SRM INSTITUTE OF SCIENCE AND TECHNOLOGY FACULTY OF SCIENCE AND HUMANITIES DEPARTMENT OF COMPUTER SCIENCE



PRACTICAL RECORD NOTE

STUDENT NAME :

REGISTER : RA23320030100

NUMBER

CLASS : M.SC-IT

YEAR & SEMESTER

: I Year & I Semester

SUBJECT CODE : PIT21C103J

SUBJECT TITLE : SOFTWARE ENGINEERING

OCT 2023



**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY FACULTY OF SCIENCE AND HUMANITIES DEPARTMENT OF COMPUTER SCIENCE**

SRM Nagar, Kattankulathur – 603 203

# CERTIFICATE

*Certified to be the bonafide record of practical work done by*

Register No. *of Degree course*

*For* **PIT21C103J- SOFTWARE ENGINEERING** *in the*

*Computer Science lab in SRM Institute of Science and Technology during the*

*academic year 2023-2024.*

Staff In-charge Head of the Department

Submitted for Semester Practical Examination held on \_ .

Internal Examiner External Examine

|  |  |  |  |
| --- | --- | --- | --- |
| EX.NO | DATE | INDEX | SIGNATURE |
| **1.** |  | **SYSTEM REQUIREMENT SPECIFICATION** |  |
| 1.1 |  | SRS FOR COLLEGE AUTOMATION SYSTEM |  |
| 1.2 |  | SRS FOR LIBRARY INFORMATION SYSTEM |  |
| 1.3 |  | SRS FOR BANKING MANAGEMENT SYSTEM |  |
| 1.4 |  | SRS FOR RAILWAY TICKET RESERVATION SYSTEM |  |
| **2.** |  | **DATA FLOW DIAGRAM** |  |
| 2.1 |  | DFD FOR COLLEGE AUTOMATION SYSTEM |  |
| 2.2 |  | DFD FOR LIBRARY INFORMATION SYSTEM |  |
| 2.3 |  | DFD FOR BANKING MANAGEMENT SYSTEM |  |
| 2.4 |  | DFD FOR RAILWAY TICKET RESERVATION SYSTEM |  |
| **3.** |  | **DEVELOP CLASS DIAGRAM** |  |
| 3.1 |  | DCD OF COLLEGE AUTOMATION SYSTEM |  |
| 3.2 |  | DCD OF LIBRARY INFORMATION SYSTEM |  |
| 3.3 |  | DCD OF BANKING MANAGEMENT SYSTEM |  |
| 3.4 |  | DCD OF RAILWAY TICKET RESERVATION SYSTEM |  |
| 4. |  | **DEVELOP UML USECASE MODEL FOR A PROBLEM** |  |
| 4.1 |  | USECASE MODEL OF COLLEGE AUTOMATION SYSTEM |  |
| 4.2 |  | USECASE MODEL OF LIBRARY INFORMATION SYSTEM |  |
| 4.3 |  | USECASE MODEL OF BANKING MANAGEMENT SYSTEM |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 4.4 |  | USECASE MODEL OF RAILWAY TICKET RESERVATION SYSTEM |  |
| **5.** |  | **SEQUENCE DIAGRAM** |  |
| 5.1 |  | SEQUENCE DIAGRAM OF COLLEGE AUTOMATION SYSTEM |  |
| 5.2 |  | SEQUENCE DIAGRAM OF LIBRARY INFORMATION SYSTEM |  |
| 5.3 |  | SEQUENCE DIAGRAM OF BANKING MANAGEMENT SYSTEM |  |
| 5.4 |  | SEQUENCEDIAGRAM OF RAILWAY TICKET RESERVATION SYSTEM |  |
| **6.** |  | **ER DIAGRAM** |  |
| 6.1 |  | ER DIAGRAM OF COLLEGE AUTOMATION SYSTEM |  |
| 6.2 |  | ER DIAGRAM OF LIBRARY INFORMATION SYSTEM |  |
| 6.3 |  | ER DIAGRAM OF BANKING MANAGEMENT SYSTEM |  |
| 6.4 |  | ER DIAGRAM OF RAILWAY RESERVATION SYSTEM |  |

**1.SYSTEM REQUIREMENT SPECIFICATION**

**1.1 SYSTEM REQUIREMENT SPECIFICATION FOR COLLEGE AUTOMATION SYSTEM**

**INTRODUCTION**

The title of the project is COLLEGE MANAGEMENT SYSTEM. CMS is an Internet based application that aims at providing information to all the levels of management within an organization. This system can be used as a information management system for the college. For a given user, the administrator will create a loginid & password, using this user can access the system to either upload or download some information from the database.

The purpose of this document is to outline the requirements for the development of a College Automation System. This system aims to streamline and automate various processes within the college to enhance efficiency, accuracy, and overall management.

**PURPOSE**

The purpose of this document is to present a detailed description of the College Management System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is

intended for both the client and the developers of the system and will be proposed to the Administrative head for its approval.

**PROJECT SCOPE AND PRODUCT FEATURES**

This software system will be a College management system for a the members of an organization. This system will be

designed to maximize the administrative, academic and overall productivity by providing tools to assist in automating the technical procedures and proccesses, which would otherwise have to be performed manually. By maximizing the users work efficiency and production the system will meet the users needs while remaining easy to understand and use.

It is a user-friendly portal to interact, manage, access the information.

**OVERALL DESCRIPTION**

**PRODUCT PERSPECTIVE**

The product will be a standalone application and may be run on multiple systems within an Internet network. The product will require a keyboard, mouse and monitor to interface with the users. The minimum hardware requirements for the product are specified in this document

**USER CLASSES AND CHARACTERISTICS**

The target audience for CMS product is the college Administrator/students/faculty/staff (Technical/Nontechnical)

* Administrator The Super user of the system. Mainly focuses on administratiive and academic related issues.
* Student A user with limited access rights.
* Staff A user of the system who has more access rights than a normal user.

**CONSTRAINTS**

The current constraints on the project are related to the provision of hardware resources and software resources.

* At present, we have a i3 gen4 intel core processor running on top of the Linux/windows operating system.
* The documents will be present only in pdf format.
* In the feedback forms, the replies will not be frequent and the petitioner will not be anonymous.
* There will not be any moderater to filter out the fake complains with the genuine ones. The superuser have to do it himself manually.
* The Internet connection is also a constraint for the application. Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function.
* The web portal will be constrained by the capacity of the database. Since the database may be forced to queue incoming requests and therefor increase the time it takes to fetch data.
* Mess Rebate Will at least of 3days.
* Registration will be open for short time.
* All Document should be in .Zip file.
* College will provide funds for SMS service if SMS service is not free.
* After submitting the course evaluation form, the user cannot revert his or her actions.
* The user cannot change his/her all personal or academic details. He/she first have to get permission from the super user to do so.

**SPECIFIC REQUIREMENTS**

This section contains all the software requirements at a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements. Throughout this section, every stated requirement should be externally perceivable by users, administrator, or, other external systems.

**EXTERNAL INTERFACES Client:**

* Hardware platform:
* PIII or above with
* RAM of 512 or above MB - HardDisk 20GB or above GB
* Software Platform: Browser :
* Mozilla Fire-Fox v12.0 or higher
* Google Chrome v27.0.1453.116m or higher. Server:

**User Interface:**

A first-time user of the web portal should see the log-in page when he/she opens the portal. If the user has not registered, he/she should be able to do that on the log-in page. It will also have a remember me button.If the user is not a first- time user, he/she should be able to see the dashboard which contains different domains like academics, Hostel, Profile, Mess, Transport.A news bulliten, some general information, list of holidays and different timetables will also be visible on this page.Every user should have a profile page where they can edit their e-mail address, phone number and password and other personal details.

**Communications interface:**

The communication between the client and the server will be done through internet.

**FUNCTIONAL REQUIREMENTS**

This section includes the requirements that specify all the fundamental actions of the software system

**LOGIN**

This section contains students login menu where students have to login by their username as well as password

**MARKSHEET**

This section contains student’s stored data,student can find their marks by entering detail in ‘student detail’

Option, and after feeling their data he/she may automatically get their marks in ‘grades point option’.

**MENU**

This section includes menu’s for students details such as student profile,library system, fee report and Marksheet.

**SEARCH PAGE**

Here student can search their stored data entering roll no.

**STUDENT INFORMATION**

Here student can store their data in database form by entering data into ‘student information’ section.

**NON FUNCTIONAL REQUIREMENTS**

**PERFORMANCE REQUIREMENTS**

Performance should not be an issue because all of our server queries involve small pieces of data.Changing screens will require very little computation and thus will occur very quickly.Server updates should only take a few seconds as long as the phone can maintain a steady signal.

**RELIABILITY**

Must maintain data integrity. Computer crashes and misuse should not affect a user’s history

**AVAILABILITY**

The CMS Portal shall be available, up and running for 24\*7 throughout the year except due to the routine maintenance activities.

**SECURITY REQUIREMENTS**

Administrator and Users with valid credentials will be able to log in to Portal.Administrator will have access to the database structures at back- end.Administrator will have the rights for modifications as well as any Updation work for the datasets and website. Access to the various subsystems will be protected by a user log in screen that requires a user name and password.To be updated in future.

**1.2 SYSTEM REQUIREMENT SPECIFICATION FOR LIBRARY INFORMATION SYSTEM**

1. **INTRODUCTION**

**PURPOSE**

The main objective of this document is to illustrate the requirements of the project Library Management system. The document gives the detailed description of the both functional and non-functional requirements proposed by the client.The purpose of this project is to provide a friendly environment to maintain the details of books and library members.The main purpose of this project is to maintain easy circulation system using computers and to provide different reports. This project describes the hardware and software interface requirements using ER diagrams and UML diagrams.

**Document Conventions**

* Entire document should be justified.
* Convention for Main title
  + Font face: Times New Roman
  + Font style: Bold
  + Font Size: 14
* Convention for Sub title
  + Font face: Times New Roman
  + Font style:Bold
  + Font Size: 12
  + Convention for body
  + Font face:Times New Roman
  + Font Size: 12

**SCOPE OF DEVELOPMENT PROJECT**

Library Management System is basically updating the manual library system into an internetbased application so that the users can know the details of their accounts, availability of books and maximum limit for borrowing.

The project is specifically designed for the use of librarians and library users. The product will work as a complete user interface for library management process and library usage from ordinary users. Library Management System can be used by any existing or new library to manage its books and book borrowing, insertion and monitoring. It is especially useful for any educational institute where modifications in the content can be done easily according to requirements.

**DEFINITIONS, ACRONYMS AND ABBREVIATIONS**

JAVA -> platform independence SQL-> Structured query Language

ER-> Entity Relationship

UML -> Unified Modeling Language

IDE-> Integrated Development Environment SRS-> Software Requirement Specification ISBN -> International Standard Book Number

IEEE ->Institute of Electrical and Electronics Engineers

**OVERALL DESCRIPTIONS**

**PRODUCT PERSPECTIVE**

This is a broad level diagram of the project showing a basic overview. The users can be either staff or student.. This System will provide a search functionality to facilitate the search of resources. This search will be based on various categories viz. book name or the ISBN. Further the library staff personnel can add/update the resources and the resource users from the system.The users of the system can request issue/renew/return of books for which they would have to follow certain criteria.

**Product Function**

The Online Library System provides online real time information about the books available in the Library and the user information. The main purpose of this project is to reduce the manual work. This software is capable of managing Book Issues, Returns, Calculating/Managing Fine, Generating various Reports for Record-Keeping according to end user requirements. The Librarian will act as the administrator to control members and manage books. The member’s status of issue/return is maintained in the library database. The member’s details can be fetched by the librarian from the database as and when required. The valid members are also allowed to view their account information.

**User Classes and Characteristics**

The system provides different types of services based on the type of users [Member/Librarian]. The Librarian will be acting as the controller and he will have all the privileges of an administrator. The member can be either a student or staff of the university who will be accessing the Library online.

* The features that are available to the Librarian are:- ➢ A librarian can issue a book to the member.
  + Can view the different categories of books available in the Library
  + Can view the List of books available in each category
  + Can take the book returned from students
  + Add books and their information to the database
  + Edit the information of existing books
  + Can check the report of the existing books
  + Can check the report of the issued books
  + Can access all the accounts of the students
* The features that are available to the Members are:-
  + Can view the different categories of books available in the Library • Can view the List of books available in each category
  + Can own an account in the library.
  + Can view the books issued to him
  + Can put a request for a new book
  + Can view the history of books issued to him previously
  + Can search for a particular book

**REQUIREMENT**

Software Configuration:-

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database. Operating System: Windows NT, windows 98, Windows XP Language: Java Runtime Environment, Net beans 7.0.1 (front end) Database: MS SQL Server (back end)

Hardware Configuration:- Processor: Pentium(R)Dual-core CPU Hard Disk: 40GB

RAM: 256 MB or more

**DATA REQUIREMENT**

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting books and putting into account. Now the output will be visible when the user requests the server to get details of their account in the form of time, date and which books are currently in the account.

**Login Interface:**

In case the user is not yet registered, he can enter the details and register to create his account. Once his account is created he can ‘Login’ which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message appears.

**Search:-**

The member or librarian can enter the type of book he is looking for and the title he is interested in,then he can search for the required book by entering the book name.

**Categories View:-**

Categories view shows the categories of books available and provides ability to the librarian to add/edit or delete category from the list.

**Librarian’s Control Panel:-**

This control panel will allow librarian to add/remove users; add, edit, or remove a resource. And manage lending options.

**OTHER NON-FUNCTIONAL REQUIREMENTS**

**PERFORMANCE REQUIREMENT**

The proposed system that we are going to develop will be used as the Chief performance system within the different campuses of the university which interacts with the university staff and students. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the university.

* + - The performance of the system should be fast and accurate
    - Library Management System shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify invalid username/password
    - The system should be able to handle large amount of data. Thus it should accommodate high number of books and users without any fault

**SECURITY REQUIREMENT**

* + - System will use secured database
    - Normal users can just read information but they cannot edit or modify anything except their personal and some other information.
    - System will have different types of users and every user has access constraints Proper user authentication should be provided
    - No one should be able to hack users’ password
    - There should be separate accounts for admin and members such that no member can access the database and only admin has the rights to update the database.

**REQUIREMENT ATTRIBUTES**

* + - There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the members or other users cannot do changes
    - The project should be open source
    - The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database
    - The user be able to easily download and install the system

**1.3 SYSTEM REQUIREMENT SPECIFICATION FOR BANKING MANAGEMENT SYSTEM**

**INTRODUCTION:**

**Purpose:**

This document gives detailed functional and non functional requirements for the bank management system. This product will support online banking transaction using ATM. The purpose of this document is that the requirements mentioned in it should be utilized by software developer to implement the system.

**Scope**

This Product will automate of banking transaction process.

**Overview**

The system provides easy solution to banks.

**Additional Information**

This system will work together with bank’s computer. It will not be operated independently. Various banks might be networked together.

**GENERAL DESCRIPTION:**

Using Automatic Teller Machine (ATM) customer can withdraw the desired amount of money. ATM is basically computerized telecommunication device that helps the customer to perform banking transactions. The customer is identified by inserting a plastic ATM card with a magnetic stripe or a plastic smartcard with a chip. This magnetic media contains a unique card number and some security information. Security is provided by the customer entering a Personal Identification Number (PIN). When a customer wants to perform some transaction ,he/she will enter the ATM card in the machine and then enter the PIN. Only authentic customer will be allowed to perform this transaction.

**FUNCTIONAL REQUIREMENTS**

This section provides the requirement overview of the product. The project will require the PHP as a front end and at the back end the database MYSQL will be running. Various functional modules that can be implemented by the product will be –

1. Login
2. Validation
3. Get balance information
4. Withdrawal of money
5. Transfer Money
6. Report Generation

**DESCRIPTION**

**Login:**

Customer logins by entering the card and typing the PIN.

**Validation:**

When a customer enters the ATM card, its validity must be ensured. Then

customer is allowed to enter the valid PIN. The validation can be for following conditions –

**Validation for lost or stolen card**

When card is already reported as lost or stolen then the message “Lost/Stolen card!!!”.

**Validation for card’s expiry date**

If the card inserted by the customer has crossed the

expiry date then the system will prompt “Expired Card”.

**Validation for PIN**

After validating the card, the validity of PIN must be ensured. If he/she fails to enter valid code for three times then the card will not be returned to him. That means the account can be locked.

The counter for number of logins must be maintained

**Withdrawal of Money:**

A customer is allowed to enter the amount which he/she wishes to withdraw. If the entered amount is less than the available balance and if after withdraw if the minimum required balance is maintained then allow the transaction.

**Transfer of Money:**

The customer can deposit or transfer the desired amount of money.

**Report:**

The bank statement showing credit and debit information of corresponding account must be printed by the machine.TECHNICAL ISSUES

This product will work on client-server architecture. It will require an internet server and which will be able to run PHP applications. The product should support some commonly used browsers such as Internet Explorer, Mozilla Firefox.

**INTERFACE REQUIREMENTS**

**Hardware Interface**

Various interfaces for the product could be

1. The Hardware interface that can read the ATM card.
2. Touch screen/Monitor
3. Keypad
4. Continuous battery backup
5. Printer which can produce the hard copy.
6. Interface that connects the device to bank’s computer.
7. An interface that can count currency notes.

**Software Interface**

1. Any windows operating system.
2. The PHP must be installed. For the database handling MYSQL must be installed. These products are open source3 products.
3. The final application must be packaged in a set up program, so that the products can be easily installed on ATM machines. This application must be networked to corresponding banks.

**Non Functional Requirements**

Security:

The ATM system must be in separate cabin.Its door must have ATM card swipe slot so that it can be opened to only authentic user.There must be emergency phone located outside the cabin. Some security watchman should be at ATM center.

Reliability:

The application should be highly reliable and it should generate all the updated information in correct order.

Availability:

Any information about the account should be quickly available from any computer to the authorized user. The previously visited customer’s data must not be cleared.

Maintainability:

The application should be maintainable in such a manner that if any new requirement occurs then it should be easily incorporated in an individual module.

Portability:

The application should be portable on any windows based system.

**1.4 SYSTEM REQUIREMENT SPECIFICATION FOR RAILWAY TICKET RESERVATION SYSTEM**

**Introduction:**

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS purpose ,scope, definitions, acronyms, abbreviations, references and overview of SRS.A **Software Requirements Specification** (**SRS**) - a requirements specification for a software system - is a complete description of the behaviour of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. Use cases are also known as functional requirements. In addition to use cases, the SRS also contains non-functional (or supplementary) requirements. Non-functional requirements are requirements which impose constraints on the design or implementation (such as performance engineering requirements, quality standards, or design constraints). The aim of this document is to gather and analyse and give an in-depth insight of the complete Marvel Electronics and Home Entertainment software system by defining the problem statement in detail. This is a documentation of the project **Railways Reservation System** done sincerely and satisfactorily by my group members. A Software has to be developed for automating the manual Railway Reservation System.

**Objective:**

The purpose of this source is to describe the railway reservation system which provides the train timing details, reservation, billing and cancellation on various types of reservation namely, • Confirm Reservation for confirm Seat. • Reservation against Cancellation. • Waiting list Reservation. • Online Reservation. • Tatkal Reservation. The origin of most software systems is in the need of a client, who either wants to automate the existing manual system or desires a new software system. The software system is itself created by the developer. Finally, the end user will use the completed system. Thus, there are three major parties interested in a new system: the client, the user, and the developer. Somehow the requirements for the system that will satisfy the needs of the clients and the concerns of the users have to be communicated to the developer. The problem is that the client doesn’t usually design the software or the software development process and the developer does not understand the client’s problem and the application area. This causes a communication gap between the parties involved in the development of the project. The basic purpose of Software Requirement Specification (SRS) is to bridge this communication gap. SRS is the medium through which the client’s and the user’s needs are accurately specified; indeed SRS forms the basis of software development. Another important purpose of developing an SRS is helping the clients understanding their own needs. An SRS establishes the basis for agreement between the client and the supplier on what the software product will do. An SRS provides a reference for validation of the final product.A high quality SRS is a prerequisite to high quality software and it also reduces the development cost. A few factors that direct us to develop a new system are given below -:

1. Faster System

2. Accuracy

3. Reliability

4. Informative

5. Reservations and cancellations from anywhere to any place.

**4.1 SYSTEM REQUIREMENT SPECIFICATION**

**Function Requirements**

performance requirements:

• User Satisfaction: - The system is such that it stands up to the user expectations.

• Response Time: -The response of all the operation is good. This has been made possible by careful programming.

• Error Handling: - Response to user errors and undesired situations has been taken care of to ensure that the system operates without halting.

• Safety and Robustness: - The system is able to avoid or tackle disastrous action. In other words, it should be foul proof. The system safeguards against undesired events, without human intervention.

• Portable: - The software should not be architecture specific. It should be easily transferable to other platforms if needed.

• User friendliness: - The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties.

Design constrian:

There are a number of factors in the client’s environment that may restrict the choices of a designer. Such factors include standards that must be followed, resource limits, operating environment, reliability and security requirements and policies that may have an impact on the design of the system. An SRS (Software Requirements Analysis and Specification) should identify and specify all such constraints.

Ø Standard Compliance: - This specifies the requirements for the standards the system must follow. The standards may include the report format and accounting properties.

Ø Hardware Limitations :- The software may have to operate on some existing or predetermined hardware, thus imposing restrictions on the design. Hardware limitations can include the types of machines to be used, operating system available on the system, languages supported and limits on primary and secondary storage.

Ø Reliability and Fault Tolerance: - Fault tolerance requirements can place a major constraint on how the system is to be designed. Fault tolerance requirements often make the system more complex and expensive. Requirements about system behavior in the face of certain kinds of faults are specified. Recovery requirements are often an integral part here, detailing what the system should do I some failure occurs to ensure certain properties. Reliability requirements are very important for critical applications.

Ø Security: - Security requirements are particularly significant in defence systems and database systems. They place restrictions on the use of certain commands, control access to data, provide different kinds of access requirements for different people, require the use of passwords and cryptography techniques and maintain a log of activities in the system.

Hardware requirements:

For the hardware requirements the SRS specifies the logical characteristics of each interface b/w the software product and the hardware components. It specifies the hardware requirements like memory restrictions, cache size, the processor, RAM size etc... those are required for the software to run.

Minimum Hardware Requirements:

* Processor Pentium III
* Hard disk drive 40 GB
* RAM 128 MB
* Cache 512 kb
* Preferred Hardware Requirements:
* Processor Pentium IV
* Hard disk drive 80 GB
* RAM 256 MB
* Cache 512 kb

Software requirements:

• Any window based operating system with DOS support are primary requirements for software development. Windows XP, FrontPage and dumps are required. The systems must be connected via LAN and connection to internet is mandatory.

Other requirements:

Software should satisfy following requirements as well:-

• SECURITY

• PORTABILITY

• CORRECTNESS

• EFFICIENCY

• FLEXIBILTY

• TESTABILTY

• REUSABILTY

**Non-Function Requirements**

Security:

• The system use SSL (secured socket layer) in all transactions that include any confidential customer information. The system must automatically log out all customers after a period of inactivity. The system should not leave any cookies on the customer’s computer containing the user’s password. The system’s back-end servers shall only be accessible to authenticated management.

Reliability:

• The reliability of the overall project depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Also the system will be functioning inside a container. Thus the overall stability of the system depends on the stability of container and its underlying operating system.

Availability:

• The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. A customer friendly system which is in access of people around the world should work 24 hours. In case of a of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the Organizer. Then the service will be restarted. It means 24 x 7 availability.

Maintainability:

• A commercial database is used for maintaining the database and the application server takes care of the site. In case of a failure, a re-initialization of the project will be done. Also the software design is being done with modularity in mind so that maintainability can be done efficiently.

Supportability:

• The code and supporting modules of the system will be well documented and easy to understand. Online User Documentation and Help System Requirements.

**Ex.NO:2 DATA FLOW DIAGRAM**

**AIM:**

Develop DFD model (level-0, level-1 AND LEVEL-2 DFD and Data dictionary) of the Project.

**DESCRIPTION:**

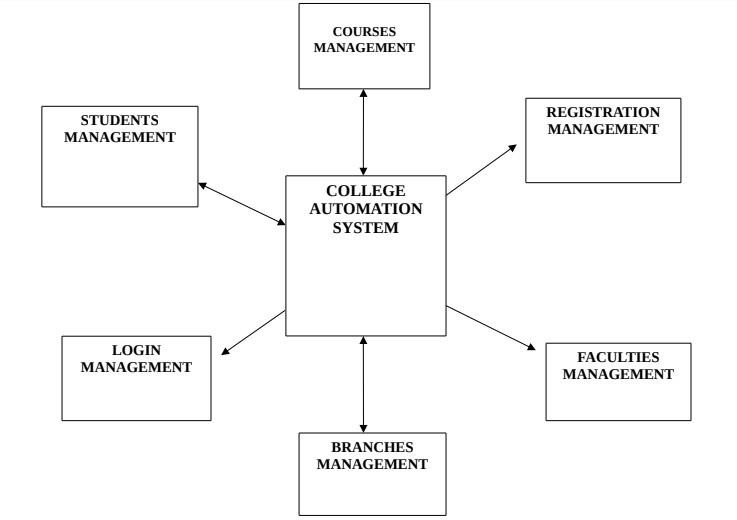
Data drive business activities and can trigger events or be processed to provide information about the activity. Data flow analysis, as the name suggests, follows the flow of data through business processes and determines how organisation objectives are accomplished. In the course of handling transactions and completing tasks, data are input, processed, stored, retrieved, used, changed and output. Data flow analysis studies the use of data in each activity and documents the findings in data flow diagrams, graphically showing the relation between processes and data.

**Data Flow Diagram (DFD):**

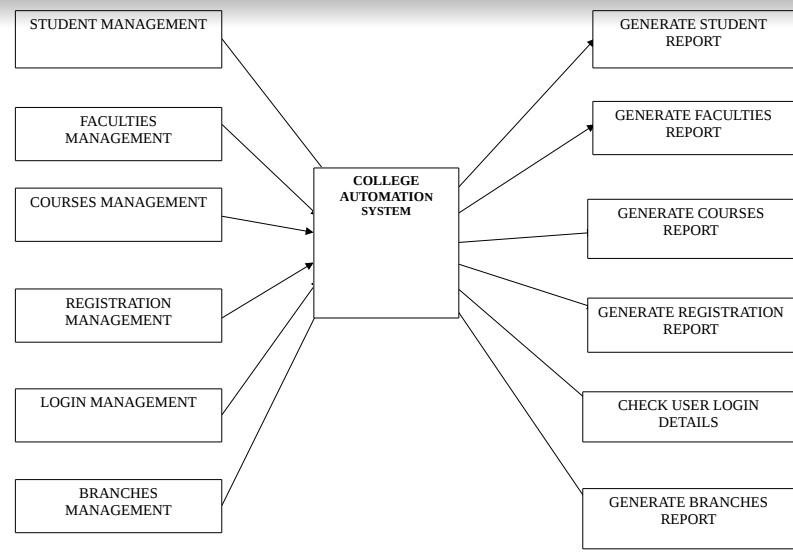
The DFD (also known as a bubble chart) is a hierarchical graphical model of a system that shows the different processing activities or functions that the system performs and the data interchange among these functions. Each function is considered as a processing station (or process) that consumes some input data and produces some output data. The system is represented in terms of the input data to the system, various processing carried out on these data, and the output data generated by the system. A DFD model uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among these functions.

**Ex.NO: 2.1 DFD OF COLLEGE AUTOMATION SYSYTEM**

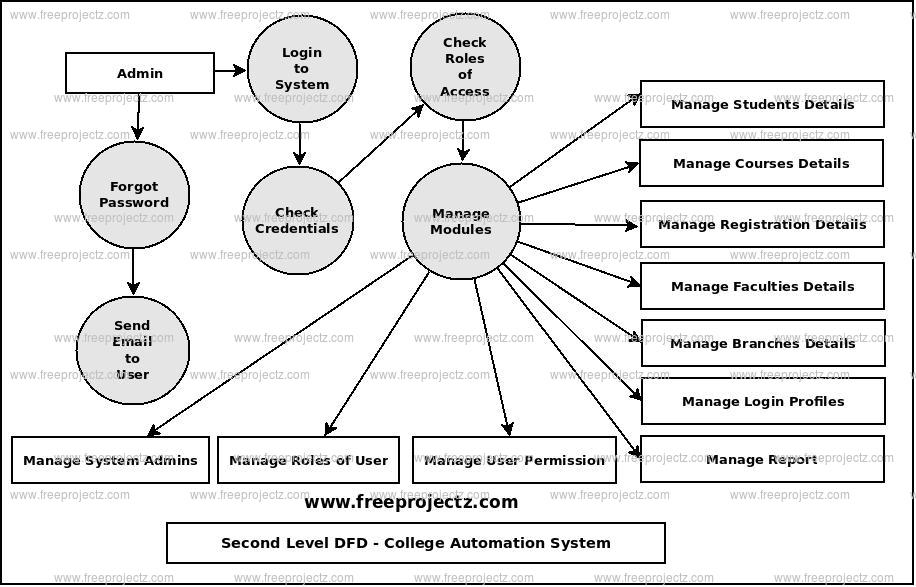
**ZERO LEVEL DFD**



**FIRST LEVEL DFD**

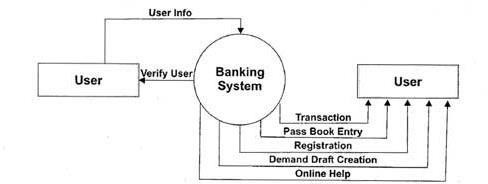


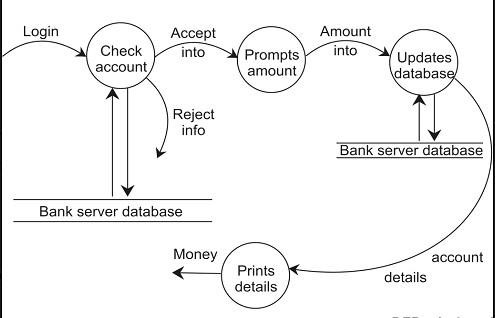
**SECOND LEVEL DFD**



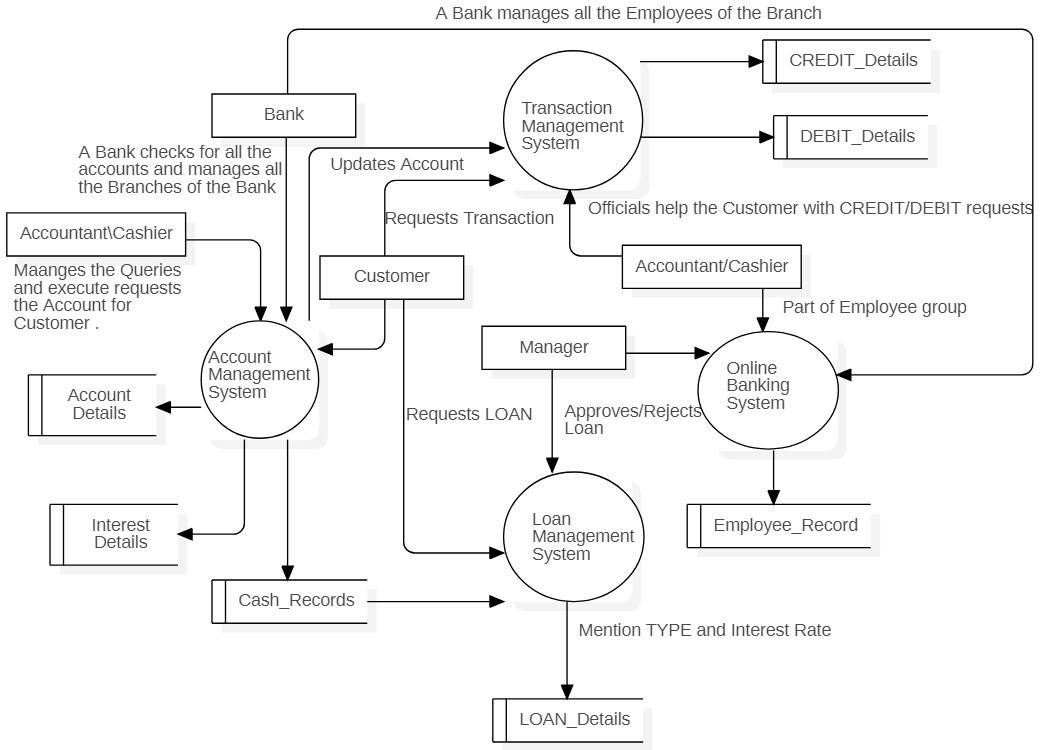
**Ex.NO: 2.2 DFD OF BANKING MANAGEMENT SYSYTEM**

**ZERO LEVEL DFD**



**FIRST LEVEL DFD**

**SECOND LEVEL DFD**



**Ex.NO: 2. DFD OF LIBRARY MANAGEMENT SYSYTEM**

**ZERO LEVEL DFD**

A diagram of a library management system

Description automatically generated

**FIRST LEVEL DFD**

A diagram of a book delivery

Description automatically generated

**SECOND LEVEL DFD**

A diagram of a book

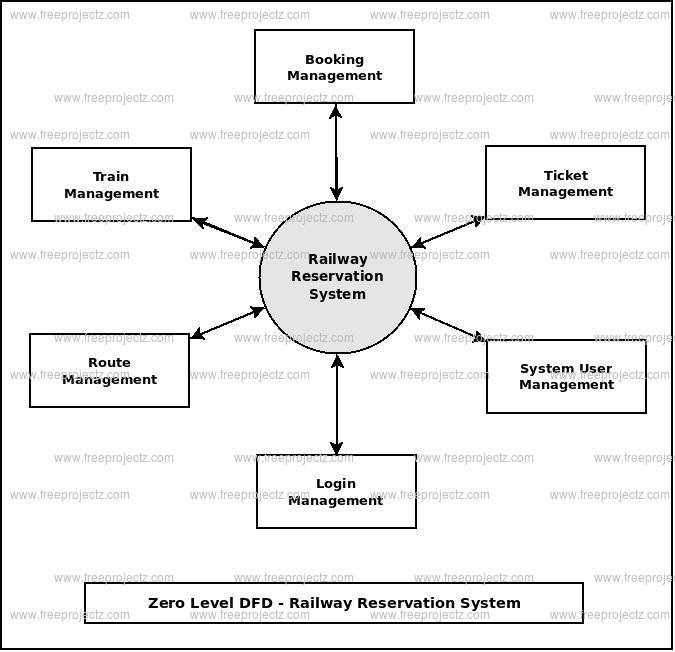
Description automatically generated

Schedules

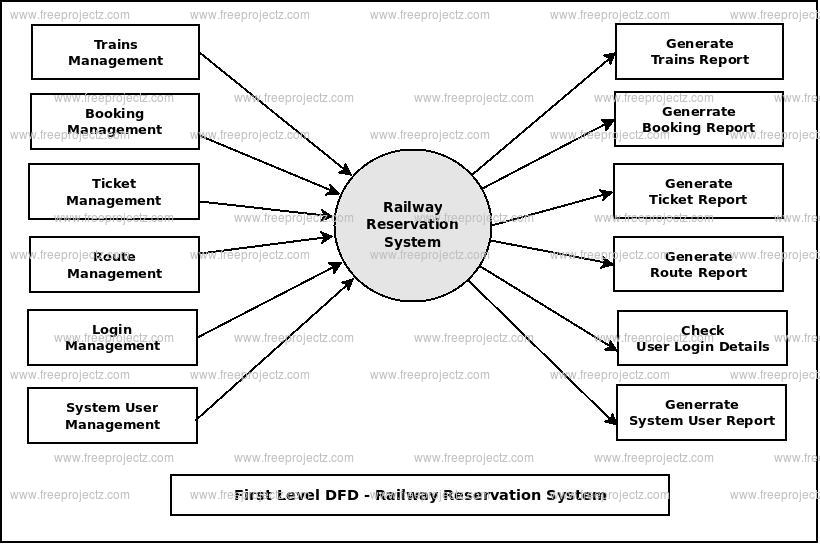
nagement

**Ex.NO: 2.4 DFD OF RAILWAY TICKET RESERVATION SYSYTEM**

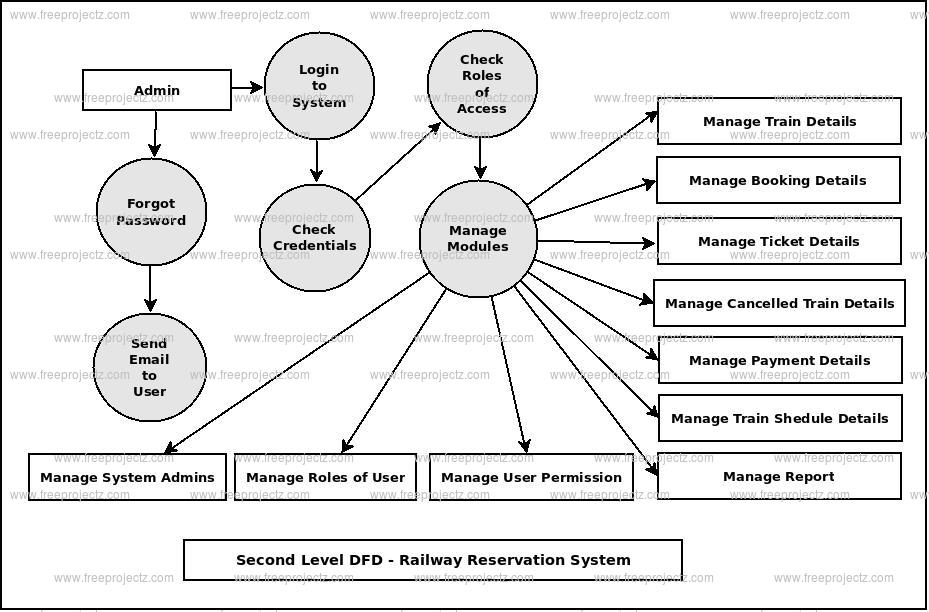
**ZERO LEVEL DFD**



**FIRST LEVEL DFD**



**SECOND LEVEL DFD**



**3.DEVELOP CLASS DIAGRAM:**

**3.1 DCD OF COLLEGE AUTOMATION SYSTEM**

To show diagrammatically the objects required and the relationships between them while developing a software product.

Software Required :- StarUML 6.0.1

Procedure:

Step 1: Launch StarUML 🡪Open StarUML on your computer.

Step 2: Create a New Project🡪Click on "File" in the top-left corner.

Select "New Project" to create a new project.

Step 3: Add a Class Diagram🡪In the project explorer, right-click on the project name. Select "Add Diagram" and then choose "Class Diagram."

Step 4: Add Classes. In the toolbar, find the "Class" icon (a rectangle with folded corner) and click on it. Click on the class diagram canvas to add a class. Double-click on the class to rename it.

Step 5: Add Attributes and Methods to Classes🡪Inside the class box, click on the "+" icon to add attributes or methods. Double-click on the attribute or method to rename it.

Step 6: Define Relationships🡪Use the "Association" tool (two boxes connected with a line) to draw associations between classes. You can also use other tools like aggregation and composition based on your requirements.

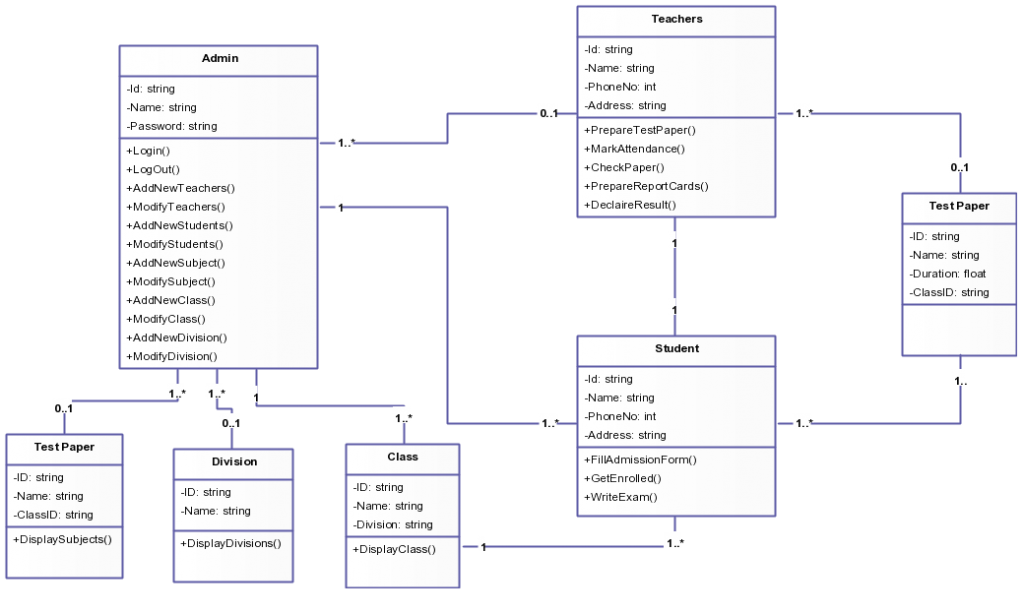
Step 7: Add Multiplicity and Role Names🡪After creating associations, you can specify multiplicity and role names. Click on the association line.

Use the toolbar to set multiplicity (e.g., 0..1, 1, \*, etc.) and role names.

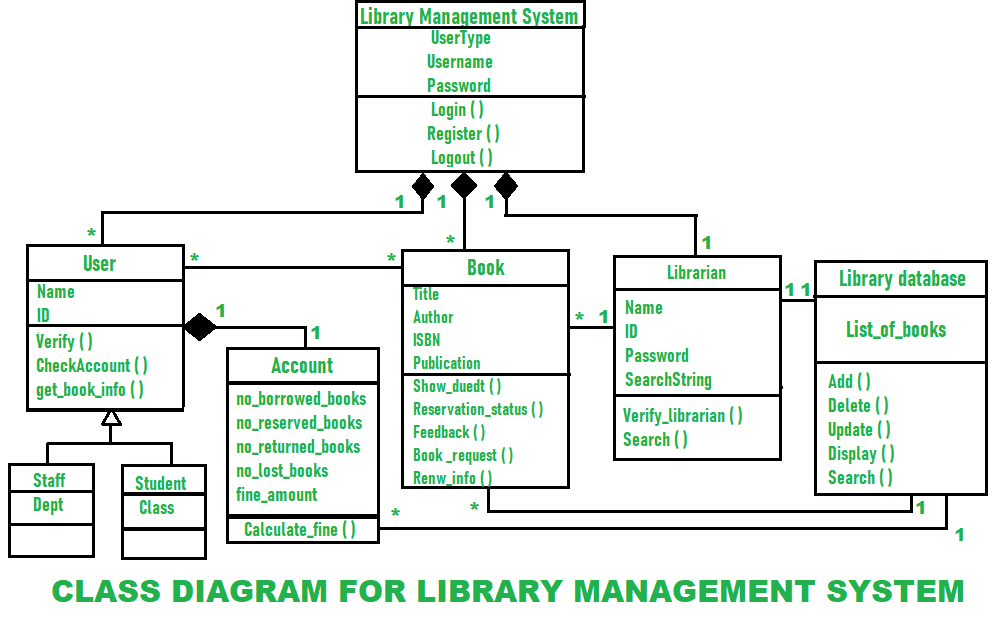
Step 8: Generalization 🡪Use the "Generalization" tool (a triangle arrow) to create inheritance relationships. Connect the child class to the parent class.

Step 9: Save Your Project🡪Click on "File" and select "Save" to save your project.

**3.1 DEVELOPE CLASS DIAGRAM FOR COLLEGE AUTOMATION SYSTEM:**



**3.2 DEVELOPE CLASS DIAGRAM FOR LIBRARY INFORMATION SYSTEM:**

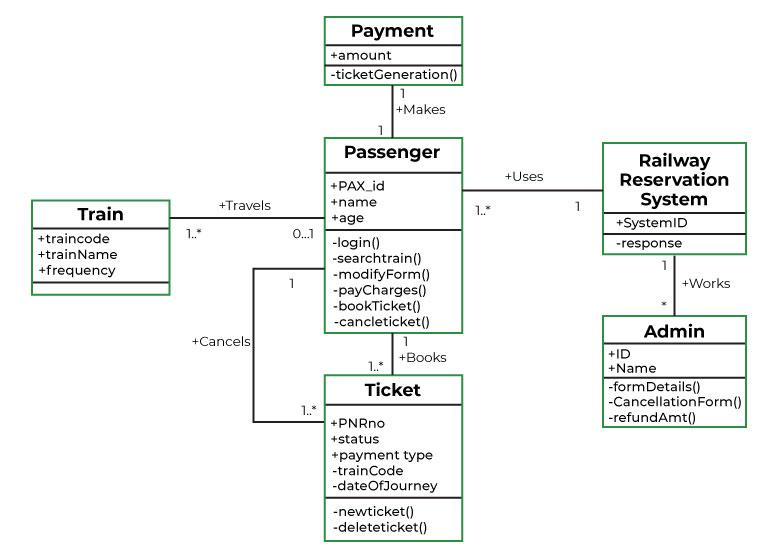


**3.3 DEVELOPE CLASS DIAGRAM FOR BANKING MANAGEMENT SYSTEM:**

A diagram of a computer

Description automatically generated

**3.4 DEVELOPE CLASS DIAGRAM FOR RAILWAY TICKET RESERVATION SYSTEM:**



**4.Develop UML Usecase model for a problem To understand users view of project using usecase diagram**

Software Required : StarUML6.0.1

Procedure:-

Step1:Open StarUML🡪Launch the StarUML application on your computer.

Step2: Create a New Project🡪Click on "File" in the menu🡪Select "New Project."

Step3: Add a Use Case Diagram🡪In the toolbar, click on the "Add Diagram" button (usually represented by a "+" symbol).Choose "Use Case Diagram" from the list.

Step4: Add Actors🡪Click on the "Actor" icon in the toolbar. Click on the diagram to place the actor. Repeat this process for each actor in your system.

Step 5: Add Use Cases🡪lick on the "Use Case" icon in the toolbar. Click on the diagram to place a use case. Repeat this process for each use case in your system.

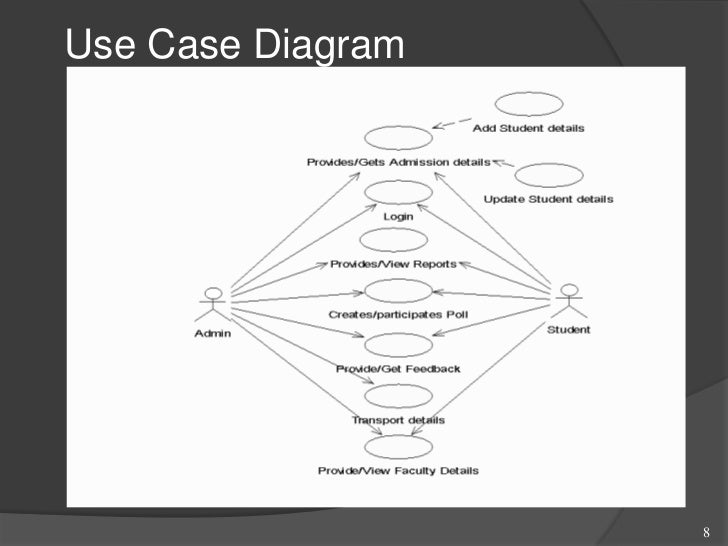
Step 6: Connect Actors and Use Cases Click on the "Association" icon in the toolbar. Click on an actor .Drag the line to the corresponding use case. Release the mouse button.

Step 7: Add Relationships🡪Click on the "Dependency" icon in the toolbar. Click on an actor or use case. Drag the line to another actor or use case. Release the mouse button.

Step 8: Add Multiplicity🡪Double-click on the association line. In the properties panel, you can set multiplicity for both ends (e.g., 1, \*, 0..1).

Step 9: Save Your Project🡪Click on "File" and choose "Save" to save your project.

**4.1 Use Case diagram FOR COLLEGE AUTOMATION SYSTEM**

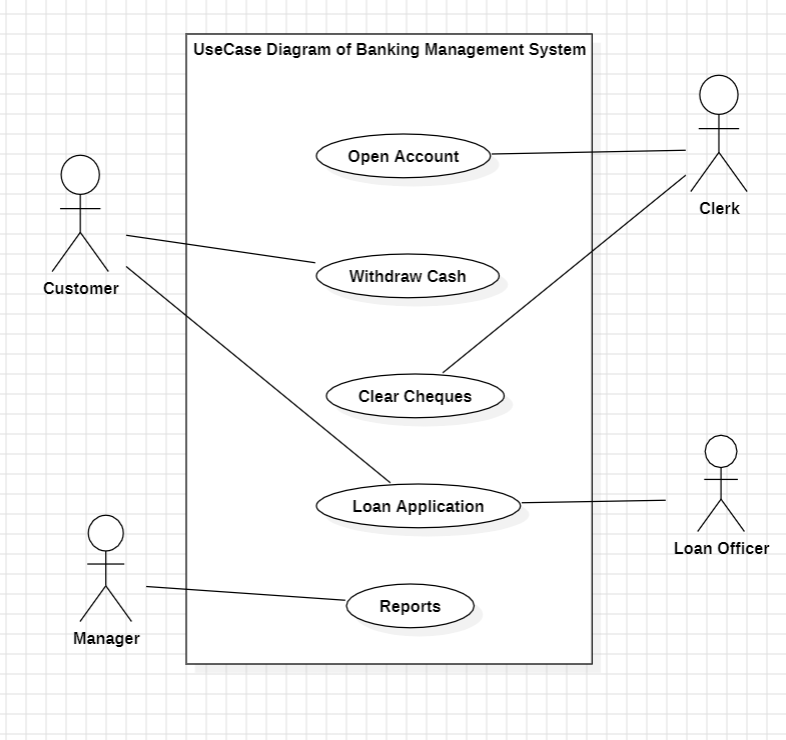


**4.2 Use Case diagram FOR LIBRARY INFORMATION SYSTEM**

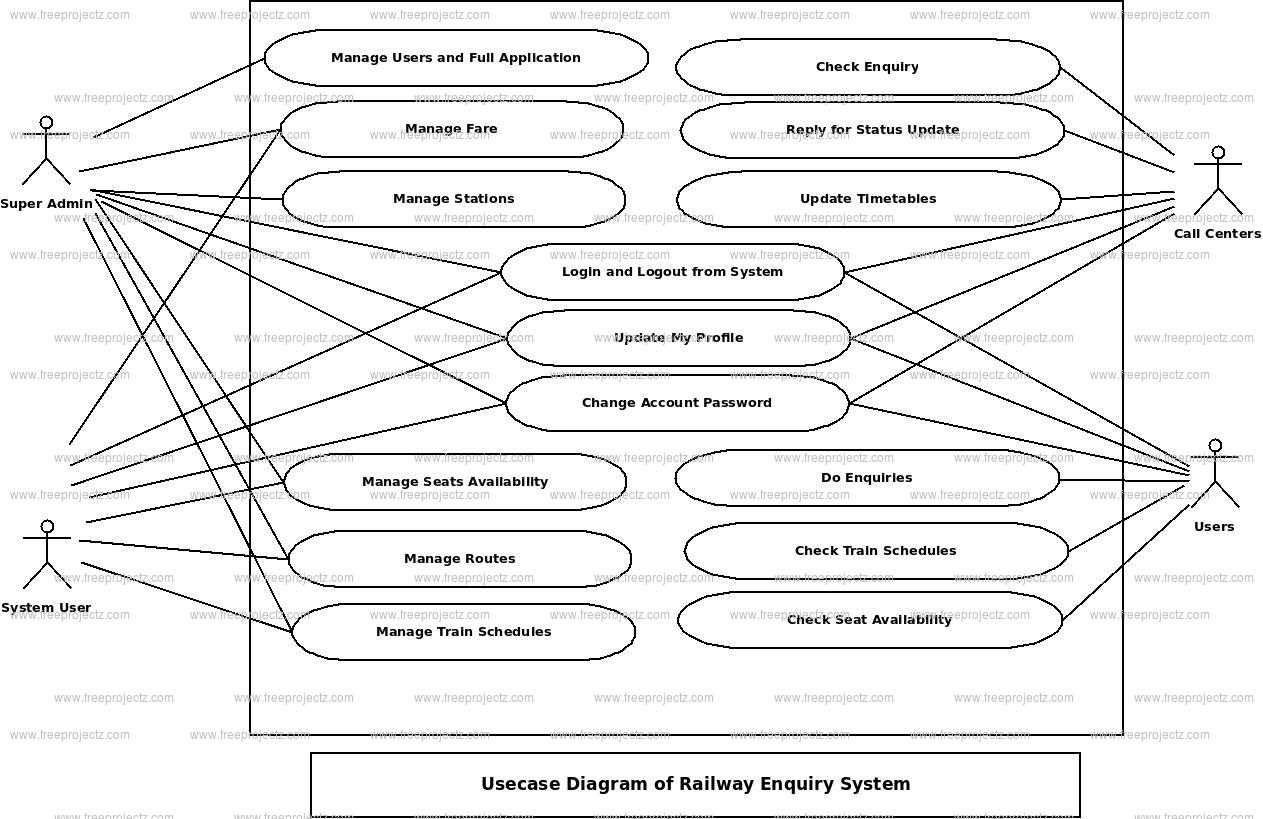
A diagram of a diagram

Description automatically generated

**4.3 Use Case diagram FOR BANKING MANAGEMENT SYSTEM**



**4.4 Use Case diagram FOR BANKING MANAGEMENT SYSTEM**



**5. Develope sequence diagram Objective :**

To understand the interactions between objects that are represented as lifelines in a sequential order of a project using Sequence Diagram.

**Procedure:**

Step 1:Open StarUML🡪Launch the StarUML application on your computer.

Step 2: Create a New Project🡪Click on "File" in the menu.Select "New Project."

Step 3: Add a Use Case Diagram🡪In the toolbar, click on the "Add Diagram" button. Choose "Use Case Diagram" from the list.

Step 4: Add Actors🡪Click on the "Actor" icon in the toolbar.Click on the diagram to place the actor.Repeat this process for each actor in your system.

Step 5: Add Use Cases🡪Click on the "Use Case" icon in the toolbar.Click on the diagram to place a use case.Repeat this process for each use case in your system.

Step 6: Connect Actors and Use Cases🡪Click on the "Association" icon in the toolbar. Click on an actor. Drag the line to the corresponding use case.Release the mouse button.

Step 7: Add Relationships🡪Click on the "Dependency" icon in the toolbar.

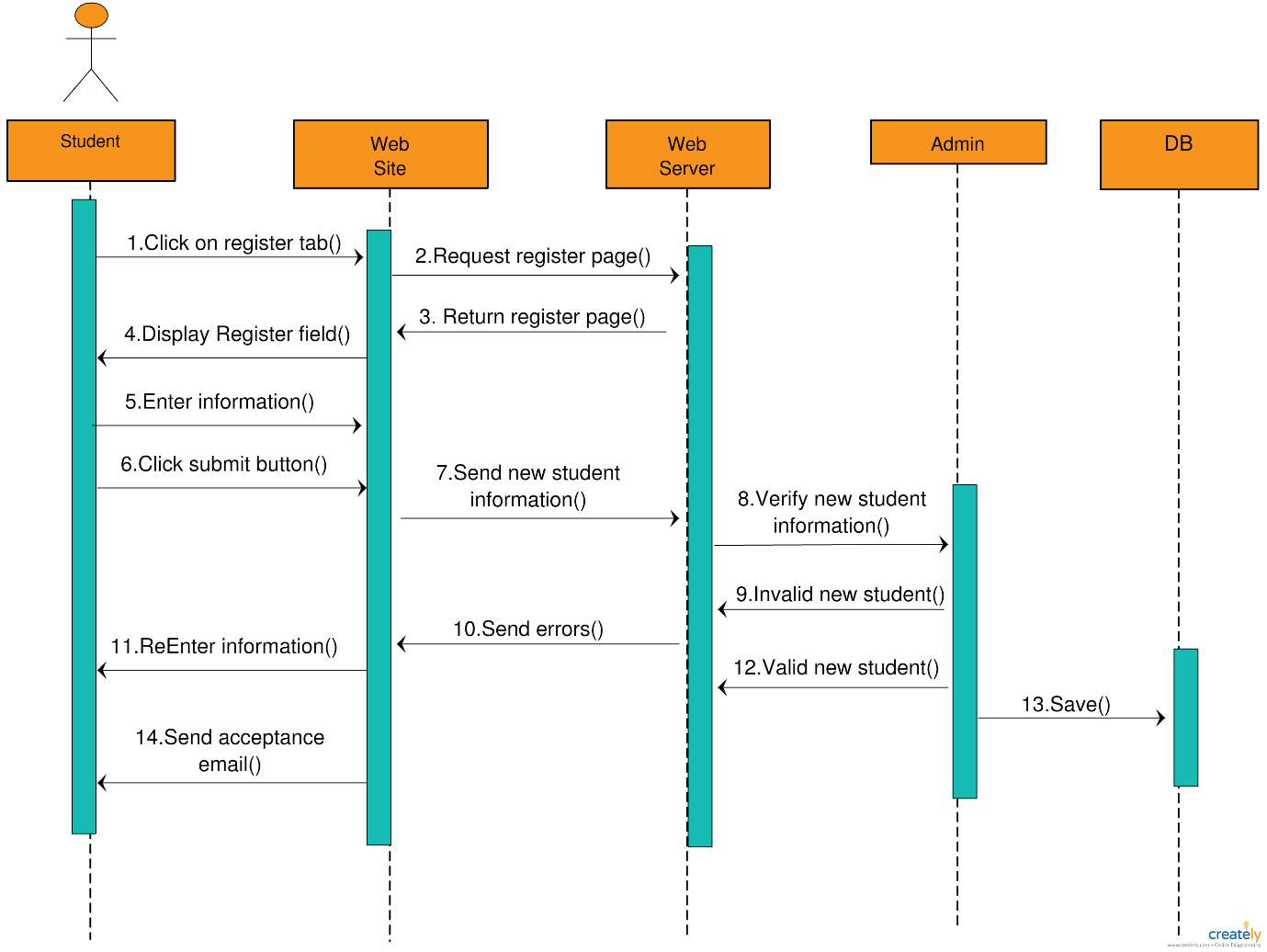
Click on an actor or use case.Drag the line to another actor or use case.Release the mouse button.

Step 8: Add Multiplicity🡪Double-click on the association line.In the properties panel, you can set multiplicity for both ends (e.g., 1, \*, 0..1).

Step 9: Save Your Project🡪Click on "File" and choose "Save" to save your project.

**5.1SEQUENCE DIAGRAM FOR COLLEGE AUTOMATION**

**SYSTEM**



**5.2 SEQUENCE DIAGRAM FOR LIBRARY INFORMATION SYSTEM**

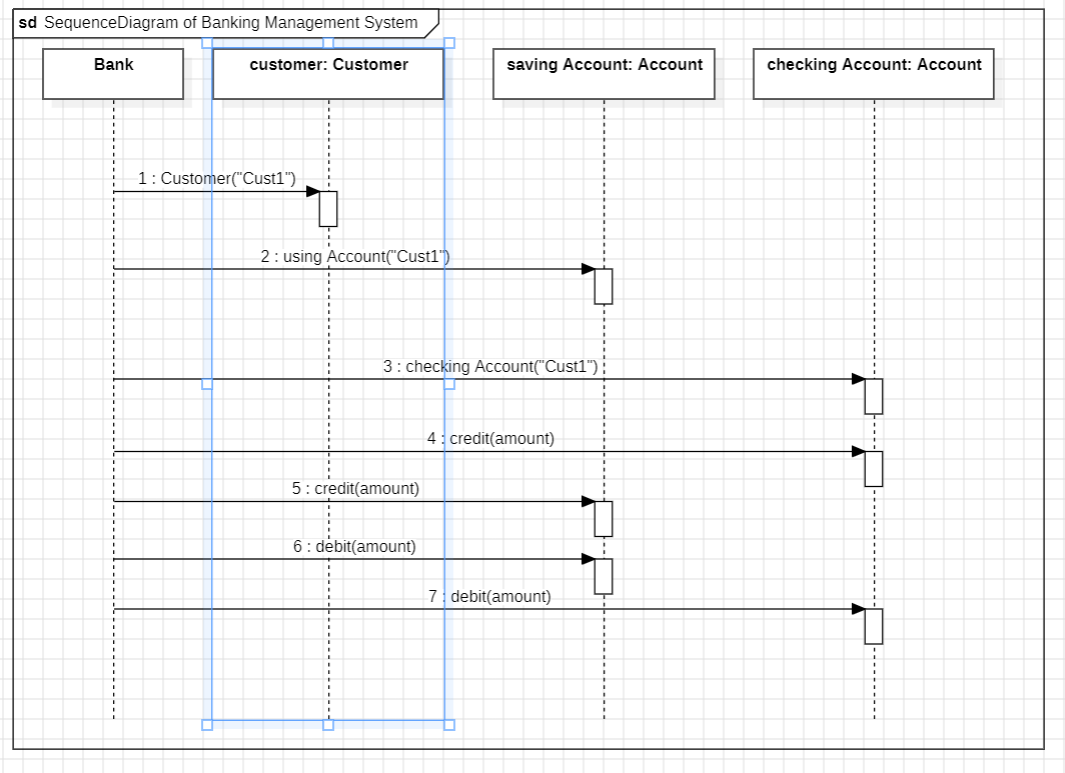
**A diagram of a diagram

Description automatically generated**

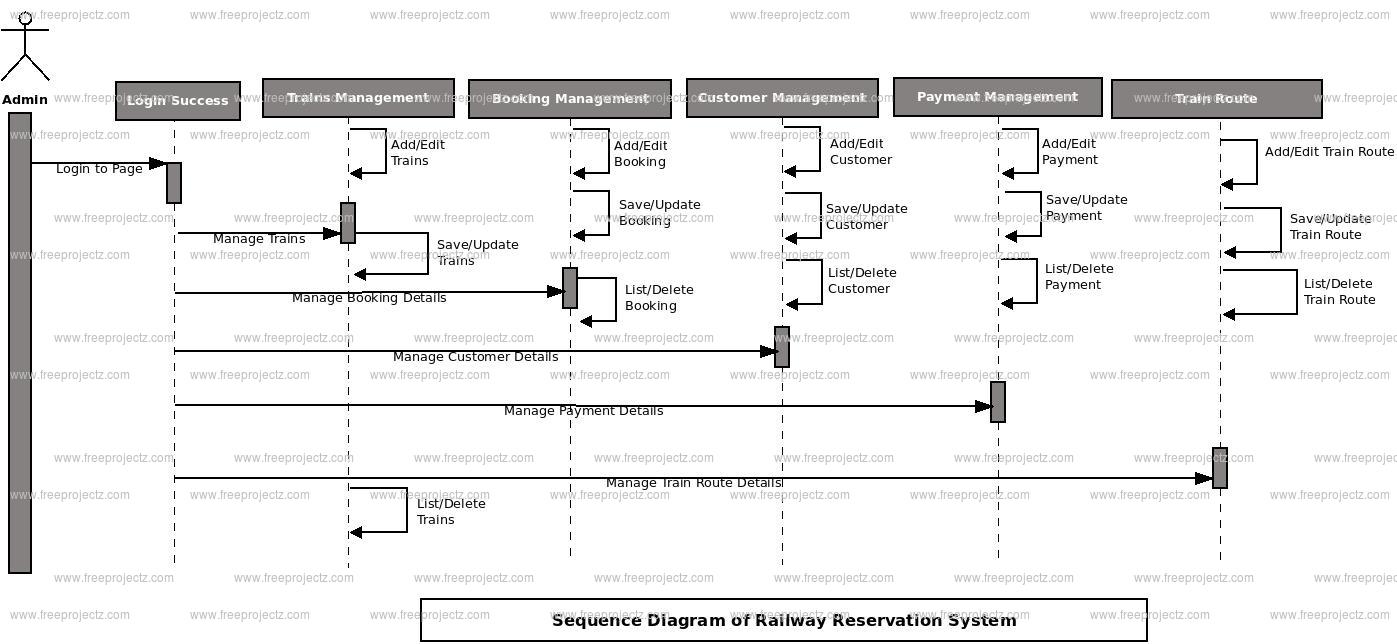
**A diagram of a process

Description automatically generated**

**5.3 SEQUENCE DIAGRAM FOR BANKING MANAGEMENT SYSTEM**

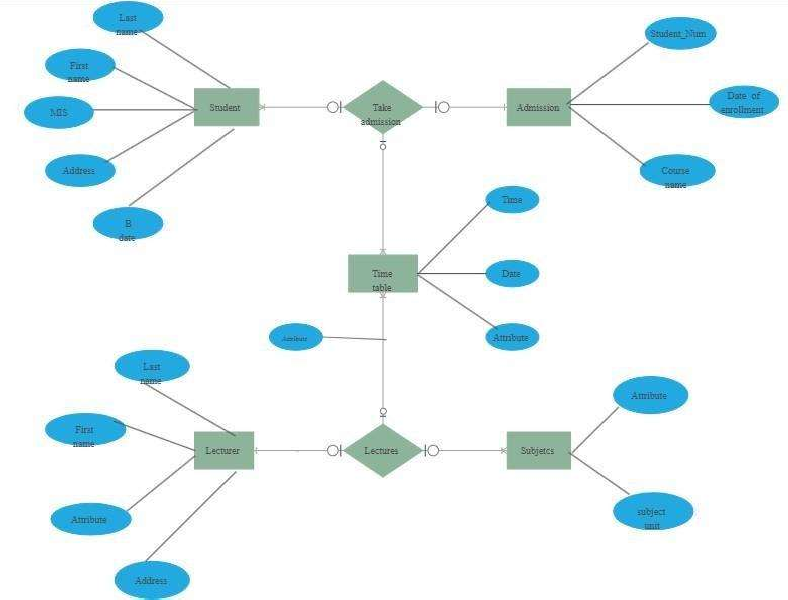


**5.4 SEQUENCE DIAGRAM FOR RAILWAY RESERVATION SYSTEM**



**6. ER DIAGRAM:**

**6.1 ER DIAGRAM OF COLLEGE AUTOMATION SYSTEM**

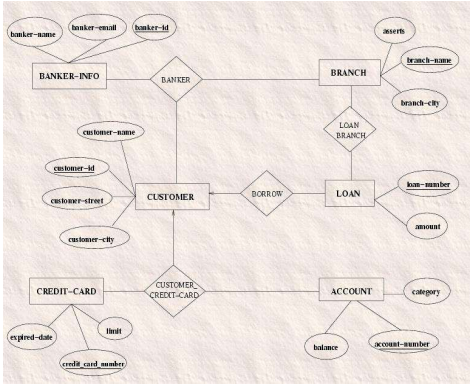


**6.2 ER DIAGRAM OF LIBRARY INFORMATION SYSTEM**

A diagram of a library management system

Description automatically generated

**6.3 ER DIAGRAM FOR BANKING MANAGEMENT SYSTEM**



**6.4 ER DIAGRAM FOR RAILWAY RESERVATION SYSTEM**

