LIBRARY MANAGEMENT SYSTEM

PROJECT REPORT

18CSC202J/ 18AIC203J - OBJECT ORIENTED DESIGN AND PROGRAMMING LABORATORY

(2018 Regulation)

II Year/ III Semester

Academic Year: 2022 -2023

By

SHYAMLI (RA2111026010069)

ARYAN CHOUDHARY (RA2111026010077)

Under the guidance of

Dr. S. AMUDHA

Assistant Professor

Department of Computational Intelligence

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

Kattankulathur, Kancheepuram

NOVEMBER 2022

BONAFIDE

This is to certify that **18CSC202J - OBJECT ORIENTED DESIGN AND PROGRAMMING LABORATORY project report** titled "**LIBRARY MANAGEMENT SYSTEM "** is the bonafide work of **SHYAMLI (RA2111026010069), ARYAN CHAUDHARY (RA2111026010077)** who undertook the task of completing the project within the allotted time.

Signature of the Guide

Dr. S. Amudha

Assistant Professor

Department of CINTEL,

SRM Institute of Science and Technology

Signature of the II Year Academic Advisor

Professor and Head

Department of CINTEL

SRM Institute of Science and Technology

About the course:-

18CSC202J/ 8AIC203J - Object Oriented Design and Programming are 4 credit courses with **L T P C as 3-0-2-4** (Tutorial modified as Practical from 2018 Curriculum onwards)

Objectives:

The student should be made to:

- Learn the basics of OOP concepts in C++
- Learn the basics of OOP analysis and design skills.
- Be exposed to the UML design diagrams.
- Be familiar with the various testing techniques

Course Learning Rationale (CLR): The purpose of learning this course is to:

1. Utilize class and build domain model for real-time programs

2. Utilize method overloading and operator overloading for real-time application development programs
3. Utilize inline, friend and virtual functions and create application development programs
4. Utilize exceptional handling and collections for real-time object-oriented programming applications
5. Construct UML component diagram and deployment diagram for design of applications
6. Create programs using object-oriented approach and design methodologies for real-time application development

Course Learning Outcomes (CLO): At the end of this course, learners will be able to:

1. Identify the class and build domain model
2. Construct programs using method overloading and operator overloading
3. Create programs using inline, friend and virtual functions, construct programs using standard templates
4. Construct programs using exceptional handling and collections
5. Create UML component diagram and deployment diagram
6. Create programs using object oriented approach and design methodologies

COURSE ASSESSMENT PLAN FOR OODP LAB

S.No	List of Experiments	Course Learning Outcomes (CLO)	Blooms Level	PI	No of Programs in each session
1.	Implementation of I/O Operations in C++	CLO-1	Understand	2.8.1	10
2.	Implementation of Classes and Objects in C++	CLO-1	Apply	2.6.1	10
3,	To develop a problem statement. 1. From the problem statement, Identify Use Cases and develop the Use Case model. 2. From the problem statement, Identify the conceptual classes and develop a domain model with a UML Class diagram.	CLO-1	Analysis	4.6.1	Mini Project Given

4.	Implementation of Constructor Overloading and Method Overloading in C++	CLO-2	Apply	2.6.1	10
5.	Implementation of Operator Overloading in C++	CLO-2	Apply	2.6.1	10
6.	Using the identified scenarios, find the interaction between objects and represent them using UML Sequence diagrams and Collaboration diagrams	CLO-2	Analysis	4.6.1	Mini Project Given
7.	Implementation of Inheritance concepts in C++	CLO-3	Apply	2.6.1	10
8.	Implementation of Virtual function & interface concepts in C++	CLO-3	Apply	2.6.1	10
9.	Using the identified scenarios in your project, draw relevant state charts and activity diagrams.	CLO-3	Analysis	4.6.1	Mini Project Given
10.	Implementation of Templates in C++	CLO-3	Apply	2.6.1	10
11.	Implementation of Exception of Handling in C++	CLO-4	Apply	2.6.1	10
12.	Identify the User Interface, Domain objects, and Technical Services. Draw the partial layered, logical architecture diagram with UML package diagram notation such as Component Diagram, Deployment Diagram.	CLO-5	Analysis	4.6.1	Mini Project Given
13.	Implementation of STL Containers in C++	CLO-6	Apply	2.6.1	10
14.	Implementation of STL associate containers and algorithms in C++	CLO-6	Apply	2.6.1	10
15.	Implementation of Streams and File Handling in C++	CLO-6	Apply	2.6.1	10

LIST OF EXPERIMNENTS FOR UML DESIGN AND MODELLING:

To develop a mini-project by following the exercises listed below.

1. To develop a problem statement.
2. Identify Use Cases and develop the Use Case model.
3. Identify the conceptual classes and develop a domain model with UML Class diagram. 4. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence diagrams.
5. Draw relevant state charts and activity diagrams.

6. Identify the User Interface, Domain objects, and Technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.

Suggested Software Tools for UML:

StarUML, Rational Suite, Argo UML (or) equivalent, Eclipse IDE and Junit

ABSTRACT

Abstract must be a single paragraph in times new roman 14pt with a maximum of 300 words.

With the advancement of technology, it is imperative to exalt all the systems into a user-friendly manner. The Library Management system (LMS) acts as a tool to transform traditional libraries into digital libraries. In traditional libraries, the students/user has to search for books which are hassle process and there is no proper maintenance of database about issues/fines. The overall progress of work is slow and it is impossible to generate a fast report. The librarians have to work allotted for arranging, sorting books in the book sells. At the same time, they have to check and monitor the lend/borrow book details with its fine. It is a tedious process to work simultaneously in different sectors. LMS will assist the librarians to work easily. The LMS supports the librarians to encounter all the issues concurrently. The users need not stand in a queue for a long period to return/borrow a book from the library. The single PC contains all the data's in it. The librarians have to assess the system and provide an entry in it. Through LMS the librarian can find the book in the bookshelves. The LMS is designed with the basic features such as librarian can add/view/update/delete books and students' details in it. Once he/she ingress into the system they can modify any data's in the database. The complete model is developed in Dot net technology, the C# language is used to build the front end application whereas the SQL server is exploiting as database. The authorized person can only access the LMS system, they have to log in with their user id and password. As aforementioned that the LMS is designed in a user-friendly manner, so the admin can smoothly activate the system without expert advice. Every data is storing and retrieving from the SQL database so it is highly secure. Thus our system contributes its new approach towards the digital library setup.

MODULE DESCRIPTION

The project Library Management System aims at developing a fully functional computerized system to maintain all the day-to-day activity of a library This project has many features which such as the facility of user login and teacher's login. Also, on top of all this, there is an admin who will be managing the entire application's authorization and authentication, not no intruder can log in and modify the data, as a login for admin is also available.

The various modules of this system areas:

User Module:

This module is further divided into various sub-modules describing the user in a better way:

New user register:

To sign up a new user to this system.

Student Login:

So as to confirm that only an authenticated user is using the project.

Search book:

The user can search book based on book id, book name, or by author name.

Issue Book:

To help the user get the required books issued.

Return Book:

To return the book before the last date without fine, or after the specified time duration with a late fine.

Admin Module:

It is to be operated by the admin with a unique id and password. The admin is the person who decides authentication and authorization for all the different users of the application.

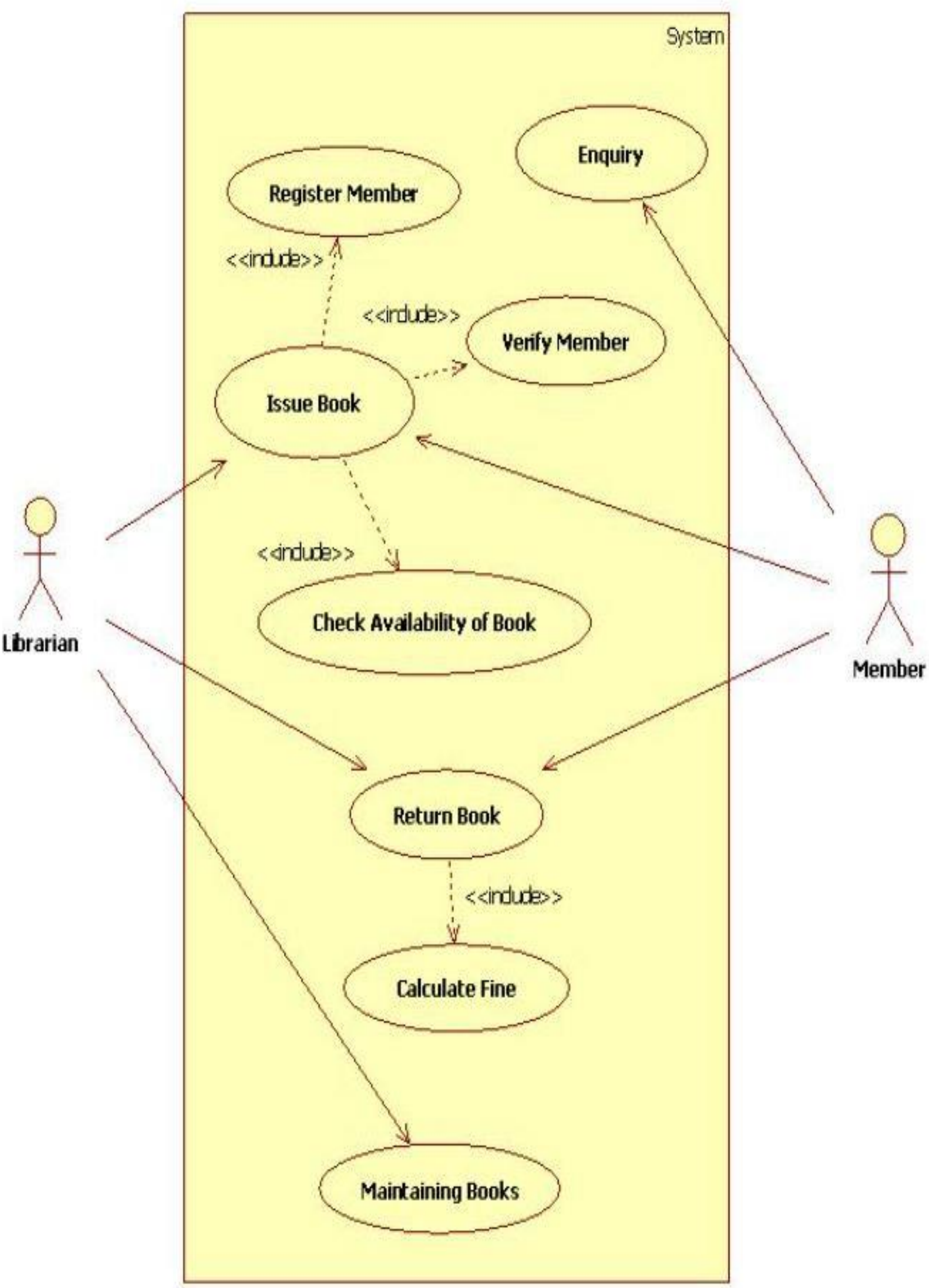
It further can be subdivided as:

- Register user.
- Issue Book.
- Maintain books in a stack, which means record the availability at a regular time interval.

Librarian:

Includes all the library staff who are required to enter the records in the system and keep an eye on the various activities like the issue of the book, the return of the book, non-availability of books, etc. through the developed system.

USECASE DIAGRAM WITH EXPLANATION



Use case diagrams referred as a behaviour model or diagram. It simply describes and displays the relation or interaction between the users or customers and providers of application service or the system. It describes different actions that a system performs in collaboration to achieve something with one or more users of the system. Use case diagram is used a lot nowadays to manage the system.

Here, we will understand the designing use case diagram for the library management system. Some scenarios of the system are as follows :

1. User who registers himself as a new user initially is regarded as staff or student for the library system.

    - For the user to get registered as a new user, registration forms are available that is needed to be fulfilled by the user.
    - After registration, a library card is issued to the user by the librarian. On the library card, an ID is assigned to cardholder or user.

2. After getting the library card, a new book is requested by the user as per there requirement.

3. After, requesting, the desired book or the requested book is reserved by the user that means no other user can request for that book.

4. Now, the user can renew a book that means the user can get a new due date for the desired book if the user has renewed them.

5. If the user somehow forgets to return the book before the due date, then the user pays fine. Or if the user forgets to renew the book till the due date, then the book will be overdue and the user pays fine.

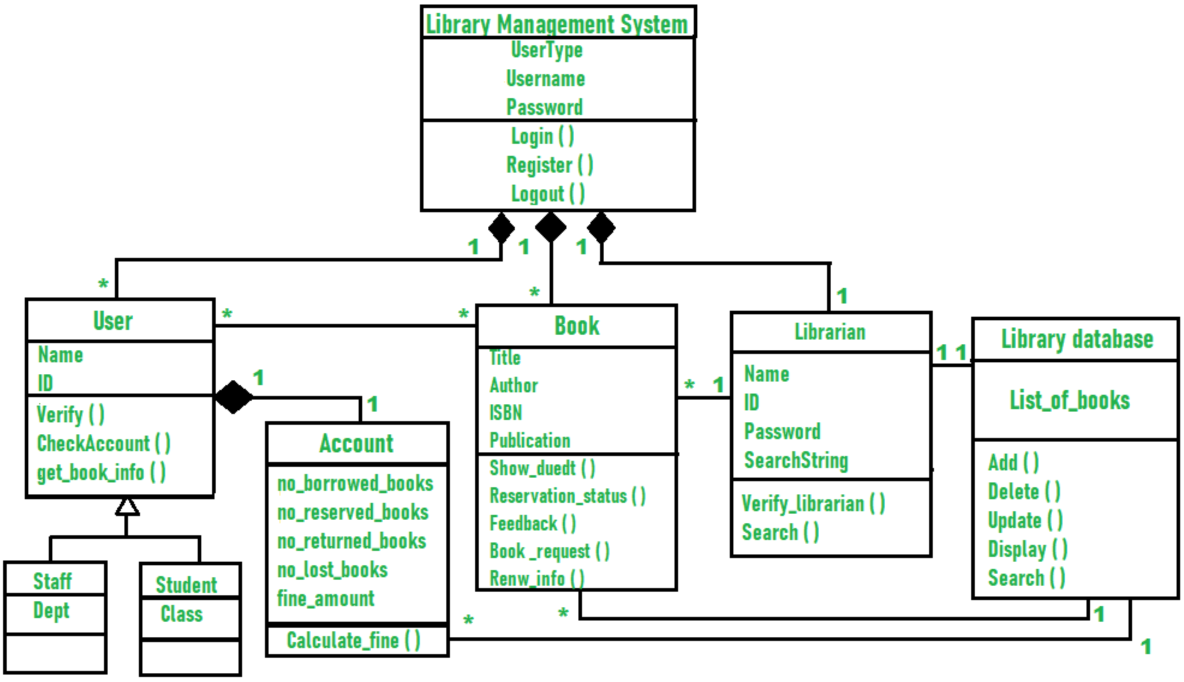
6. User can fill the feedback form available if they want to.

7. Librarian has a key role in this system. Librarian adds the records in the library database about each student or user every time issuing the book or returning the book, or paying fine.

8. Librarian also deletes the record of a particular student if the student leaves the college or passed out from the college. If the book no longer exists in the library, then the record of the particular book is also deleted.

9. Updating database is the important role of Librarian.

CLASS DIAGRAM WITH EXPLANATION



CLASS DIAGRAM FOR LIBRARY MANAGEMENT SYSTEM

Class Diagram for Library Management System simply describes structure of Library Management System class, attributes, methods or operations, relationship among objects.

Classes of Library Management System :

- **Library Management System class –**
  It manages all operations of Library Management System. It is central part of organization for which software is being designed.
- **User Class –**
  It manages all operations of user.
- **Librarian Class –** It manages all operations of Librarian.
- **Book Class –**
  It manages all operations of books. It is basic building block of system.
- **Account Class –**
  It manages all operations of account.
- **Library database Class –**
  It manages all operations of library database.
- **Staff Class –**
  It manages all operations of staff.

- **Student Class –**
  It manages all operations of student.

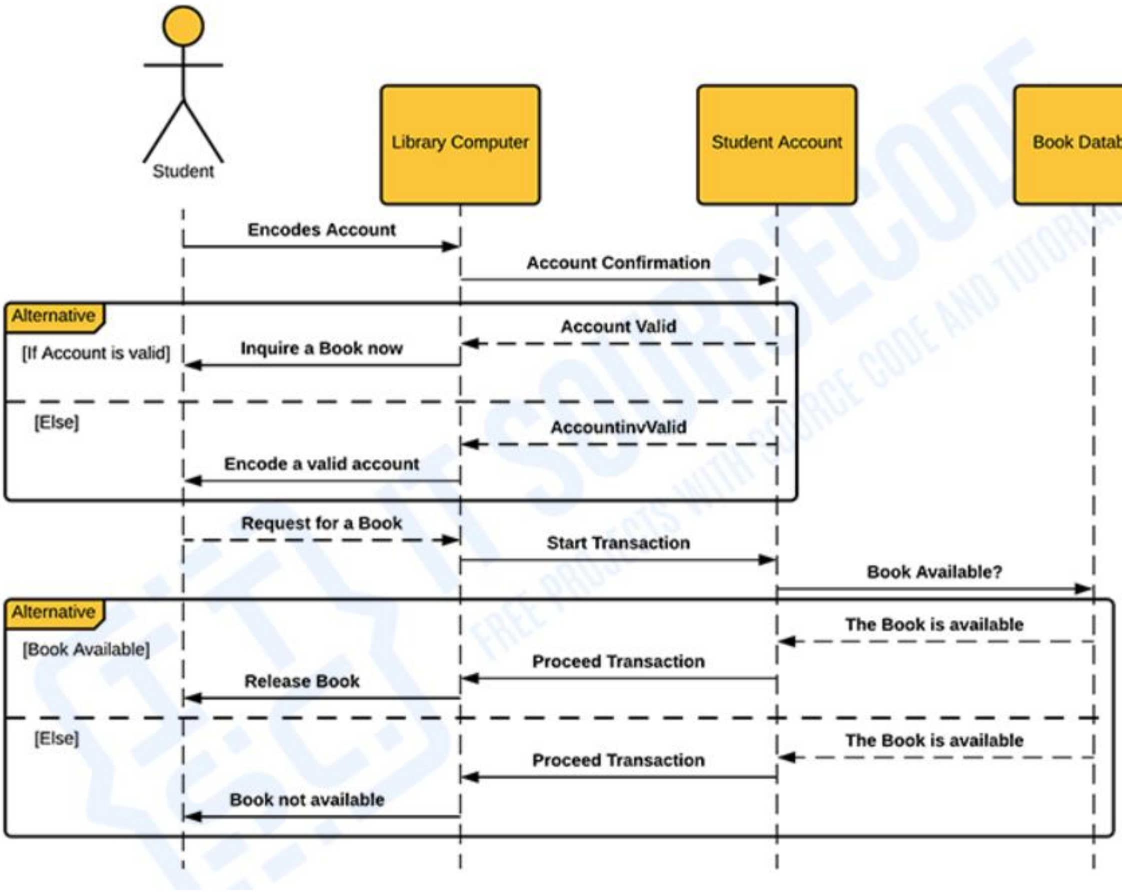
Attributes of Library Management System :

- **Library Management System Attributes –**
  UserType, Username, Password
- **User Attributes –**
  Name, Id
- **Librarian Attributes –**
  Name, Id, Password, SearchString
- **Book Attributes –**
  Title, Author, ISBN, Publication
- **Account Attributes –**
  no_borrowed_books, no_reserved_books, no_returned_books, no_lost_books fine_amount
- **Library database Attributes –**
  List_of_books
- **Staff Class Attributes –**
  Dept
- **Student Class Attributes –**
  Class

Methods of Library Management System :

- **Library Management System Methods –**
  Login(), Register(), Logout()
- **User Methods –**
  Verify(), CheckAccount(), get_book_info()
- **Librarian Methods –**
  Verify_librarian(), Search()
- **Book Methods –**
  Show_duedt(), Reservation_status(), Feedback(), Book_request(), Renew_info()
- **Account Methods –**
  Calculate_fine()
- **Library database Methods –**
  Add(), Delete(), Update(), Display(), Search()

SEQUENCE DIAGRAM WITH EXPLANATION

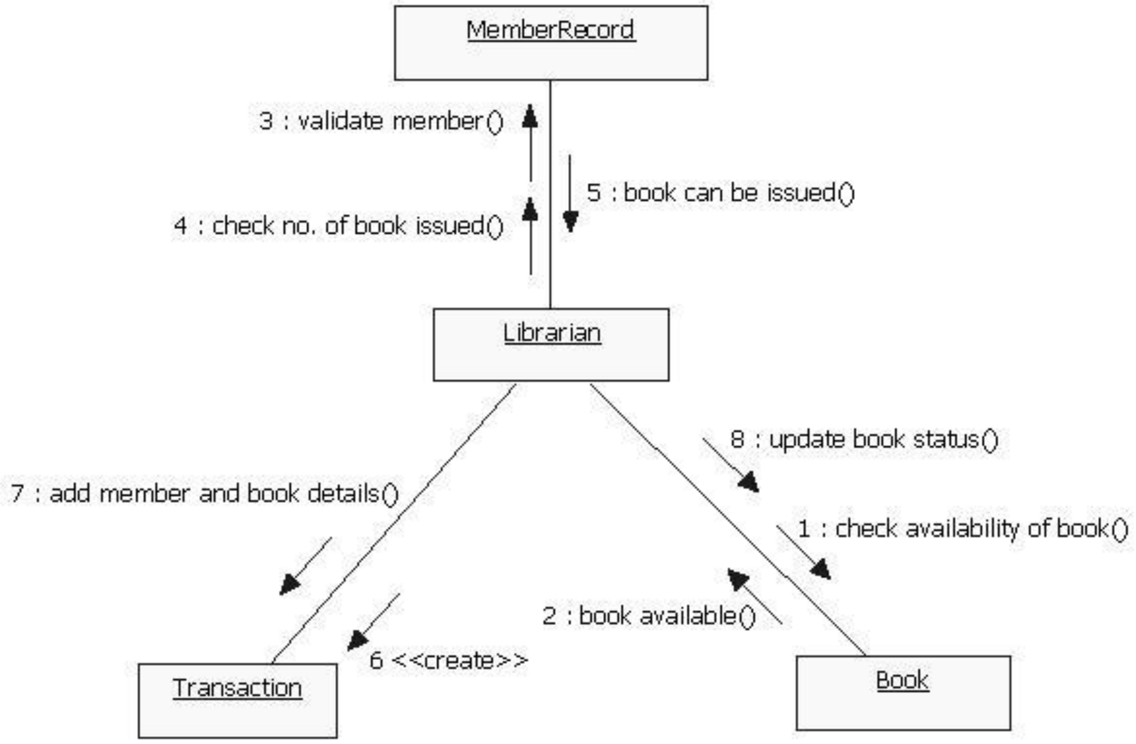


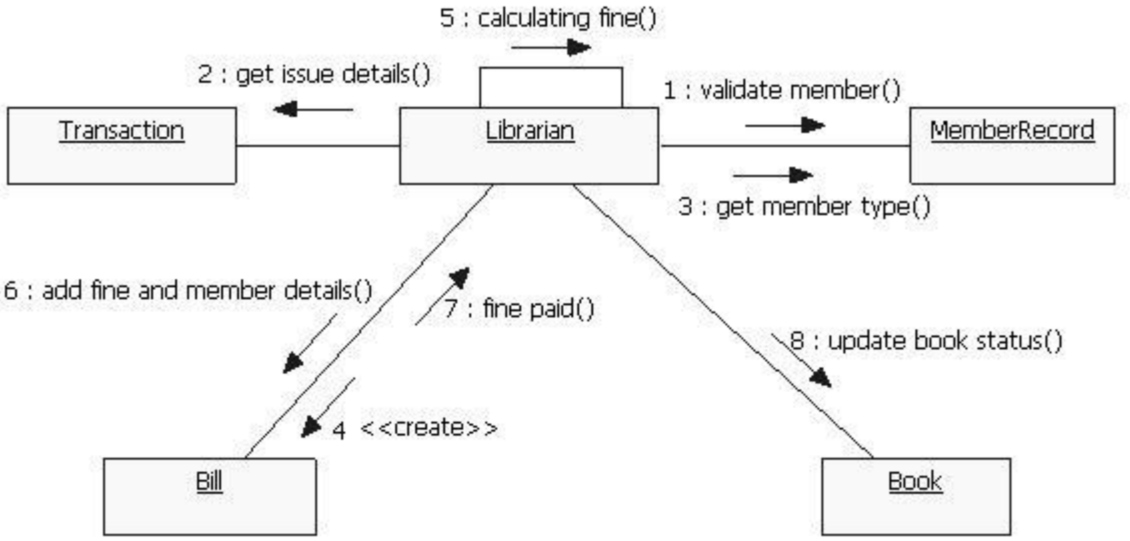
The diagram shows the series of messages that occurred to complete the task of managing Library. The diagram portrays a single scenario which is very common for library management system. This scenario is presenting the sequence of events and messages when a borrower borrows a book.

The sequence diagram given shows 4 objects which are: the student, library server (computer), account database, and book database. These objects were based on practical activities that happen in library management. The sequence of messages was plotted below the objects to determine how the process is being performed.The sequence diagram can be used to model the system’s interactions. These systems are also captured in a sequence diagram, which describes the flow from one object to another.

COMMUNICATION DIAGRAM WITH EXPLANATION

Communication diagram for issuing and returning a book-





A communication diagram is an extension of object diagram that shows the objects along with the messages that travel from one to another. In addition to the associations among objects, communication diagram shows the messages the objects send each other.

Purpose of Communication Diagram: -

- Model message passing between objects or roles that deliver the functionalities of use cases and operations

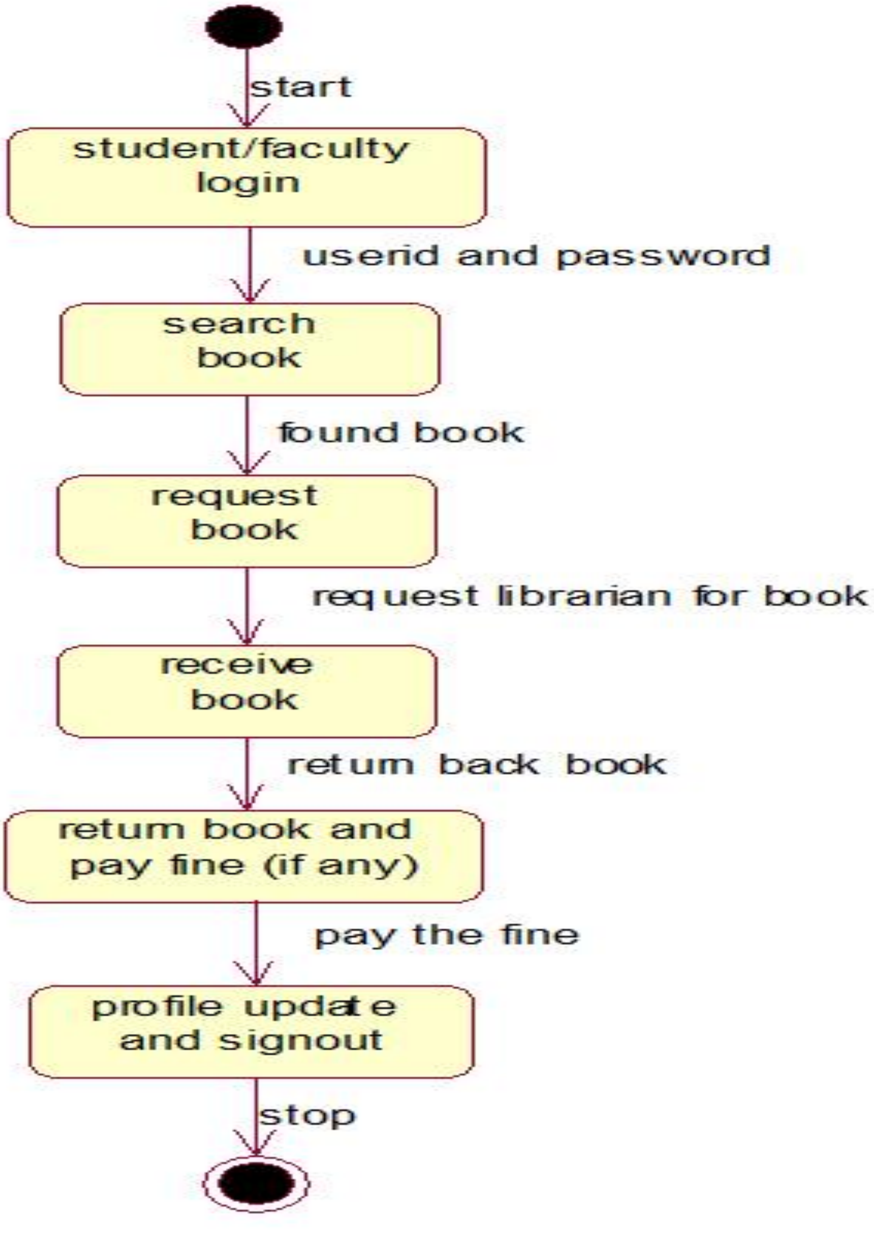
- Model mechanisms within the architectural design of the system

- Capture interactions that show the past messages between objects and roles within the collaboration scenario

- Model alternative scenarios within use cases or operations that involve the collaboration of different objects and interactions

- Support the identification of objects (hence classes), and their attributes (parameters of message) and operations (messages) that participate in use cases

STATE CHART DIAGRAM WITH EXPLANATION

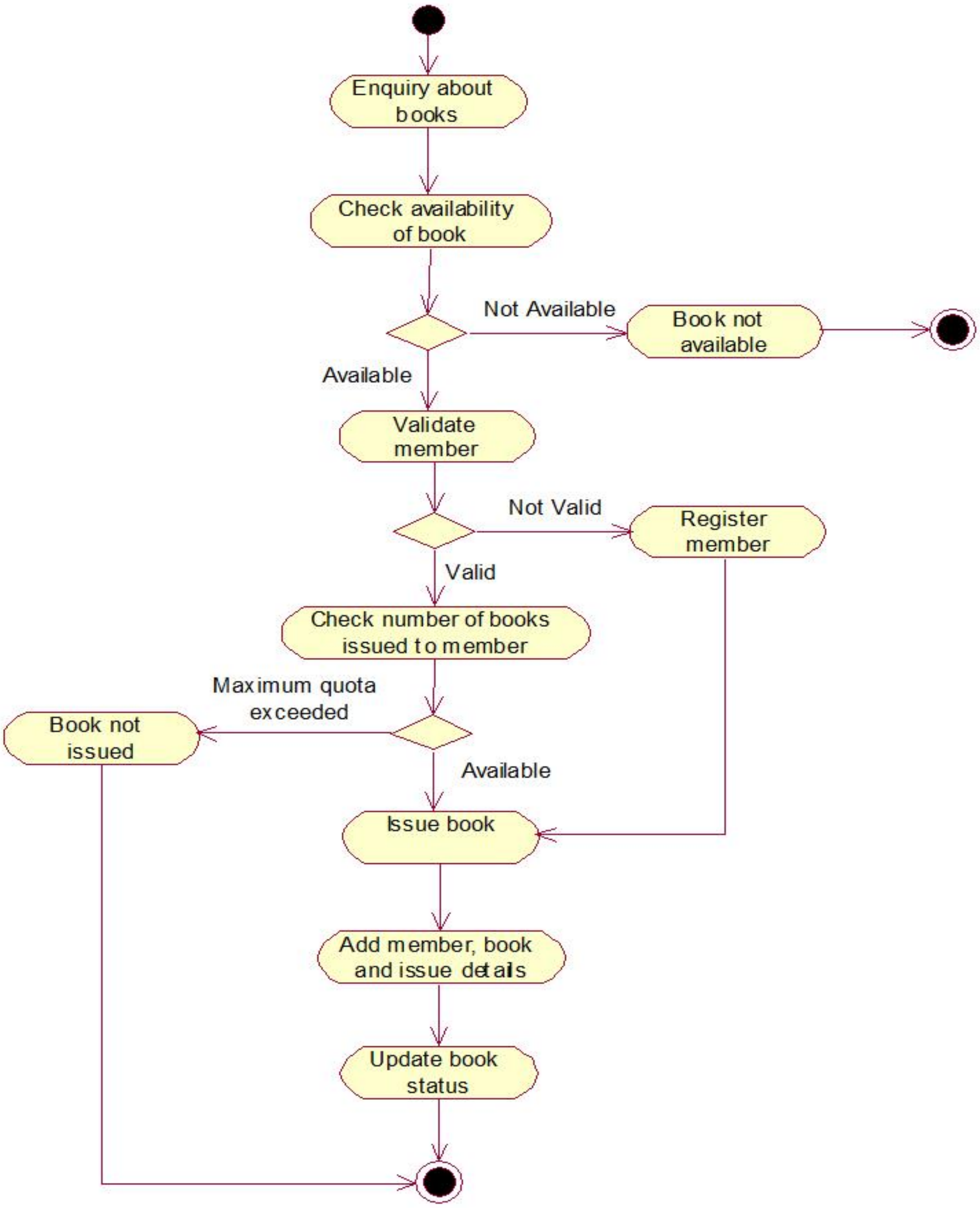


A State chart diagram describes a state machine. State machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events.

Purpose of State Chart Diagram: -

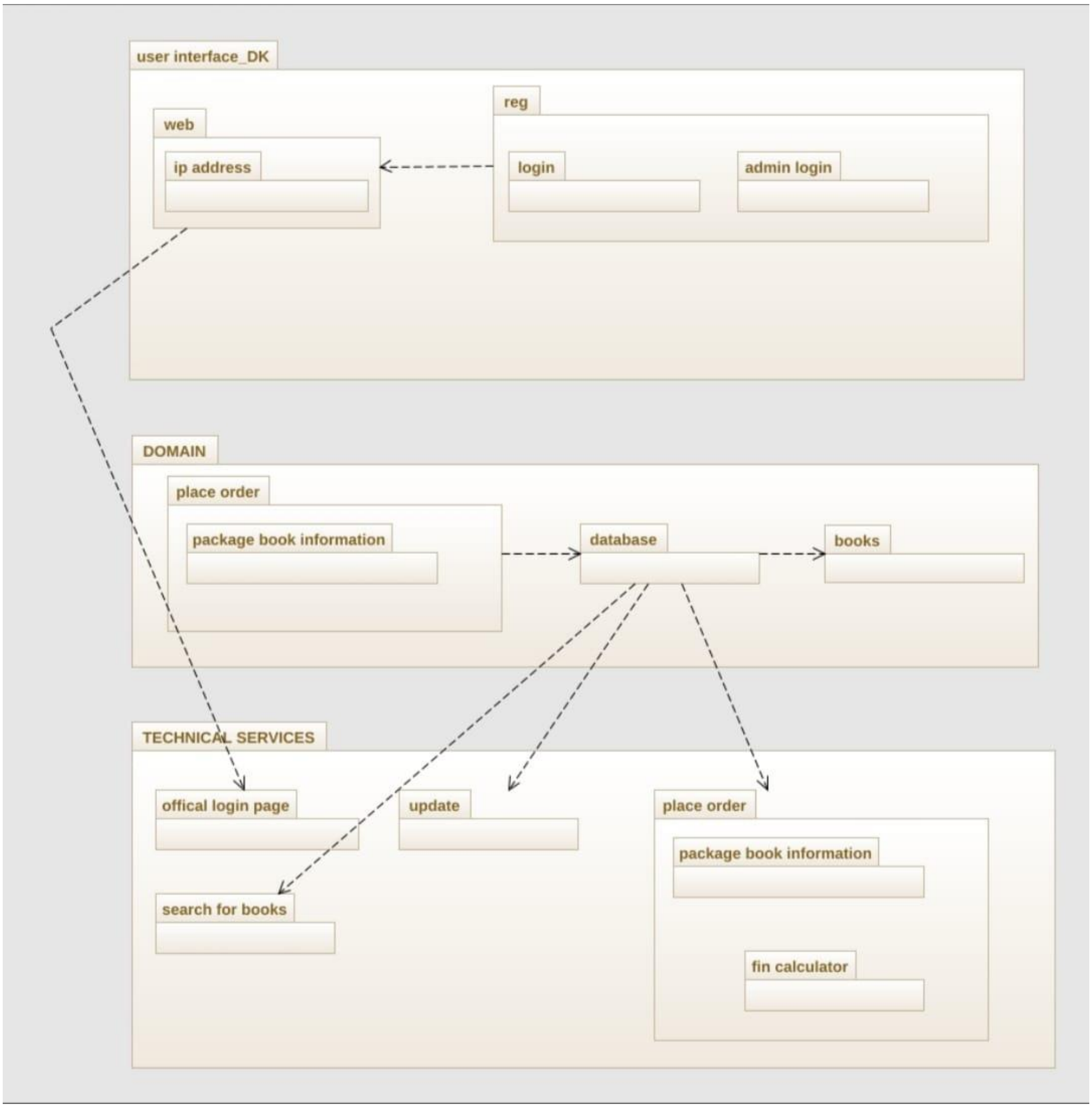
- To model the dynamic aspect of a system.
- To model the life time of a reactive system.
- To describe different states of an object during its life time.
- Define a state machine to model the states of an object

ACTIVITY DIAGRAM WITH EXPLANATION



The activity diagram used to describe flow of activity through a series of actions. Activity diagram is a important diagram to describe the system. The activity described as a action or operation of the system.

PACKAGE DIAGRAM WITH EXPLANATION



Package diagrams are structural diagrams used to show the organization and arrangement of various model elements in the form of packages. A package is a grouping of related UML Elements, such as diagrams, documents, classes, or even other packages. Each element is nested within the package, which is 32 depicted as a file folder within the diagram, then arranged hierarchically within the diagram.

Benefits of this diagram:

- They provide a clear view of the hierarchical structure of the various UML elements within a given system.

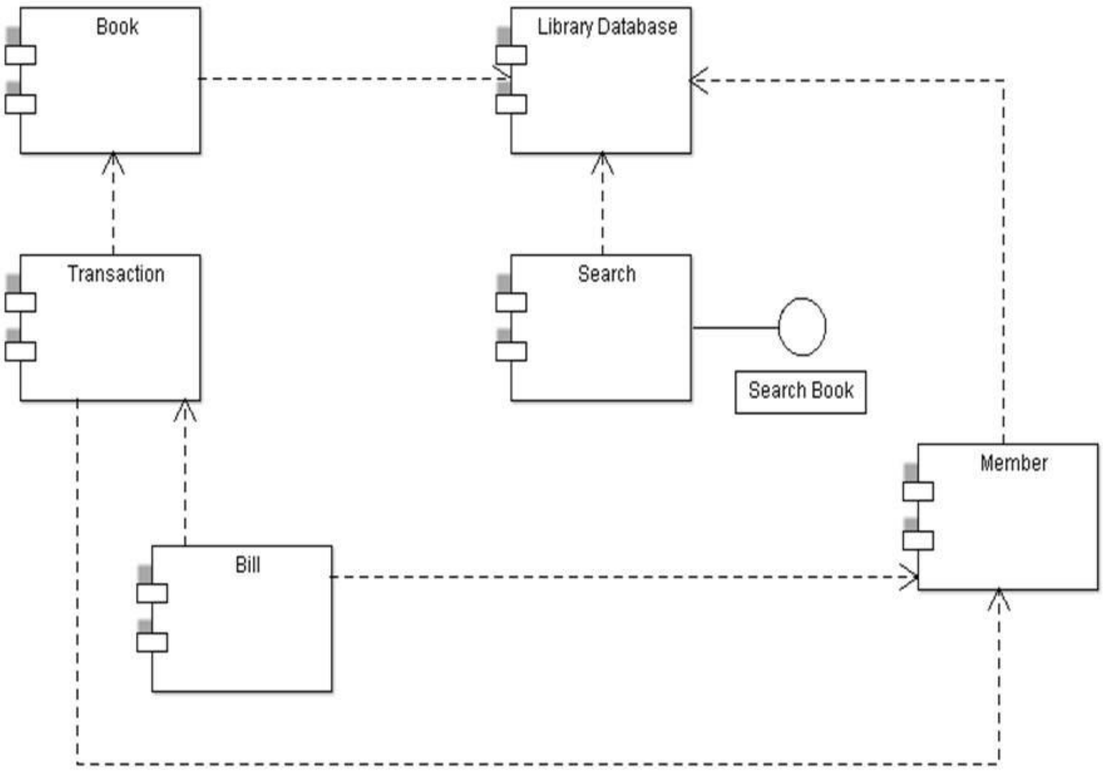
- These diagrams can simplify complex class diagrams into well-ordered visuals.

- They offer valuable high-level visibility into large-scale projects and systems.

- Package diagrams can be used to visually clarify a wide variety of projects and systems.

- These visuals can be easily updated as systems and projects evolve.

COMPONENT DIAGRAM WITH EXPLANATION



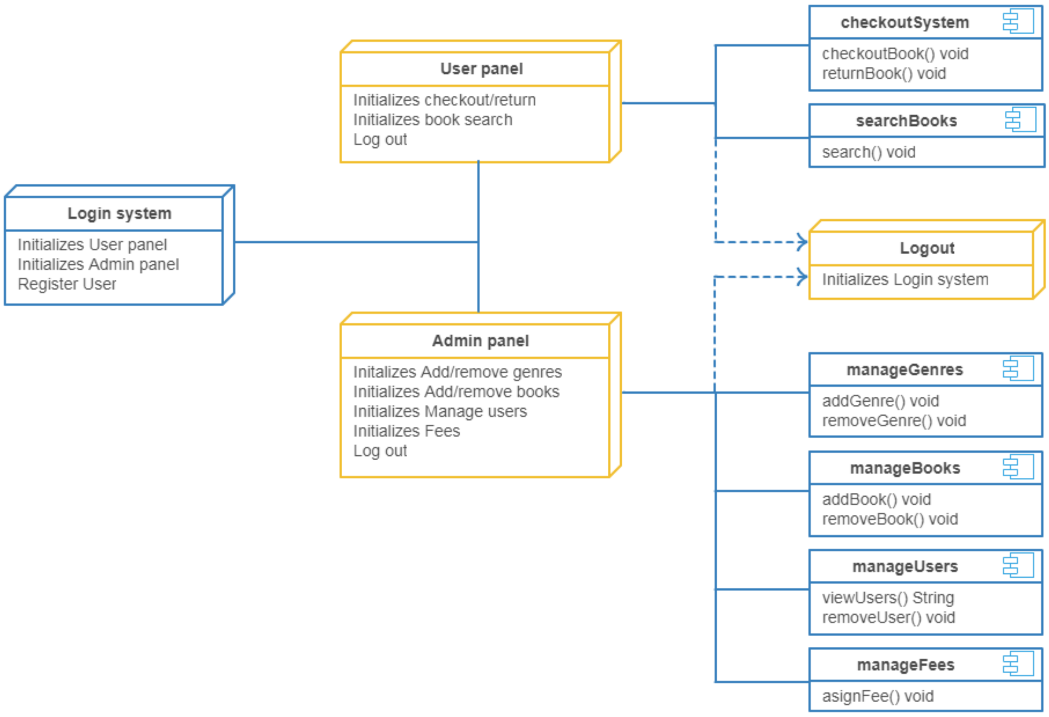
The **Library Management System UML component diagram** explains the sketch of the required software and hardware components and the dependencies between them. These components are labelled to clarify their part in the system's operation. They were represented by symbols that explain their function and role in the overall library management operation.

The component diagram of library management system has 6 components which are book, library database, transaction, search, bill and the member. The component "output" is the required interface which is dependent on the provided book database component. The included components were just based on the main function of the system.

Characteristics of Component Diagram:

- In component-based development, they are used to describe systems that have a service-oriented architecture.
- Shows how the code itself looks.
- It can be used to focus on the relationship between the parts while hiding the specifics.
- Help stakeholders understand how the system being built works and how it will be used.

DEPLOYMENT DIAGRAM WITH EXPLANATION



Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

The purpose of deployment diagrams can be described as –

- Visualize the hardware topology of a system.

- Describe the hardware components used to deploy software components.

- Describe the runtime processing nodes.

CONCLUSION –

Our project is only a humble venture to satisfy the needs to manage their project work. The objective of software planning is to provide a frame work that enables the manager to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

At the end it is concluded that we have made effort on following points…

- A description of the background and context of the project and its relation to work already done in the area.

- Made statement of the aims and objectives of the project.

- We define the problem on which we are working in the project.

- We have made all the UML diagrams which will help explain all the components of the system in detail

REFRENCES - https://www.startertutorials.com/uml/uml-diagrams-library-management-system.html

https://www.geeksforgeeks.org/class-diagram-for-library-management-system/amp/

https://itsourcecode.com/uml/library-management-system-project-uml-diagrams/