



```
            cout<<"\n\t\tEnter correct option";
            goto start;
        }
}
```

Section 9.4: Conclusion

It is important provide users a great experience and make sure modern applications are introduced in the market. We are living in a technology society where people are even more connected to the world outside, with a small device they can access any kind of information required. Gathering participants opinions and suggestions can help companies to find out what users need and what they are looking for, however you must remember participants are not UX professionals, but they are experts in what they do: using the website or app. Not that every suggestion they give must be taken as the last word, but they will certainly raise questions and points for improvement that will benefit UX teams.

The most impactful observations should be included in the usability test report/presentation, as part of the results, which will justify design adjustments, investments in technology, validation of assumptions and potentially resolve internal conflicts between stakeholders about the direction the system should go.

Don't forget that the participant is a person with their own ways of using your system. Just because you are patient and considerate, don't expect everyone to be like you. Each user has a way to test. Remember, you're not testing the user, you're testing your system.

Section 10: References

Stakeholder Analysis for UX Projects. (n.d.). Nielsen Norman Group. Retrieved October 12, 2022, from https://www.nngroup.com/articles/stakeholder-analysis/

UX Stakeholders: Study Guide. (n.d.). Nielsen Norman Group. Retrieved October 12, 2022, from https://www.nngroup.com/articles/stakeholders-study-guide/

Open-Ended vs. Closed-Ended Questions in User Research. (n.d.). Nielsen Norman Group. Retrieved October 12, 2022, from https://www.nngroup.com/articles/open-endedquestions/

Stevens, E. (2022, August 24). *What is UI design? A complete introductory guide*. UX Design Institute. Retrieved October 12, 2022, from https://www.uxdesigninstitute.com/blog/what-is-ui-design/

Introduction to UX Research: The Ultimate Guide For Getting Started. (2021, December 13). Optimal Workshop. Retrieved October 12, 2022, from https://blog.optimalworkshop.com/intro-to-ux-research/?utm_source=google

ClearPoint Design & Product Strategy. (n.d.). ClearPoint. Retrieved October 12, 2022, from https://clearpoint.digital/what-do-you-need/design-productstrategy/?campaignid=17872203542