Online Library Access Android Application

Thesis submitted to the Institute of Information Technology, Quaid-i-Azam University, Islamabad, for the partial fulfillment of the degree of Master of Science in Information Technology.

By

Muhammad Ali Shahzad

Supervised by :

Madam Robina Rashid

Institute of Information Technology

Quaid-e-Azam University Islamabad

Session 2019-2021

Quaid-e-Azam University Islamabad

Statement of submission

This is to certify that **Muhammad Ali Shahzad** Reg No. **01161911032** has successfully completed the final year project named as: **Online Library** at the faculty of Information Technology, Quaid-I-Azam university Islamabad, to fulfil the partial requirement of the degree of **Master's in information technology**.

Project Management
Office Faculty of IIT
QAU

Project Advisor Designation

Department

Head of Department IT

Faculty of IIT , QAU

Acknowledgement

All praise to Almighty Allah for giving me the courage to complete this project within the specified time. I want to express my gratitude to my kind supervisor **Madam Robina Rashid** who kept my morale high by her appreciation and motivation and guided and taught me different techniques. She was always available whenever I consult her. Her valuable suggestion significantly improved this work and without her precious guidance I would never be able to complete such project. I am extremely fortunate to have her as supervisor.

I would like to acknowledge the support of my family members, I owe all my achievements to my truly, sincere, and most loving parents, Brother, and Sister. They mean a lot to me.

I am also thankful to all the worthy teachers and staff members of Institute of Information Technology, Quaid-i-Azam University Islamabad. At the end I would like to extend thanks to all my friends, class fellows who have helped me in this project.

Muhammad Ali Shahzad

DECLARATION

I Muhammad Ali Shahzad Reg No. 01161911032 Student of Master of Information Technology; Department of Information Technology, Quaid-I-Azam university Islamabad, solemnly declare that the data quoted in this project title "Online Library" is based on my original work and has not yet been submitted or published elsewhere.

Muhammad Ali Shahzad ——————————

I certify that the mentioned above Student of Master of Information Technology; Department of Information Technology, Quaid-I-Azam university Islamabad, worked under my supervision and the above stated

declaration is true to the best of my knowledge.

Madam Robina Rashid ——————————
Department of IIT , QAU

Table Of Contents

CHAPTER 1
INTRODUCTION

1.1 Introduction :-

Books are the back bone in a life of a human beings specially students and teachers. We cant study all time on electronic devices as it cause many hazard effect on human health. It create stress on brain and eyes on the other side if we compare books with electronic devices with books so books are more user friendly then as they say " books are the good friends".

The reason why I selected this project mobile app for the library system as most of the time student faces difficulty in searching books and looking for its availability. The objective of this project is to develop an mobile app for the library system form where all the students in the institution can access the library on their palm with comfort through internet.

1.2 Literature Review :-

My project is based on educational concerns and this app support the interaction between students and books available in library as well as the management of the library to maintain the record of the books in the library and of those students who borrowed books and for those who are requesting for books.

   As now a days every thing is available on android mobiles as well as education so it is the demand of time the time that we have to produce the ways that which makes the interaction between books and student easy and modern .

1.3 Project Description :-

This app facilitates student to access the library of their institution through android mobile .As a student myself I know the difficulties faced by a student if they need to take a book from the library. Online LMA is an app from where student can access the books in the library and come to know about the status of the book on their palm. If a student want to take a book from the library he has to login through his student id which was given by the administration .then he has to select the subject to whom book is related like c++ programming with computer sciences etc.. then if the book is available in the library then the front cover of the

book is appeared on the screen. And its status is also shown on its top wheather it is free or borrowed. Other advantage of this book is for the library administration to maintain the record of books present in library and the records of students who has taken the book from the library and requested for the book. The librarians is authorized by the administrator while sign up or login and user are authorized by librarians.

1.4 key Features :-

Following are the key features of this system:

- Administrator can login
- Administrator can approve librarian or user accounts
- Librarian can sign up/login
- Librarian can search add, update status or delete the books
- Librarian issue books to the user
- Student can login
- Student can search or request for the book
- Student can also see the status of requested books

1.5 Tools and technologies :-

- Xml [Front End] in android studio
- Android studio, java JDK
- Java and FIRE BASE (Back End)

CHAPTER 2
REQUIREMENT ANALYSIS

2.1 Introduction

In a software engineering application developers need to clearly understand the problems to be solved.It is therefore important for a developer to properly model the scenarios that can influence the solution to the problem by collecting relevant information. Requirements define the function of the system from the client's viewpoint. The requirements establish the system's functionality, constraints, and goals by consultation with the client, customers, and users. The requirements may be developed in a self-contained study, or may emerge incrementally. The requirements form the basis for acceptance testing. The development team and the client need to work together closely during the requirements phase of a so(ware project. The requirements must be developed in a manner that is understandable by both the client and the development staff.

2.2 Requirement Elicitation

In requirement engineering, requirements elicitation is the practice of researching and discovering the requirements of a system from users, customers, and other stakeholders.[1] The practice is also sometimes referred to as "requirement gathering".

The term elicitation is used in books and research to raise the fact that good requirements cannot just be collected from the customer, as would be indicated by the name requirements gathering. Requirements elicitation is non-trivial because you can never be sure you get all requirements from the user and customer by just asking them what the system should do or not do (for Safety and Reliability). Requirements elicitation practices include interviews, questionnaires, user observation, workshops, brainstorming, use cases, role playing and prototyping.

2.3 Requirement Analysis

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the needs of different stakeholders. Therefore requirements analysis means to analyze, document, validate and manage software or system requirements.

2.4 Requirement Specification

A Requirement Specification is a collection of the set of all requirements that are to be imposed on the design and verification of the product. The specification also contains other related information necessary for the design, verification, and maintenance of the product.

2.5 Functional Requirements

This section describe different requirements that are accomplished by the online library access system for the students .In order to achieve the desired goal of this project, the functional requirements must be met. The following are the three major actions performed by the online library access system.

2.5.1 Input Requirements

Input requirements are those which a user must fulfil before gaining access to use the online library access system.A registered user can use the application by providing the correct sign_in credentials. After being authorized the user should be able to browse through the collection of books, view status, select desired book and request for it then take it physically through library.

2.5.2 Operational requirements

Operartions performed in the system are registering user and checking the availability of an internet connection and accessing data from firebase and display the content to the user via the UI.

2.5.3 Output Requirements

It ensures that the application is able to issue ther order to the admin and after selecting the desire book and confirming the status of the book librarian issue the book physically to the students and update the current status of book. some of the functional requirements from admin side of the applications are as follow.

2.5.4 Admin log_in

Admin is only one person so it is not needed to signup for admin in this application. So the data of admin can be manually added to the firebase and admin have to just login.

2.5.5 Authenticate accounts

Admin can authenticate the accounts of newly registerd user and can suspend or approve the account.

2.5.6 View books

Admin can browse collection of books.

2.5.7 Logout

Admin can logout form the system

2.6 Non-Functional Requirements

Non-functional requirements are requirements that do not affect the proper running of the Online Bookstore system. However, It is worthwhile to mention and consider these requirements for the purpose of software quality and analysis.

2.6.1 External Interface or Hardware Requirements

In the development of this application, certain hardware requirements and specifications were considered in order for the application to be functional and result-oriented. This application is developed for Android-based phones and tablets and hence runs on an Android OS platform. With the help of the Android application framework and APIs, this application can utilize the device hardware features such as camera, sensor and touch screen capability. The following are the hardware specifications:

- The device must support for minimum network capability (EDGE - Enhanced Data Rates for Global Evolution, HSPA — High Speed Packet Access, EV-DO — Evolution Data Optimized, 802.11g and Wi-Fi)

- The device must have at least 128MB of memory available to the Linux Kernel

- The device must have at least 1GB of non-volatile storage for user data

- The download Manager capable of downloading individual files of at least SSMB in size e The device must implement at least a soft keyboard for user input

- The device must have a touch screen (capacitive or resistive touch screen)

- The device must support dynamic orientation by application to either portrait or landscape screen orientation

2.6.2 Performance

Performance requirements define acceptable response times for system functionality. This means that the retrieval speeds of any information, data about projects and comma should be fast to perform the actions quickly.
- The performance of this system include:
- The load time for user interface screens shall not take longer than two seconds
- The log in information shall be verified within five seconds.
- Queries shall return results within five seconds.

2.6.3 Security and privacy

System must be developed in such way that it would provide security and privacy It should prevent any unauthorized access to system. Security and privacy of system will be:

- A manger's password used for login must have strong password eg) password should be combination of characters, symbols and letters.

- A manager's password used for login must be changed after every three months,

- The display shall not require any unauthorized user to log-in eg) biometrie authorization system should be there.

2.6.4 Usability

Usability is very important non-functional attribute which help the new user to understand the system. This means system should be designed In an Interactive waythat our system should be easy to use, easy to learn and easy to handle. Usability of this system includes;

- The interface is easy to understand as It contains images of every dish,

- The interface appears easy to use rather than demanding and frustrating,

2.6.5 Reliability

System should be reliable and it must perform all the tasks for what it Is made. it should not be accepted if system crashes during run time. System will have said to be more raliable.

2.7 Resource Requirements:

Resource requirements are the requirements that we need to achieve the objectives of our Project. To be used efficiently, all Computer Resource software needs certain hardware components or other software resources to be present on a computer.

2.7.1 Software Requirement

The software requirements are description of features and functionalities of the target system.

2.7.1.1 Mobile based Development

- Java, XML, Jason Is used as developing languages.

- Android operating system. Java JDK.

- Android version latest.

2.7.2 Hardware Requirements .

For android, based development system on which this application will be developed will have at least:

- 64-bit operating system
- Updated processor
- 8.00GB Ram
- 1TB hard disk

Chapter 3
Analysis, Design And Use Cases

3.1 Analysis

The Analysis Phase is where the project lifecycle begins. The Analysis Phase Is where you break down the deliverables in the high-level Project Charter into the more detailed business requirements. The Analysis Phase Is also the part of the project where you identify the overall direction that the project will take through the creation of the project strategy documents.

3.1.1 Objectives

Successful completion of the Analysis Phase should comprise:

- Definition of approved requirements

- Creation of the System Requirements Document and Requirements Traceability Matrix

- Development of planned test activities

- Approval to progress to the Design Phase

3.1.2 Goals

The purpose of the Analysis Phase is to transform the needs and high-level requirements specified in earlier phases into unambiguous (measurable and testable), traceable, complete, consistent, and stakeholder-approved requirements.

3.2 Software Design

Software design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation.

For assessing user requirements, an SRS (Software Requirement Specification) document is created whereas for coding and implementation, there is a need of more specific and detailed requirements in software terms. The output of this process can directly be used into implementation in programming languages.

Software design is the first step in SDLC (Software Design Life Cycle), which moves the concentration from problem domain to solution domain. It tries to specify how to fulfill the requirements mentioned in SRS.

Software Design Levels

software design yields three Levels of results:

3.2.1 Architectural Design

The architectural design is the highest abstract version of the system. It identifies the & software as a system with many components interacting with each other. At this level, the designers get the idea of proposed solution domain.

3.2.2 High-level Design

The high-level design breaks the 'single entity-multiple component' concept of architectural design into less-abstracted view of sub-systems and modules and depicts their Interaction with each other. High-level design focuses on how the system along with all of its components can be implemented in forms of modules. It recognizes modular structure of " each sub-system and their relation and interaction among each other.

3.2.3 Detailed Design

Detailed design deals with the implementation part of what is seen as a system and its subsystems in the previous two designs. It is more detailed towards modules and their implementations. It defines logical structure of each module and their interfaces to communicate with other modules.

3.3 UML diagram

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with Its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system. Following are UML diagrams for this project .

3.3.1 Use case

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It consists of a group of elements (for example, classes and interfaces) that can be used together in a way that will have an effect larger than the sum of the separate elements combined. The use case should contain all system activities that have significance to the users.

A use case has three characteristics:

- Organizes functional requirements
- Models the goals of System/actor (user) interactions
- Record Ss Paths (called Scenarios) from trigger events to goals
- Describes one main flow of events (also called a basic course of action), and possibly Other ones, called exception flows of events.
- Is multilevel, so that one use case can use the functionality of another one.
- Use cases can be employed during several stages of software development, such as planning system requirements, validating design, testing software,

and creating an outline for online help and user manuals.

3.3.1.2 Use Case For Admin
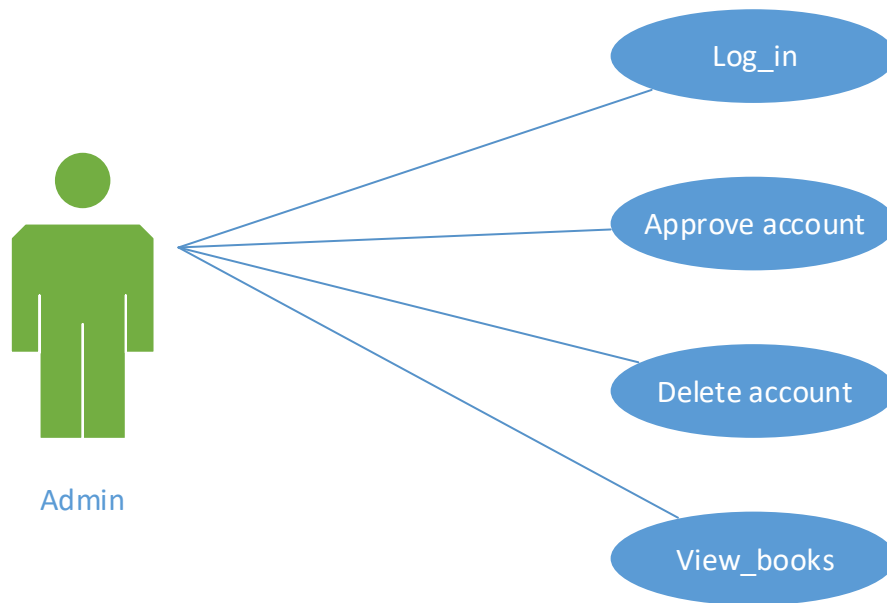
3.3.1.2 Use Case For Librarian
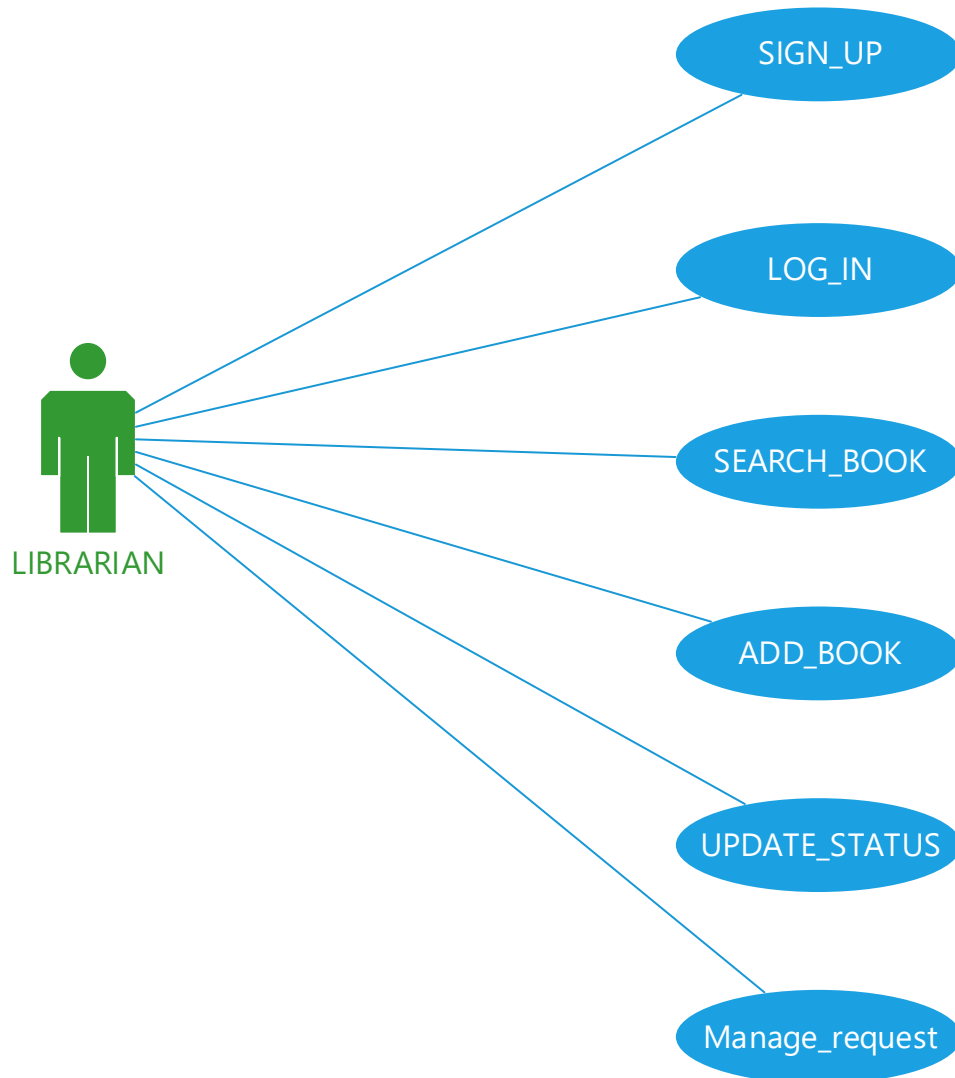
3.3.1.3 Use Case For Student
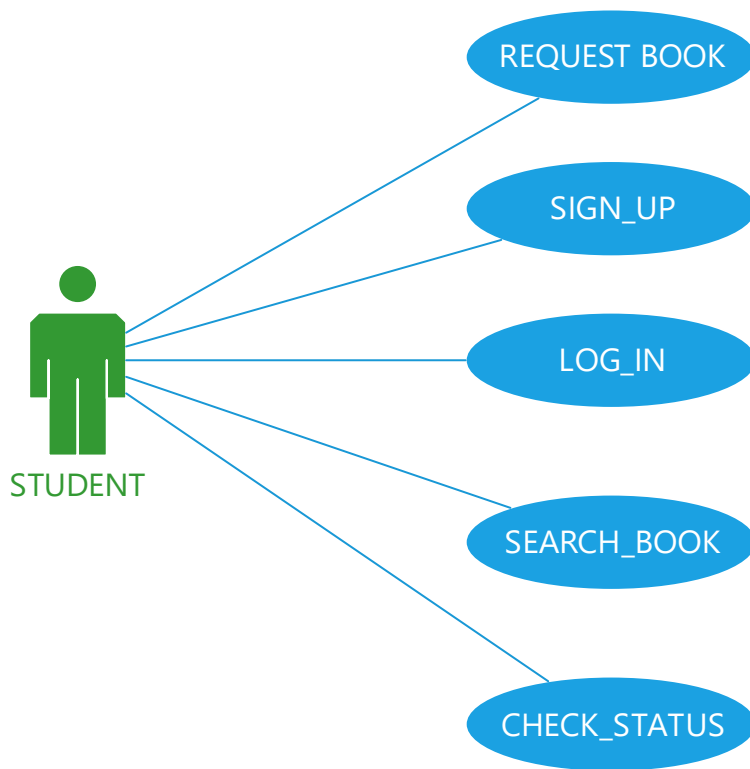
3.3.2 Use Case Description
List the steps involved in completing a use case, with the flow of events tool. Flow of Events is a critical element of any use case description.

Table 3.1 Login

USE CASE ID	01
USE CASE NAME	Login
ACTOR	Librarian/ Student
DESCRIPTION	Both actor can loin by signing up
PRE CONDITION	Both should have internet connection
POST CONDITION	Logged in
BASIC FLOW	Both actor should enter email, password and then login.

Table 3.2 Signup

USE CASE ID	02
USE CASE NAME	SIGN UP
ACTOR	LIBRARIAN/STUDENT
DESCRIPTION	BOTH CAN SIGN UP FIRST
PRE CONDITION	SHOULD HAVE INTERNET CONNECTION
POST CONDITION	SHOULD ENTER DATA CORRECTLY
BASIC FLOW	FILLS THE NECESSARY FIELDS
EXCEPTIONAL FLOW	INVALID INFORMATION /INTERNET CONNECTION ERROR

Table 3.3 Authentication

USE CASE ID	03
USE CASE NAME	Authentication
ACTOR	Admin
DESCRIPTION	Admin can authenticate the accounts of other entities
PRE CONDITION	Should have internet connections
POST CONDITION	Admin is able to suspend or approve accounts
BASIC FLOW	Admin login and see the request of he account he or she wants to delete or approve
EXCEPTIONAL FLOW	Invalid login details. Internet connection error.

Table 3.4 Add Books

USE CASE ID	04
USE CASE NAME	Add Books
ACTOR	Librarian
DESCRIPTION	Librarian can add new books
PRE CONDITION	Should have internet connection
POST CONDITION	Librarian can add books
BASIC FLOW	Lib login and add the book
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.5 Delete Books

USE CASE ID	05
USE CASE NAME	Delete Books
ACTOR	Librarian
DESCRIPTION	Librarian can delete books
PRE CONDITION	Should have internet connection
POST CONDITION	Librarian can delete books
BASIC FLOW	Lib login and delete the book
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.6 Edit Books

USE CASE ID	06
USE CASE NAME	Edit Books
ACTOR	Librarian
DESCRIPTION	Librarian can Edit books
PRE CONDITION	Should have internet connection
POST CONDITION	Librarian can edit books
BASIC FLOW	Lib login and edit the book
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.7 Update Status

USE CASE ID	07
USE CASE NAME	Update Books status
ACTOR	Librarian
DESCRIPTION	Librarian can update status of books
PRE CONDITION	Should have internet connection
POST CONDITION	Librarian can update status of books
BASIC FLOW	Lib login and edit status the book
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.8 Manage Book Request

USE CASE ID	08
USE CASE NAME	Manage request of Bools
ACTOR	Librarian
DESCRIPTION	Librarian can manage book request
PRE CONDITION	Should have internet connection
POST CONDITION	Librarian can manage book request
BASIC FLOW	Lib login and manage request
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.9 Search Books

USE CASE ID	09
USE CASE NAME	Search Books
ACTOR	Admin/Librarian/Student
DESCRIPTION	Each actor can search books
PRE CONDITION	Should have internet connection
POST CONDITION	Each actor can search for desired books
BASIC FLOW	login and search the books
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.10 Send Book Request

USE CASE ID	10
USE CASE NAME	Send request for Bools
ACTOR	Student
DESCRIPTION	Student can send book request
PRE CONDITION	Should have internet connection
POST CONDITION	Student can select the book request for it
BASIC FLOW	Student login and send request
EXCEPTIONAL FLOW	Invalid login details Internet connection error

Table 3.11 Logout

USE CASE ID	11
USE CASE NAME	Logout
ACTOR	Admin/Librarian/ Student
DESCRIPTION	Each actor can loin by signing up
PRE CONDITION	Each should have internet connection
POST CONDITION	Logged out
BASIC FLOW	Each can logged out by tapping the logout button

3.3.3 Activity Diagram

Activity diagram is another Important behavioral diagram in UML diagram to describe dynamic aspect of the system. Activity diagram is essentially an advanced version of flow & the Chart that modeling the flow from one activity to another activity.

Following are the notations used for activity diagrams:

Activity: The rounded rectangles represent activities that occur.

initial Node: The filled in circles is the starting point of the diagram. Final Node: The filled circle with a border is the ending point.

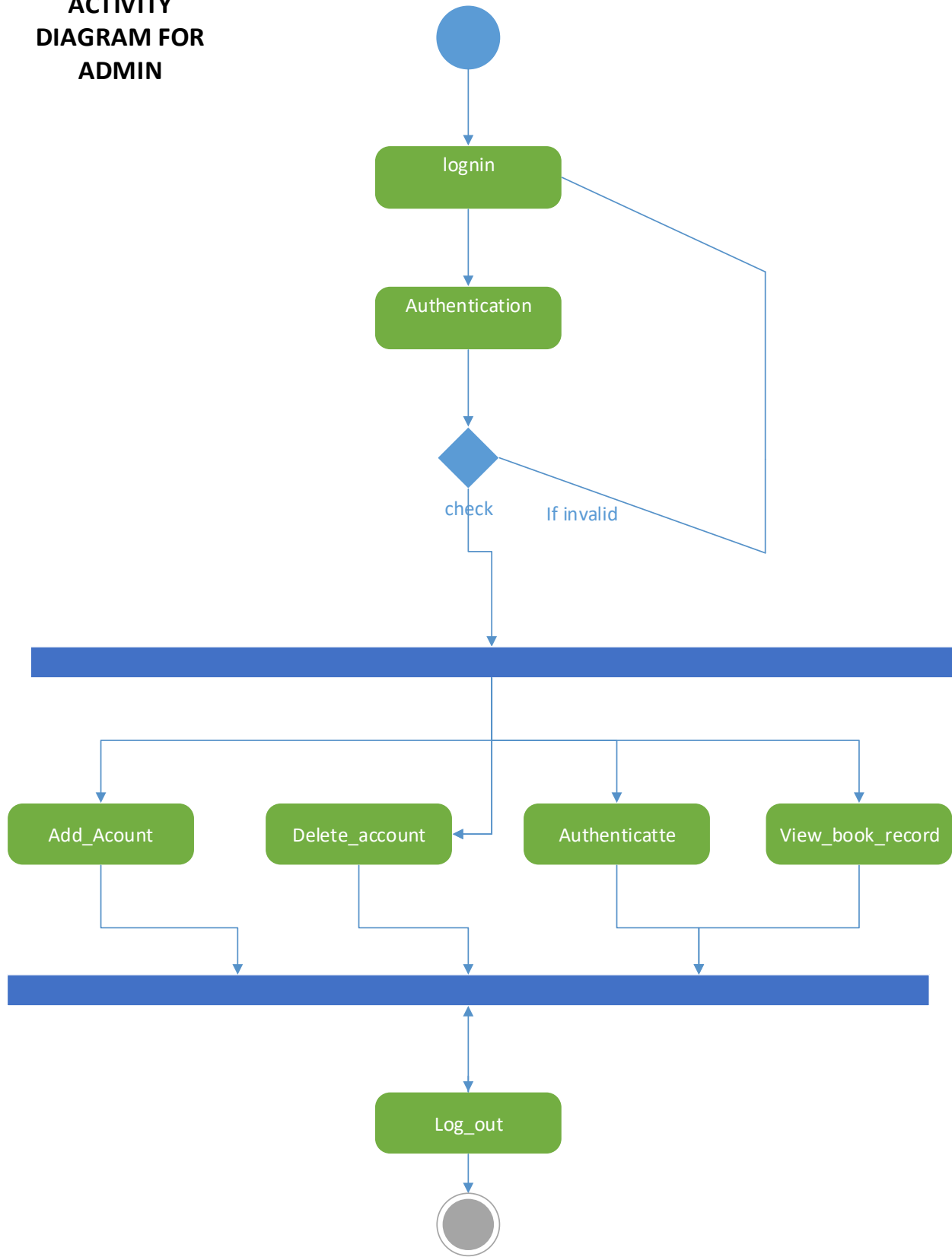**Fork**: A black bar with one flow going into it and several leaving it. Join: A black bar with several flows entering it and one leaving it.

Decision: A diamond with one flow entering and several leaving.

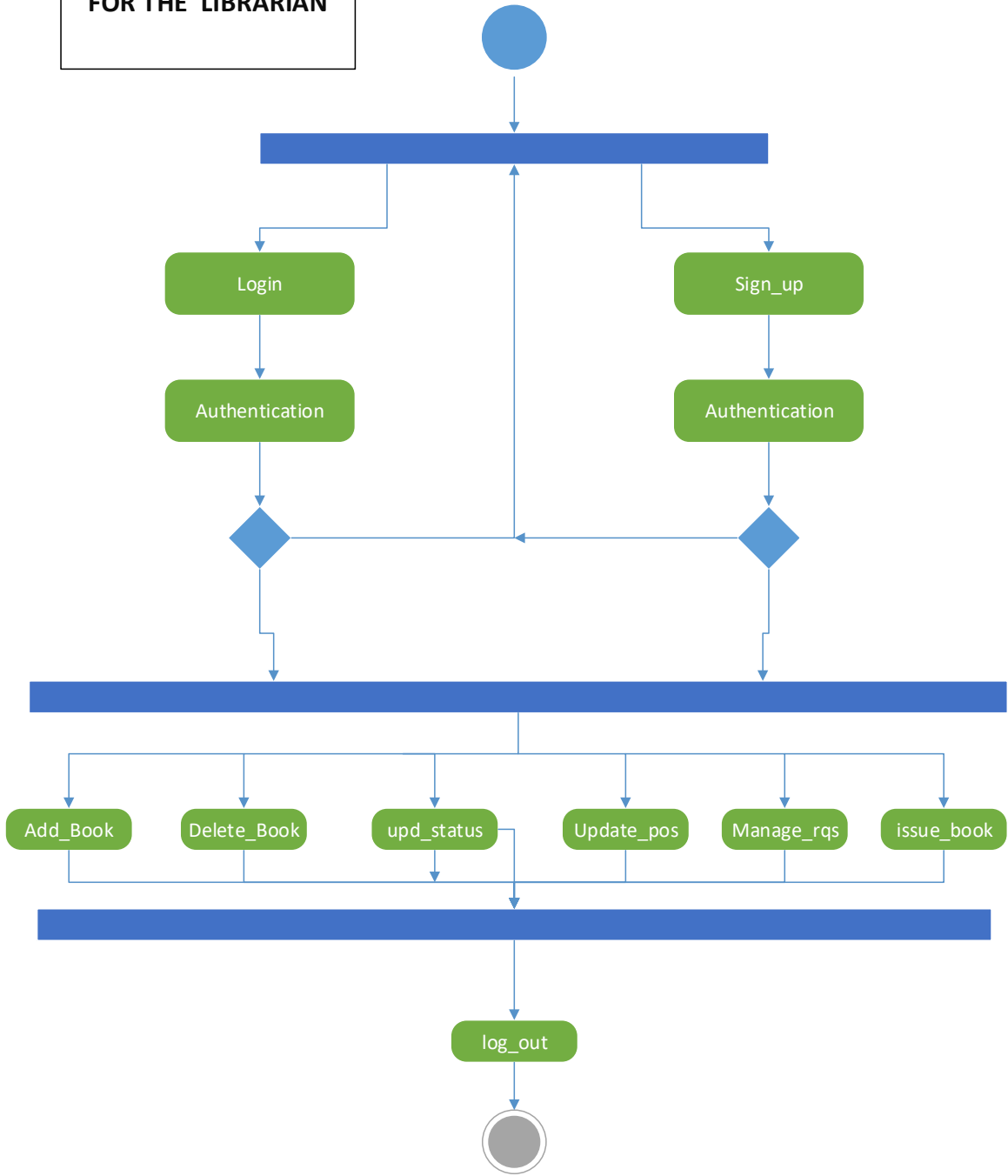**Merge:** A diamond with several flows entering and one leaving.

Flow Final: The circle with e X through it. This indicates that the process stops at this point.
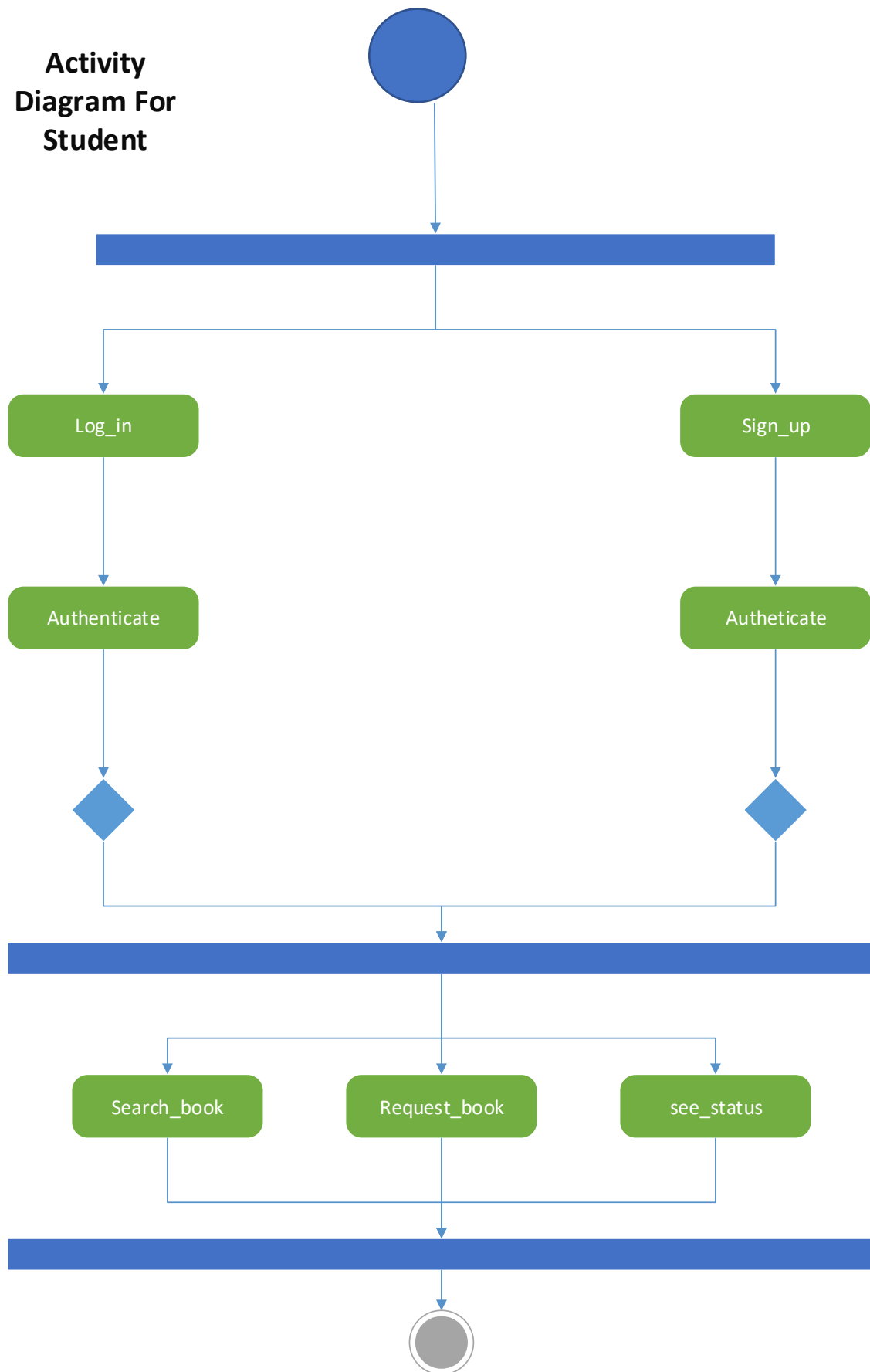
**ACTIVITY
DIAGRAM FOR
ADMIN**

ACTIVITY DIAGRAM FOR THE LIBRARIAN

Login

Sign_up

Authentication

Authentication

Add_Book

Delete_Book

upd_status

Update_pos

Manage_rqs

issue_book

log_out

Activity Diagram For Student

3.3.4 Class Diagram

In software engineering, a class diagram in the unified Modeling language is a type of static static structure diagram that describes the structure of a system by showing the systems classes ,their attributes, operations and the relationships among objects
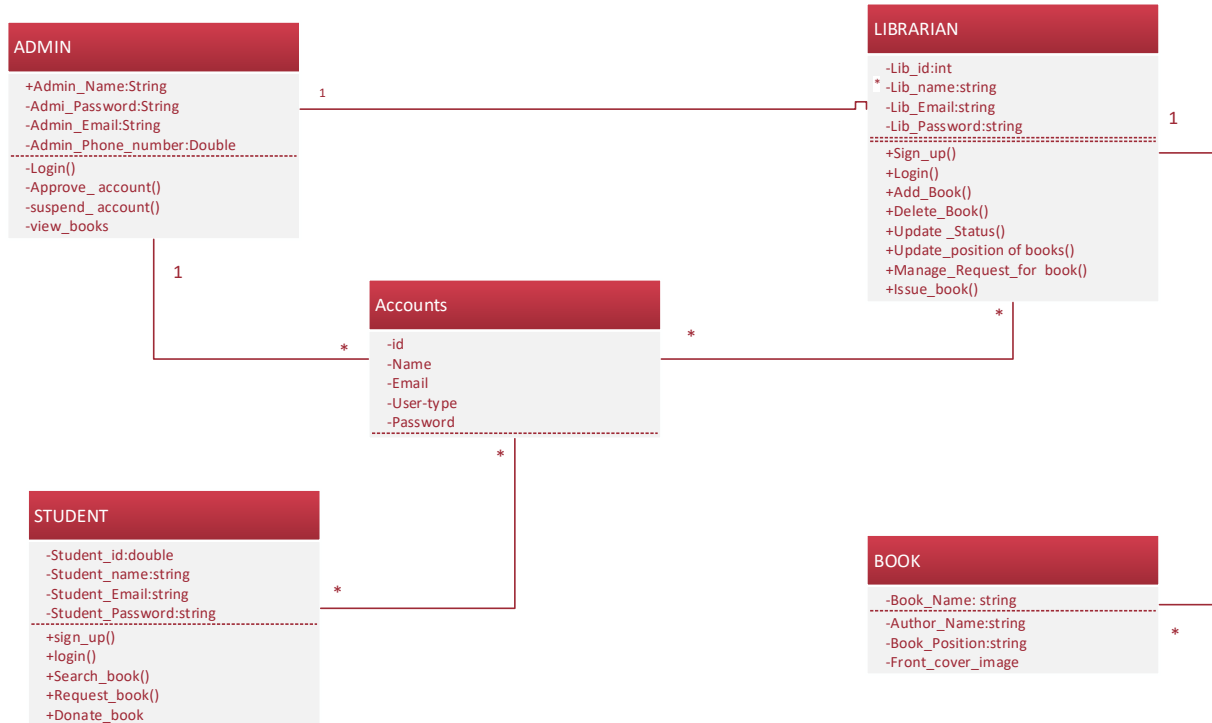**Purpose of class diagram Is:**

- Shows static structure of classifiers In a system
- Diagram provides basic notation for other structure diagrams prescribed by UML
- Helpful for developers and other team members too
- Business Analysts can use class diagrams to model systems from business perspective

A UML class diagram Is made up of:
- A set of classes and
- A set of relationships between classes

3.3.4.1 Class Diagram for the system

ADMIN

+Admin_Name:String
-Admi_Password:String
-Admin_Email:String
-Admin_Phone_number:Double

-Login()
-Approve_ account()
-suspend_ account()
-view_books

LIBRARIAN

-Lib_id:int
-Lib_name:string
-Lib_Email:string
-Lib_Password:string

+Sign_up()
+Login()
+Add_Book()
+Delete_Book()
+Update _Status()
+Update_position of books()
+Manage_Request_for book()
+Issue_book()

Accounts

-id
-Name
-Email
-User-type
-Password

STUDENT

-Student_id:double
-Student_name:string
-Student_Email:string
-Student_Password:string

+sign_up()
+login()
+Search_book()
+Request_book()
+Donate_book

BOOK

-Book_Name: string
-Author_Name:string
-Book_Position:string
-Front_cover_image

1

1

1

1

*

*

*

*

*

*

4.1 Introduction

implementation is the realization of an application, or execution of plan, idea, model, design, specification, standard, algorithm or policy. System implementation covers a broader scale of activities from a detailed work flow analysis to the formal go-live of the new system. After the designing phase of software design cycle, the next phase is system implementation. This phase is based on design specification. Up to now in software development life cycle, system has been planned, analyze and designed. Now it need to be developed which is usually known as coding.

4.2 Implementation

Implementation phase is very important in every aspect of business. Design phase only provide design but the actual product we get comes from development phase. From start to finish, starting other steps, first step is always needed to be completed. Until first step is not completed, the next step will never be started. Effective planning not only requires proper process management throughout the whole activities of SDLC like planning, analysis, design and implementation but it also requires proper execution of steps within the implementation phase. System implementation generally benefits from high levels of user involvement and management support.

Implementation is the fundamental phase in which actual code is being written to develop a system. Implementation phase includes code as well as extra issues that can be taken care of before implementing the system. This includes performance of hardware devices or tools that give support to developments or the problems faced during their installation. These changes must be recorded before or after applying them. For the implementation of this project, tools and platform are selected. Details are following

4.3 Platform Selection

| selected android platform for developing this application because android is an open source platform for developing applications and games for android user everywhere. Android also gives us tools for creating apps that look great and take advantage of the hardware capabilities available on each device. It automatically adapts our UI! to look its best on each device, while giving you as much control as you want over your UI on different device types.

4.3.1 Android Platform

Android is an open source operating system for mobile devices. Android was initially developed by Android Inc., and sold to Google in 2005. The goal was to develop an open mobile platform every developer to contribute towards improving the performance and features of the product. Android is built on top of Linux kernel and GNU software. Software Stack of the Android runs java applications using java core libraries. Each instance of java application runs on its own virtual machine (VM) called Dalvik. Android relies on Linux kernel to perform system level function such as memory management & even more dependent on it for hardware interactions and power management. Developers can build applications using the software development kit (SDK) developed by Google. It consists of Application Programming Interface (API) used to develop robust java applications. These AP\'s facilitate to access the contents on the phone such as contacts and calendar information and also integrate them with external web services in order to provide online services.

4.4 Language Selection
4.4.1 JAVA
I selected java for this project because java is one of the fastest languages around and also it has easy syntax that relatively easy to learn compare to the languages.

4.4.1.1 Why Android Choose Java
Android choose java mainly because:
- Android runs on many different hardware platforms. You would need to compile and optimize your native code for each of these different platforms to see any real benefits.
- There are a large number of developers already proficient in Java.
- Java has huge open source support, with many libraries and tools available to make developers life easier.

- Java protects you from many of the problems inherent In native code, like memory leaks, bad pointer usage, etc.
- Java allows them to create sandbox applications, and create a better security model So that one bad App can't take down your entire OS.
- Secondly, there is not Java Virtual Machine specified for Android. Instead all Java code is compiled for running on DaLvik, which is a lightweight, optimized VM specifically designed for running in mobile environments. it purportedly enhances battery life and maximizes efficient use of resources.

4.4.1.2 Features Of Java
The primary objective of Java programming language creation was to make it portable, simple and secure programming language. Apart from this, there are also some excellent features which play an important role in the popularity of this language

Java is Simple
The Java programming language is easy to learn. Java code Is easy to read and write.

Java is Familiar
Java is similar to C/C++ but it removes the drawbacks and complexities   of C/C++ tike pointers and multiple inheritances. So if you have background in C/C++, you will find Java familiar and easy to learn.

3.Java is an Object-Oriented programming language
Unlike C/C++ which is semi object-oriented, Java is a fully object-oriented programming language. It has all OOP features such as abstraction, encapsulation, inheritance and polymorphism.

4. Java supports Functional programming
Since Java SE version 8 (JDK 8), Java is updated with functional programming feature like functional interfaces and Lambda Expressions. This increases the flexibility of Java.

5. Java is Robust
With automatic garbage collection and simple memory management model (no pointers like C/C++}, plus language features like generics, try-with-resources,.. Java guides programmer toward reliable programming habits for creating highly reliable applications.

6. Java is Secure
The Java platform is designed with security features built into the language and runtime System such as static type-checking at compile time and runtime checking (security manager}, which let you creating applications that can't be invaded from outside. You never hear about viruses attacking Java applications.

7. Java Is High Performance
Java code is compiled into bytecode which Is highly optimized by the Java compiler, so that the Java virtual machine (JVM) can execute Java applications at full speed. In addition, compute-intensive code can be re-written in native code and interfaced with Java platform via Java Native interface (JNI) thus improve the performance.

8. Java Is Multithreaded
Java platform is designed with multithreading capabilities built into the language. That means you can build applications with many concurrent threads of activity, resulting in highly interactive and responsive applications.

9. Java is Platform Independence
java code is compiled into intermediate format (bytecode), which can be executed on any systems for which Java virtual machine is ported. That means you can write a Java

program once and run it on Windows, Mac, Linux or Solaris without re-compiling. Thus the slogan
"write once, run anywhere" of Java.
 Besides the above features, programmers can benefit from a strong and vibrant Java ecosystem

- Java is powered by Oracle - one of the leaders in the industry. Java also gets enormous support from big technology companies like IBM, Google, Redhat,... so it has been always evolving over the years.
- There are a lot of open source libraries which you can choose for building your applications.
- There are many superior tools and IDEs that makes your Java development easier.
- There are many frameworks that help you build highly reliable applications quickly.
- The community around Java technology is very big and mature, so that you can get support easily.

4.5 Database Selection

4.5.1 Fire Base

Firebase is Google's mobile application development platform that helps you build, improve, and grow your app.
Authentication — user login and identity
Realtime Database — real time, cloud hosted, NoSQL database
Cloud Firestore — real time, cloud hosted, NoSQL database
Cloud Storage — massively scalable file storage
Cloud Functions — "serverless", event driven backend
Firebase Hosting — global web hosting
ML Kit —SDK for common ML tasks

4.6 Microsor Word 2010

The Microsoft Word is use in my project for documentation of the whole 'Microsoft word is word is a Widely used commercial Word processor designed by Microsoft. we can purchased a component of Microsoft Office suite of a productively software but can also as a stand-alone Product. Simitar to other documents Paper, Some of the tools are spelling correction , grammar checker , word count, and newer versions include speech recognition. This program, like other programs can also make documents attractive like insert pictures make web pages graphs etc. you can also create tables. It also displays synonyms of the words. Can also print different ways.

4.7 Microsoft Visio 2016 : I used MS Visio for creating different types of diagrams for example :class diagram, activity diagram, use case diagram. Visio is a drawing and diagramming program.it contain pre drawn shapes and templated which are dragged over and dropped into specific program.

Chapter 5
Software Testing

5.1 Introduction

Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system Is Defect free. It involves execution of a software component or system component to evaluate one or more properties of interest. it can also be stated as the process of validating and verifying that a software program or application or product:
Meets the business and technical requirements that guided it's design and development
Works as expected
Can be implemented with the same characteristic.

5.2 Software Testing

Software testing is the process of ensuring the good quality of software. Testing is an evaluating process which is conducted by testing team testing whether the software behaves in the required manner or not. It validates. and verifies the software after development to avoid any future problems and issues.
Lets break down software testing into parts:

5.2.1 Process

Testing is a process rather than a single activity.

5.2.2 All Life Cycle Activities

Testing is a process that's take place throughout the Software Development Life cycle (SDLC).
The process of designing tests early in the life cycle can help to prevent defects from being introduced in the code. Sometimes it's referred as "verifying the test basis via the test design".

The test basis includes documents such as the requirements and design specifications.

5.2.3 Static Testing

It can test and find defects without executing code. Static Testing Is done during verification process. This testing includes reviewing of the documents (Including source code) and static analysis. This is useful and cost effective way of testing. For example: reviewing, walkthrough, Inspection, etc.

5.2.4 Dynamic Testing
In dynamic testing the software code is executed to demonstrate the result of running tests. it's done during validation process. For example: unit testing, integration testing, system testing, etc.

5.2.5 Planning
We need to plan as what we want to do. We control the test activities, we report on testing progress and the status of the software under test

5.2.6 Preparation
We need to choose what testing we will do, by selecting test conditions and designing test cases.

5.2.7 Evaluation
During evaluation we must check the results and evaluate the software under test and the completion criteria, which helps us to decide whether we have finished testing and whether the software product has passed the tests

5.2.8 Software products and related work products
Along with the testing of code the testing of requirement and design specifications and also the related documents like operation, user and training material is equally Important .

5.3 Testing Levels
Testing levels are basically used to identify missing areas and prevent overlap and repetition between the development life cycle phases. In software development life cycle models there are defined phases like requirements gathering and analysis, design, coding or implementation, testing and deployment. Each phase go through the testing. Hence there are various levels of testing.

The various levels of testing are:
- Unit testing
- Component testing
- Integration testing
- Component integration testing
- System integration testing
- System testing
- Acceptance testing
- Alpha testing
- Beta testing

5.4 Testing Techniques

Different testing techniques are used to test a system, which are as following:

5.4.1 Black box testing

The technique of testing without having any knowledge of the interior workings of the application is called black box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

5.4.2 White Box Testing

White-box testing is the detailed investigation of the internal logic and structure of the code. White box testing is also called open-box testing. For performing white box testing, tester needs to know the internal working of the system. The tester needs to have a look inside the source code and find out which unit of the code is behaving appropriately.

5.5 Test Cases

In software engineering, a test case is a set of conditions or variables under which a tester will determine if a requirement upon an application is partially or fully satisfied. In order to fully test that all the requirements of an application are met, there must be at least one test case for each requirement. A test has components that describe an input, action or event and an expected response, to determine if a feature of an application is working correctly Test cases can be described in a tabular form like this:

Table 5.1 Generic Test Cases

Test Case ID	Id of the test
Tester	Muhammad Ali Shahzad
Test Type	What technique is used
Test Case Name	Name of the test case
Description	Description of functional requirement
Procedure	Describe the steps of that function
Expected Result	What should it do?
Actual Result	What it did?
Status	Success or fail

5.6 Test Cases of Project

The test cases planned for testing this system are as follows:

Table 5.2 Test Case login

Test Case ID	01
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Login
Description	To check whether each of the user is login or not
Procedure	Users enter email, password and select user type
Expected Result	Each user is logged in
Actual Result	User successfully logged in
Status	Success

Table 5.3 Test Case sign up

Test Case ID	02
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Sign up
Description	Lib/student have to signup to register their accounts.
Procedure	Each user will enter first name , last name , email , password and phone number and then have to select whether he is librarian or student.
Expected Result	After authentication by admin both will login.
Actual Result	Success if data is valid.
Status	Success

Table 5.4 Test Case authentication

Test Case ID	03
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Authentication
Description	Purpose of this case to authenticate the accounts of new lib/student by the Admin.
Procedure	When lib/student register their accounts then their request will go towards admin and admin will decide that if the data is valid then he will approve the account otherwise suspended.
Expected Result	Depends on the given data.
Actual Result	If suspended then both user can see the home page but can perform any of the operation.
Status	Success

Table 5.5 Test Case add book

Test Case ID	04
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Adding books
Description	To check Librarian can add new books after successful login or not.
Procedure	Only lib can add books after login.
Expected Result	Books will be added.
Actual Result	New books will be added by lib.
Status	Success

Table 5.6 Test Case edit book

Test Case ID	06
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Edit book
Description	To check whether the lib can change the data about book or not.
Procedure	After successful login lib can edit the name, front cover etc, of the book
Expected Result	Books will be Edited.
Actual Result	New data will be saved into the data base.
Status	Success

Table 5.7 Test Case update status

Test Case ID	07
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Update position/status of the book
Description	To check whether the lib can change the status/position of books or not after successful login.
Procedure	Lib will login and the select the desired book to change the status whether it is free or not.
Expected Result	Status/position will be updated
Actual Result	Status/position will be updated
Status	Success

Table 5.8 Test Case manage book request

Test Case ID	08
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Manage book request
Description	Lib will manage the request of books by students
Procedure	Students will request the selective book if it is free and then request to lib to allow them to take the physically
Expected Result	Lib will receive the requests
Actual Result	Request will be managed .
Status	Success

Table 5.9 Test Case search books

Test Case ID	09
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Search books
Description	Admin/student/librarian will search for the book after successful login.
Procedure	After login each can search book.
Expected Result	Success
Actual Result	success
Status	Success

Table 5.10 Test Case sending book request

Test Case ID	10
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Send book request
Description	Student will search for the book after successful login and then select the desired book and send the request to the lib.
Procedure	After login student can send book request.
Expected Result	Request will be sent.
Actual Result	success
Status	Success

Table 5.11 Test Case logout

Test Case ID	11
Tester	Muhammad Ali Shahzad
Test Type	Black box test
Test Case Name	Logout
Description	Admin/student/librarian will logout when they want.
Procedure	By clicking the logout buton they can logout from the application
Expected Result	logout
Actual Result	success
Status	Success

Chapter 6
Interfaces

6.1 Interface for sign up

10:36

Register to OnlineLibrary

First Name   Last Name

Email Address

Password

Phone Number

+92 ▾   312xxxxxxx

Who you are?

Select who you are?

REGISTER

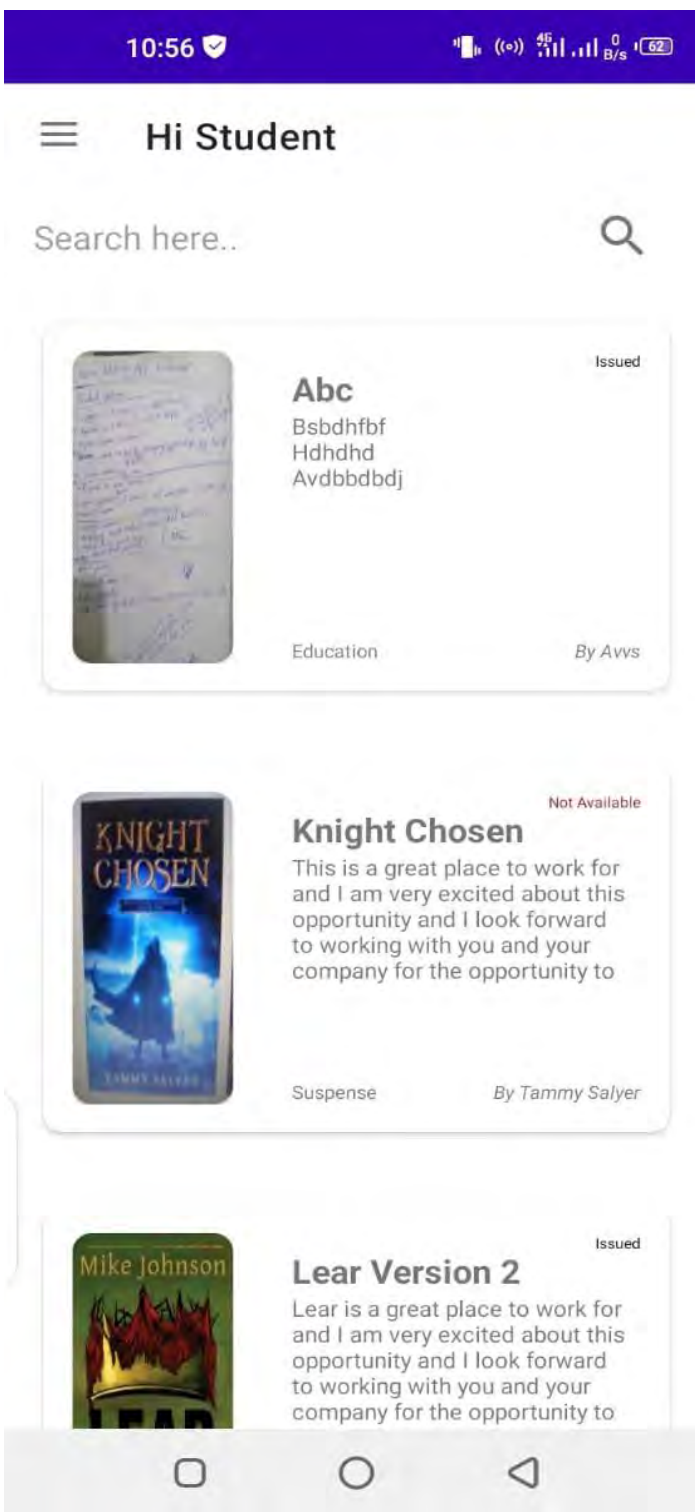Already have an account sign in

6.2 Interface for Login

6.3 Interface for Admin
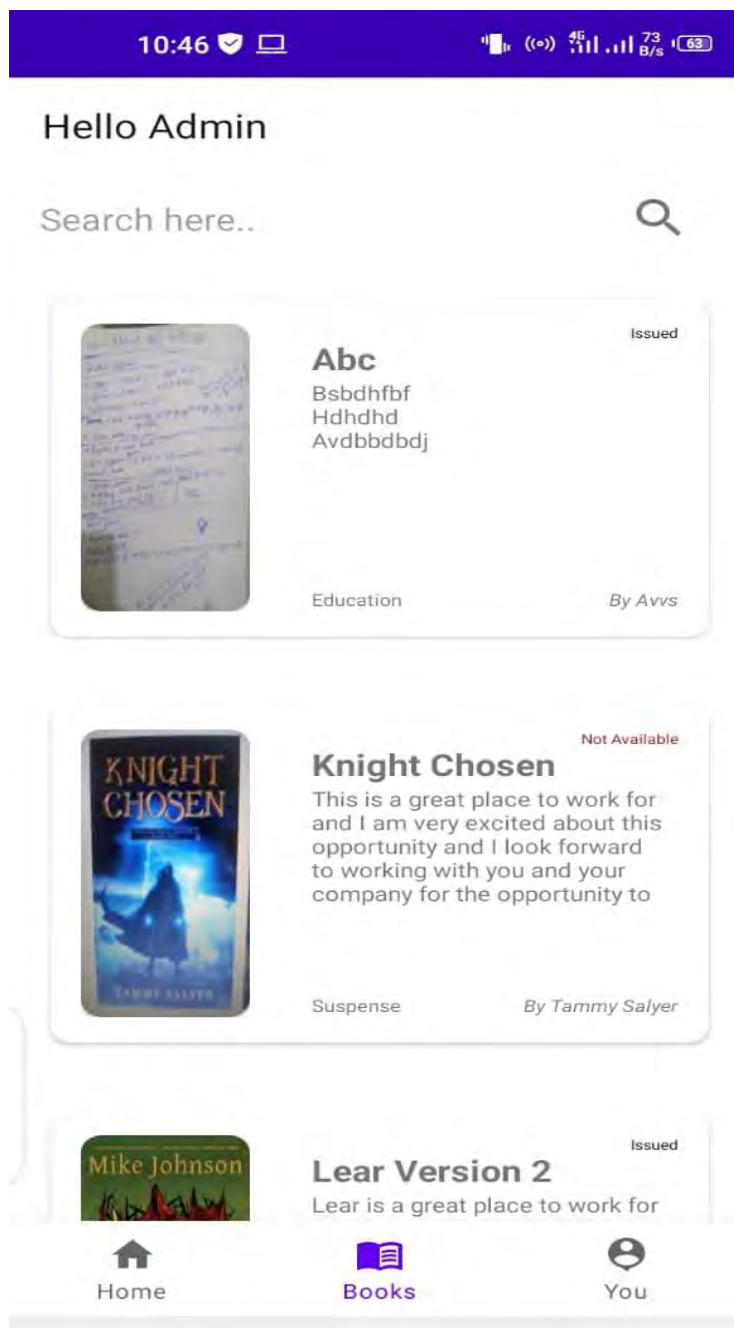
6.4 Interface for librarian
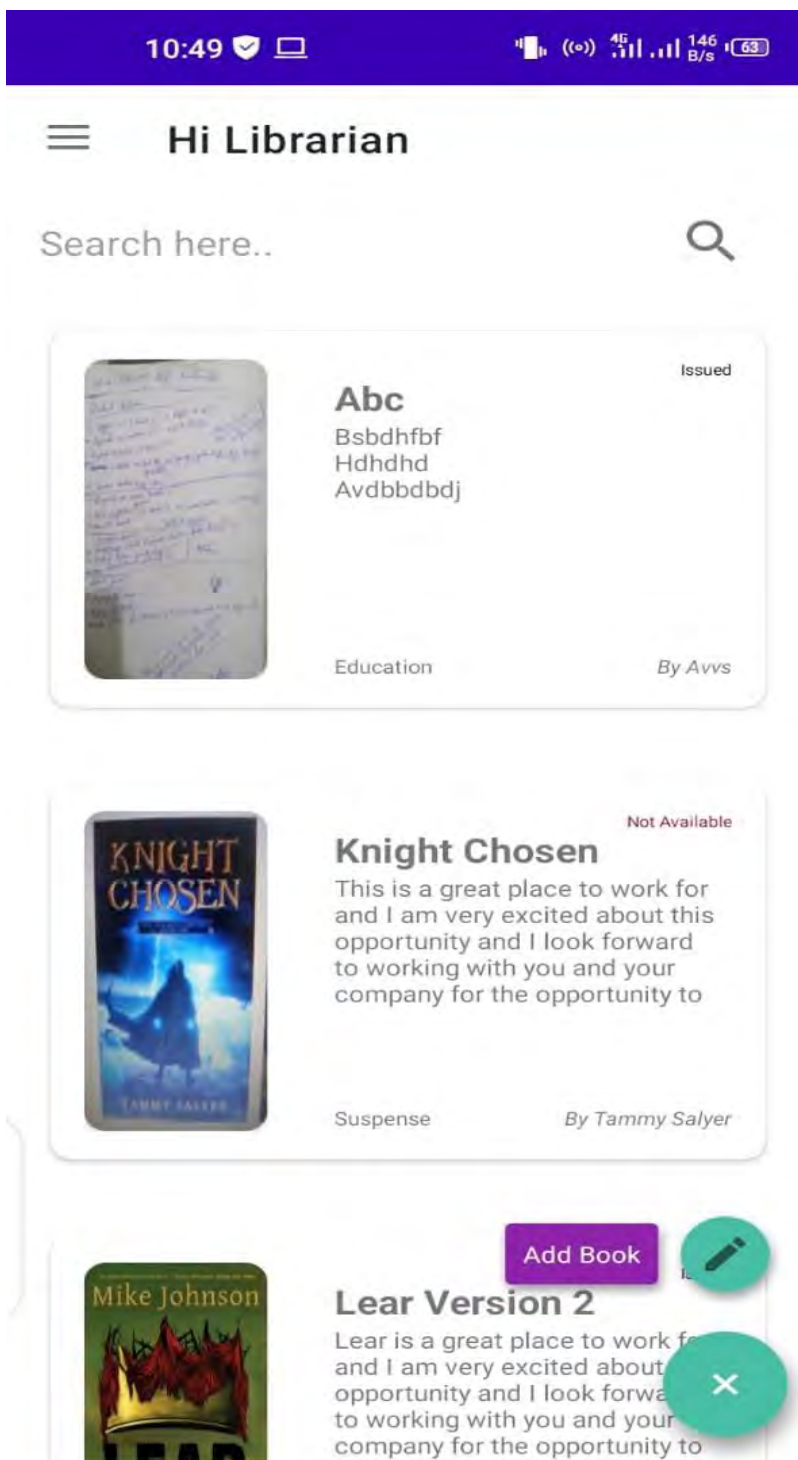
6.5 Interface for student

CHAPTER 7
CONCLUSION

7.1 Conclusion

Technology integration into instruction is beneficial for student achievement and learning and, in an ideal world, would be fully assimilated into the curriculum. Unfortunately, there are often significant barriers to incorporating such a program. Through planning, resourcefulness and instructional redesign, however, the determined media specialist can overcome these barriers. The media center is the ideal place in which to implement a more directed and cohesive technology program as it is the center of inquiry and research in a school. Additionally, as more and more teachers develop classroom libraries of books in their rooms, the media center becomes a place less defined by its books and more defined by the independent and collaborative intellectual behaviors that occur there, offering diverse perspectives and types of resources (Bush, 2009). In other words, the library is inherently a place of discovery, evaluation and information, so it is well suited to the implementation of an integrated technology program. Additionally, the quality of the technology integration is more important than the sheer presence of technology. The media specialist is well situated to offer advice and instruction in order to achieve a higher-quality curriculum that utilizes technology. Through this website, librarians and other educators have the opportunity to review the research supporting the use of technology. This can be used to campaign for grants or greater technology budgets from an administration. If this is unfeasible due to any of the hurdles mentioned in the Barriers section, the Overcoming Obstacles section is meant to serve as practical advice for the

typical school media specialist. Above all, when considering the integration of technology, it is important to focus on what type of instruction, pedagogy and schools best serve the needs of the students. Librarians often bemoan a lack of resources, administrative support, or time. However, through thoughtful dedication and a rethinking of current teaching methods, every librarian has the capability to create a truly digital, truly twenty-first century learning environment.

In the development of this system we uses android studio which is the best known SDK for android app development. Android Studio provides a complete IDE, including an advanced code editor and app templates. It also contains tools for development, debugging, testing, and performance that make it faster and easier to develop apps. ... You can also build production apps and publish apps on the Google Play store.