Software Requirements Specification

for

Student Information System

Version 1.0 approved

Prepared by Shubham Bankar

192050024

VJTI, Matunga

29/11/2019

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The main objective of this document is to illustrate the requirements of the project Student Information System. The document gives the detailed description of the both functional and non-functional requirements for this project. SIS is mainly designed for the use of students. The purpose of this project is to provide a user friendly environment for the students and view the information regarding marks, attendance details, placement details. This project also maintain easy circulation system using computers to provide efficient access to the end user.

1.2 Document Conventions

Throughout the document, the font used is Droid Sans and the font sizes are such that the topic is easily understood and followed as in the case of any conventional document practice. IEEE Standard Software Requirement Specification is used.

Title:
Font Style: Droid Sans
Size: 32
Heading:
Font Style: Droid Sans
Size: 22
Subheading:
Font Style: Droid Sans
Size: 16
Content:
Font Style: Droid Sans
Size: 12

1.3 Intended Audience and Reading Suggestions

This document is intended for readers that include:

- Project Managers
- Developers
- Testers
- Staff
- Researchers working in this domain.

This SRS is categorized into five sections, viz. Introduction, Overall Description, System Features, External Interface Requirements, Other Non-functional Requirements and Other Requirements. The document will give you all the details of the project that has been laid down after very careful analysis.

1.4 Product Scope

This software system will be a java applicaton which can be accessed from anywhere within the college/university campus for any university/college wishing to manage their student details online. More specifically to design and develop a simple and intuitive system which shall cater the need of a suitable interface for students as well as for faculties, to get the up-to-date notifications from TPO regarding placements. It shall also carter the details about the student's academics. The portal increases the efficiency of college record management because previously the college is dependent on paper records only. The time required is less when compared with the paper records. For example generating the eligibility list for students based on the companies requirements will consume more time because the list differs from company to company, by using SIMS it can be generated easily and reduces the man power to do this. Also all users are authenticated to avail the services of this portal.

Benefits:

- User friendly automated tool.
- Paper work minimized.
- Cost effective.
- More security through authentication.
- Data can be retrieved at any moment and from anywhere within the campus.

About Software:

- Easy insertion, updating and retrieval of data using GUI is achieved through this system.
- Data is being stored in multiple tables, helping to achieve redundancy which may be useful if the system crashes.
- Can be executed on any system within the university/college campus.
- Input validation can be performed.

1.5 References

Books

- Software Requirements and Specifications: A Lexicon of Practice, Principles and Prejudices (ACM Press) by Michael Jackson
- Software Engineering: A Practitioner's Approach Fifth Edition By Roger S. Pressman

Papers

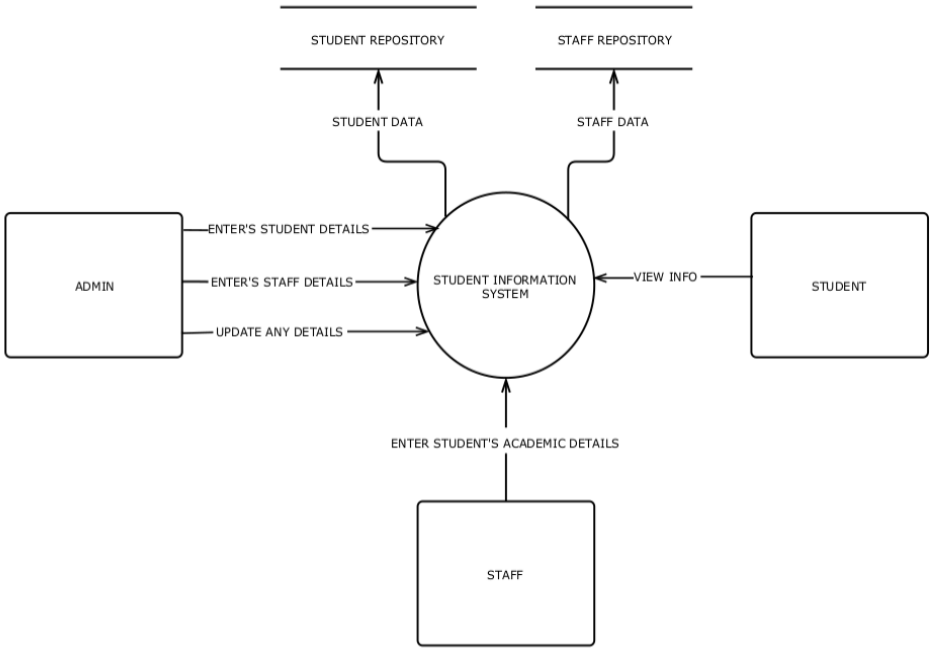
- Standard IEEE 830: Recommended practice for SRS is referred.

2. Overall Description

2.1 Product Perspective

SIS shall be developed using the client-server architecture. SIS is an upgrade to traditional SIS as it's an Internet based application that can be accessed throughout the Net and can be accessed by anyone who has a net connection. It is an automatic system, where we will automate the viewing of all the info about student. After inserting the data to database, student can simply login into the portal using his own id and password and can see the details regarding his marks, attendance and placements. The cool thing about this is that the student need not go to professors and ask each and every time about details.

DFD – Level 0 (Context Diagram)



2.2 Product Functions

The SIS will allow access only to authorized users with specific roles (System administrator, Staff and Student). Depending upon the user's role, he/she will be able to access only specific modules of the system.

A summary of major functions that the SIS will perform are :

- A login interface enabling authorized access to the system.
- Faculty will be able to add in the attendance details, and marks.
- Making a separate database for attendance and marks details.
- TPO will create a eligibility list of students based on their marks. This makes the work of TPO very easy as he doesn't have to search for marks of students by searching many papers.
- Student upon login will be able to view their personal and academic details.
- Finally an administrator who will be adding and updating details of the students. Also administrator can drop any student data.

2.3 User Classes and Characteristics

User of the system should be able to retrieve the information of his/her academic, personal and placement details from database easily. Also, some users act as an administrator who insert data into the database like marks and attendance details. The system will support three types of user privileges, Admin, Staff, Students.

Administrator:

The main role of administrator is to add or update the details of the students. This adding and updating entirely takes place on the Student database. Updation is entirely done using studentID to withdraw ambiguity. Administrator also stores in the Staff information which is helpful as it makes adding details like attendance and marks. Once new student or staff data is being added into database, admin provides with the username and password for the same. Finally, admin can also drop any student data, also taking aid of studentID to remove nay ambiguity.

Faculty:

The main role a faculty plays is that he/she is the one who collects all the personal and academic records of the student. Like he sends in the google sheet through which he collects the data of the students. Also he/she enters the marks and attendance details of the students. Faculty database contains all the details about the year, branch and respective subject he/she teaches. Thus the system automatically fetches all the information once the faculty logs in. All faculty needs to do is enter studentID and the respective count of attendance or marks

TPO:

The main role of TPO is to create an eligibility list. He checks the marks details of the students through mark database and shortlists the eligible students who can sit for the process of company placements. The eligible students are stored into another database which includes his name and studenetID and students have access to this database and can view if they are eligible for any company or not. Shortlisting is done automatically saving a lot of time.

Student:

The most important role is that of the student. He/She is the one who views the details, checks about his marks or attendance, if any queries still persists than informs the respective faculty, and also checks for placement details. He sends the info to the faculty upon request.

2.4 Operating Environment

The product will be operating in windows environment or Linux environment. The SIS is a JAVA based application and shall operate in all computers supporting Java. For database purpose we will take MS Access or MySQL. Anyone with latest Java (8 or higher) can access this application.
The hardware configuration include Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor etc.

2.5 Design and Implementation Constraints

The design constraint is that SIS is able to handle only a maximum of 2000 users, 50 faculty members, and 1 admin concurrently at a time. Also faculty cannot perform more than two operations at a time like adding marks and attendance simultaneously for the same student. Also student can run this application provided he uses his internet based login id and password; hence if the net is slow, he might not be able to access the system.

Users must use the latest operating system win7 and above as well as latest JAVA (8 or higher) to access the content of the application safely without any failure to gain access. SIS uses MySQL as its primary database and Java as its primary language and hence many external libraries like JDBC to connect mysql with java is required. Adding to it is the fact that t the system is implemented using JavaFX, and thus, JavaFX libraries

also needs to be added.

2.6 User Documentation

The manual to use the SIS will be available on the college or university website. Also the the application itself is easy to use since it is very user friendly. The steps for using SIS will be given in the manual itself.

2.7 Assumptions and Dependencies

The assumptions are:

- The code should be error free
- The system should be user-friendly so that it is easy to use for the users
- The information of all users, transactions, components collected must be stored in a database that is accessible by the administrator.
- The system should have more storage capacity and provide fast access to the database.
- Users may access from any computer within the university premises that has Internet browsing capabilities and an Internet connection, since student can access the application only if he log in using his internet id and password, since SIS is uploaded on server.
- Users must have their correct userID and passwords to enter into their online accounts and do actions.
- Staff should also use proper userID and password in order to enter details regarding students. Assumption is that while entering student details like marks and attendance, studentID must be correctly specified.
- Since JavaFX and MySQL is used, all necessary libraries must be present on the server in order to run the application without any type of error.

The dependencies are:-

- The specific hardware and software due to which the product will be run.
- On the basis of listing requirements and specification the project will be developed and run.
- The end users, staff and admin should have proper understanding of the system.
- The information of all the users must be stored in a database that is accessible by the administrator.
- Any update regarding the student is to be recorded to the database and the data entered should be correct.
- Since TPO, Faculty uses the student data to add details, studentID will be the dependent identity.

3. External Interface Requirements

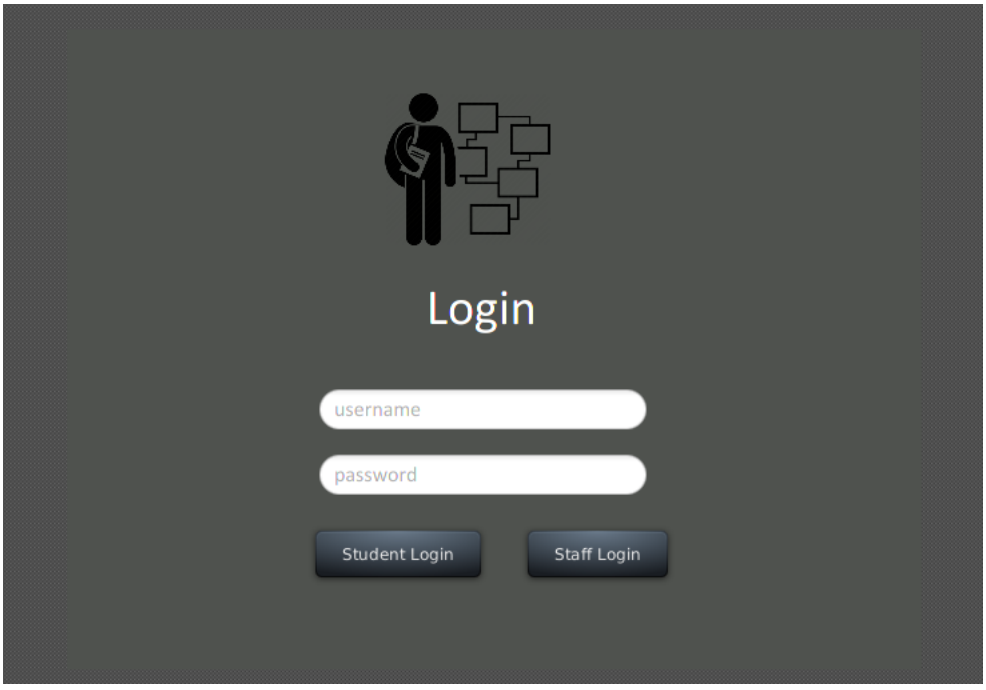
3.1 User Interfaces

The software provides good graphical interface for the user, staff and the administrator and hence they can operate on the system, performing the required tasks.

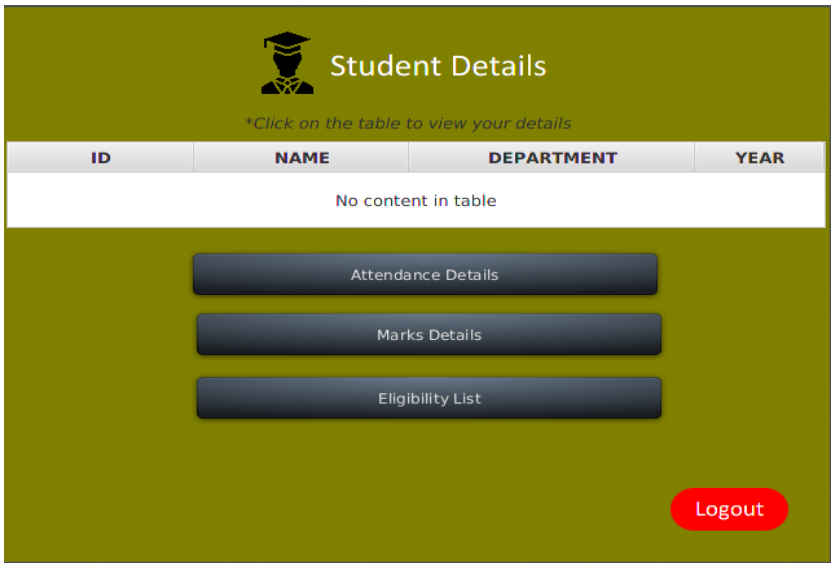
- User-interface must be simple, secure and user friendly.
- It must be simplified and self-explanatory.
- There must be robust error message system if user commits some error.
- It should visualize and explain the basic motivation behind this application.
- The system has a standardized user interface for all types of user.

  Following are some of the snapshots of the user interfaces used in the system.

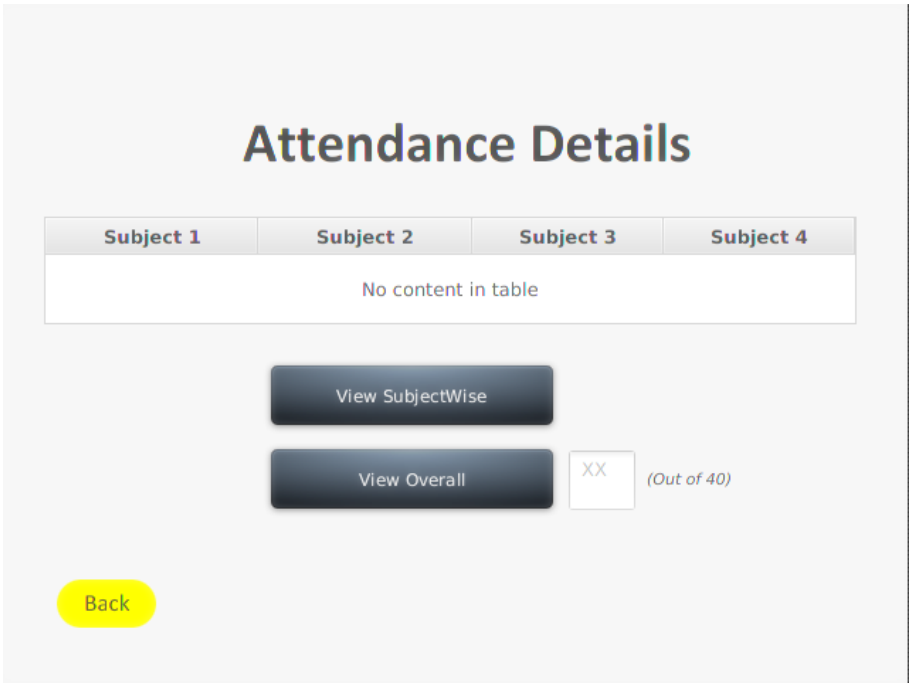
Common Login Page for Student and Staff



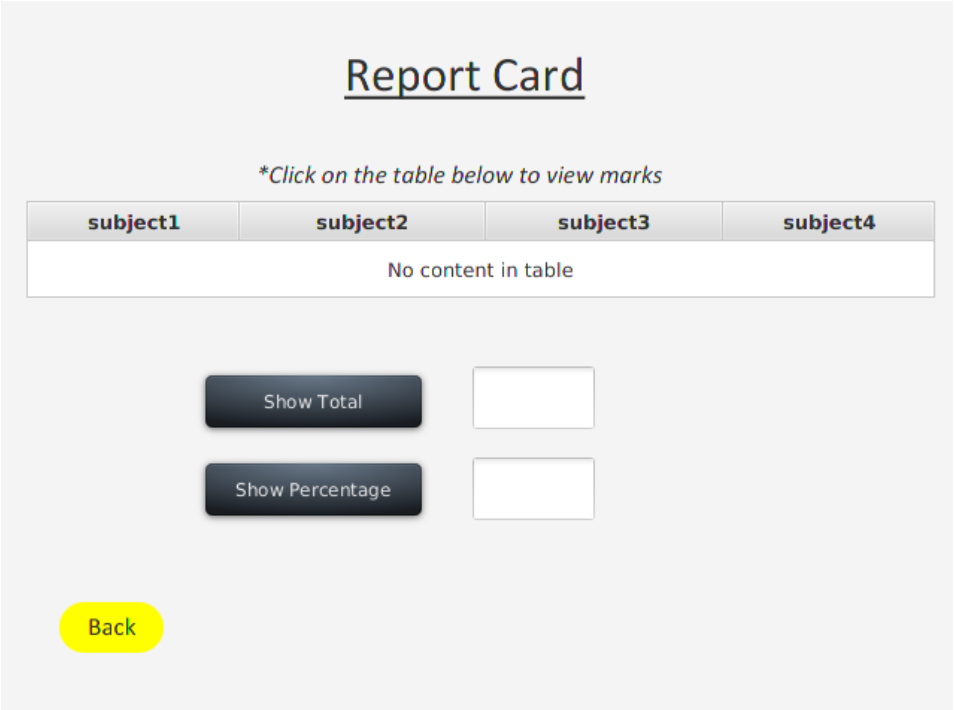
Student Details Page



Attendance Details Page



Marks Details Page



3.2 Hardware Interfaces

Processor: Pentium(R) Dual-core CPU
Hard Disk: 40GB
RAM: 4GB or more

3.3 Software Interfaces

This software package is developed using Java as front end; it uses JavaFX which is a MVC framework used by Java. Microsoft MS Access or MySQL can be used as the back end,primary Database used in the application is MySQL to store the databases.
Operating System: Windows 8/10, Any LINUX
Language: JavaFX (front end)
Language:Java(back end)
Database: MySQL (back end)

3.4 Communications Interfaces

The communication interface for this software is possible using various internet

protocol that we use to transfer the data and information online.

- FTP (File transfer protocol)
- HTTP (Hypertext transfer protocol)

4. System Features

Type of User	Function
1. Administrator	• Login • Add details of students • Add details of Staff • Drop details of student
2. Students	• Login • View personal details • View academic results • View eligibility for placements
3. Faculty	• Login • Updates marks • Updates attendance • View Student details
4. TPO	• Checks marks • Creates eligibility list

4.1 View Attendance

4.1.1 Description and Priority

After signing in, student is automatically navigated to student details page. Here upon clicking the view attendance details, student is again navigated to new page wherein he can view subject-wise attendance or total attendance.

4.1.2 Stimulus/Response Sequences

Students on clicking view attendance button, a backend java program runs which opens another page, wherein students can either see subject wise attendance or overall total attendance. On clicking subwise

attendance, a query is fired at backend which retrieves the results from the attendance repository.

4.1.3 Functional Requirements

The only functional requirement needed to carry out the operation is that the student id should be correct. If incorrect username or password is entered, a dialog appears telling that the username or password is incorrect.

4.2 View Marks

4.2.1 Description and Priority

After signing in, student is automatically navigated to student details page. Here upon clicking the view attendance details, student is again navigated to new page wherein he can view subject-wise attendance or total attendance.

4.2.2 Stimulus/Response Sequences

Students on clicking view marks button, a back end java program runs which opens another page, wherein students can either see subject wise marks, overall total marks, and percentage . On clicking sub wise marks, a query is fired at back end which retrieves the results from the attendance repository. Similarly for total and percentage, also a query is fired on clicking the button.

4.2.3 Functional Requirements

The only functional requirement needed to carry out the operation is that the student id should be correct. If incorrect username or password is entered, a dialog appears telling that the username or password is incorrect.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

SIS that we are going to develop can be used as the main system within the many colleges or universities of different states where the it will be implemented to view student's information in a simple and efficient way. Therefore, it is expected that the database would perform functionally all the requirements properly and maintain all the transactions details.

5.2 Safety Requirements

Since the data we are going to access is stored on the database, it may crash due to hardware or software failures. Hence it is must to keep the backup of the data so that it is not lost.

5.3 Security Requirements

- System will use secured database
- Normal users can just read information but they cannot edit or modify anything
- Proper user authentication should be provided
- There should be separate accounts for admin and end users such that no user can access the database and only admin has the rights to update the database.

5.4 Software Quality Attributes

SQA attributes define the quality of the product as per the fulfillment of demands of the end user. The users are assumed to have basic knowledge of the computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to tell how portal works must be sufficient to educate the users on how to use the system without any problems.

The admin provides certain facilities to the users in the form of:-

- Backup and Recovery
- Data migration i.e. whenever user registers for the first time then the data is stored in the server
- Maintaining files i.e. File Organization
- The server must be maintained regularly and it has to be updated from time to time

6. Other Requirements

6.1 Error handling

If any of the validations or if authentication fails then appropriate error messages will be prompted to the user for doing the needful.

6.2 Reliability

Data validation and verification needs to be done at every stage of activity.

- Validating user input

Appendix A: Glossary

- SIS: Student Information System
- CPU: Central processing unit
- RAM: Random access memory

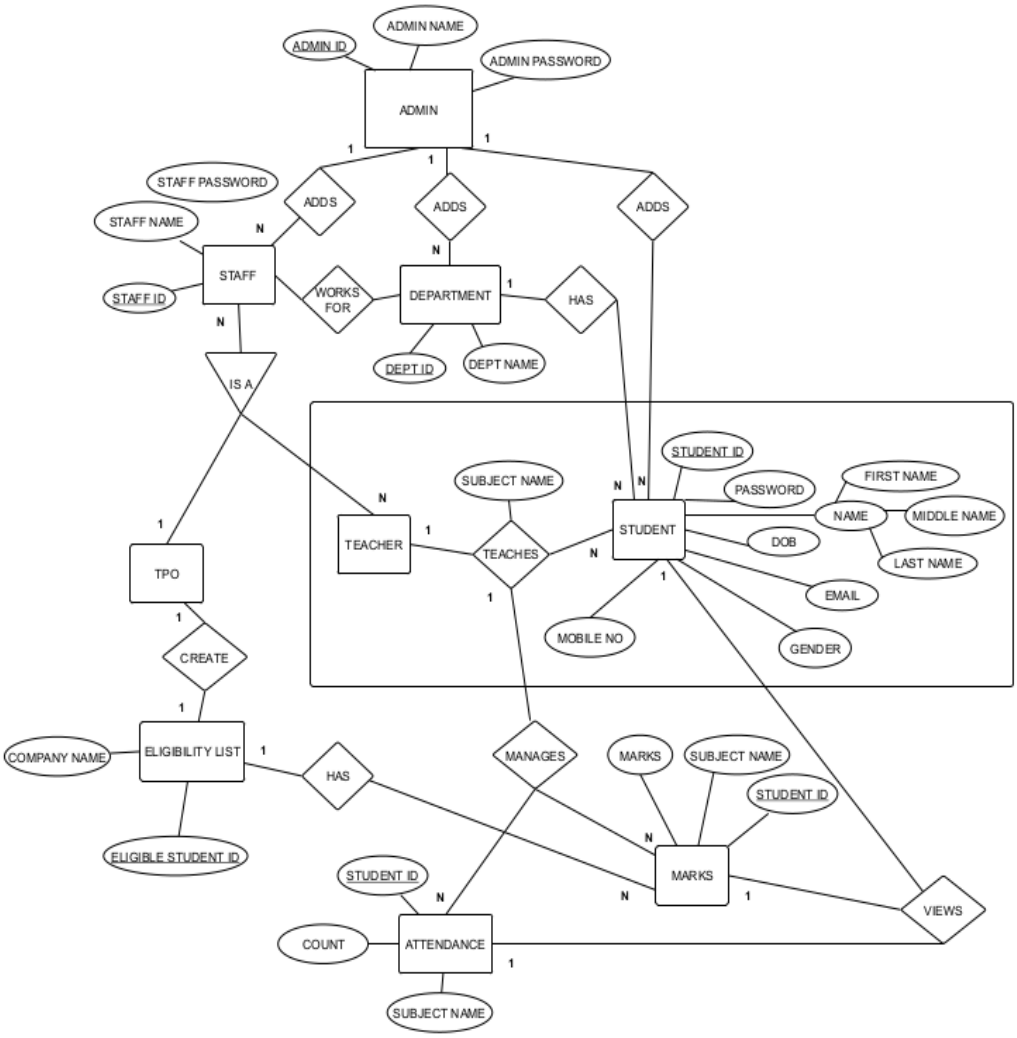
Structured Diagrams

1. E-R Diagram

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and the relationship between them. ER modeling helps you to analyze data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modeling before implementing your database. This model is based on three basic concepts:

- Entities
- Attributes
- Relationships

The E-R Diagram for the system:



- **STUDENT:**
  Attributes: STUDENTID, PASSWORD, NAME, MOBILE, DOB, GENDER, EMAIL, DEPARTMENT, YEAR
- **STAFF:**
  Attributes: STAFFID, STAFF_PASSWORD, STAFF_NAME
- **MARKS:**
  Attributes: SUBJECT_NAME, MARKS, STUDENTID
- **ATTENDANCE:**
  Attributes: STUDENTID, SUBJECT_NAME, COUNT
- **ELIGIBILITY LIST:**
  Attributes: STUDENTID, COMPANY_NAME
- **TEACHES (RELATIONSHIP):**
  Attributes: TEACHES
- **DEPARTMENT:**
  Attributes: DEPT_ID, DEPT_NAME
- **ADMIN:**
  Attributes: ADMIN_ID, ADMIN_NAME, ADMIN_PASSWORD

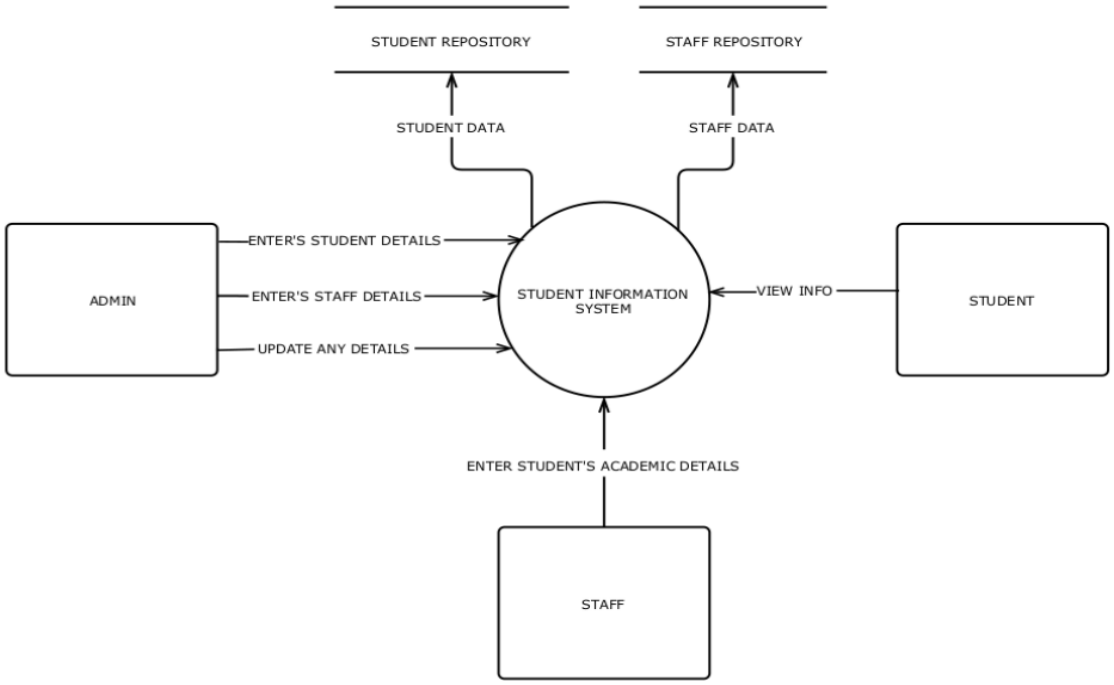
2. Data-Flow Diagram

Data flow diagrams describes the processes that are involved in a system to transfer data from the input to the file storage and result generation. DFD are used to graphically represent the flow of data in a business information system. There are four basic symbols that are used to represent a data-flow diagram:

1. A process receives input data and produces output with a different content or form.
2. A data-flow for data to move from one part of the information system to another.
3. A data store or data repository is used in a data-flow diagram to represent a situation when the system must retain data.

4. An external entity is a person, department, outside organization that provides data to the system or receives outputs from the system.

2.1 DFD – Level 0 (Context Diagram)



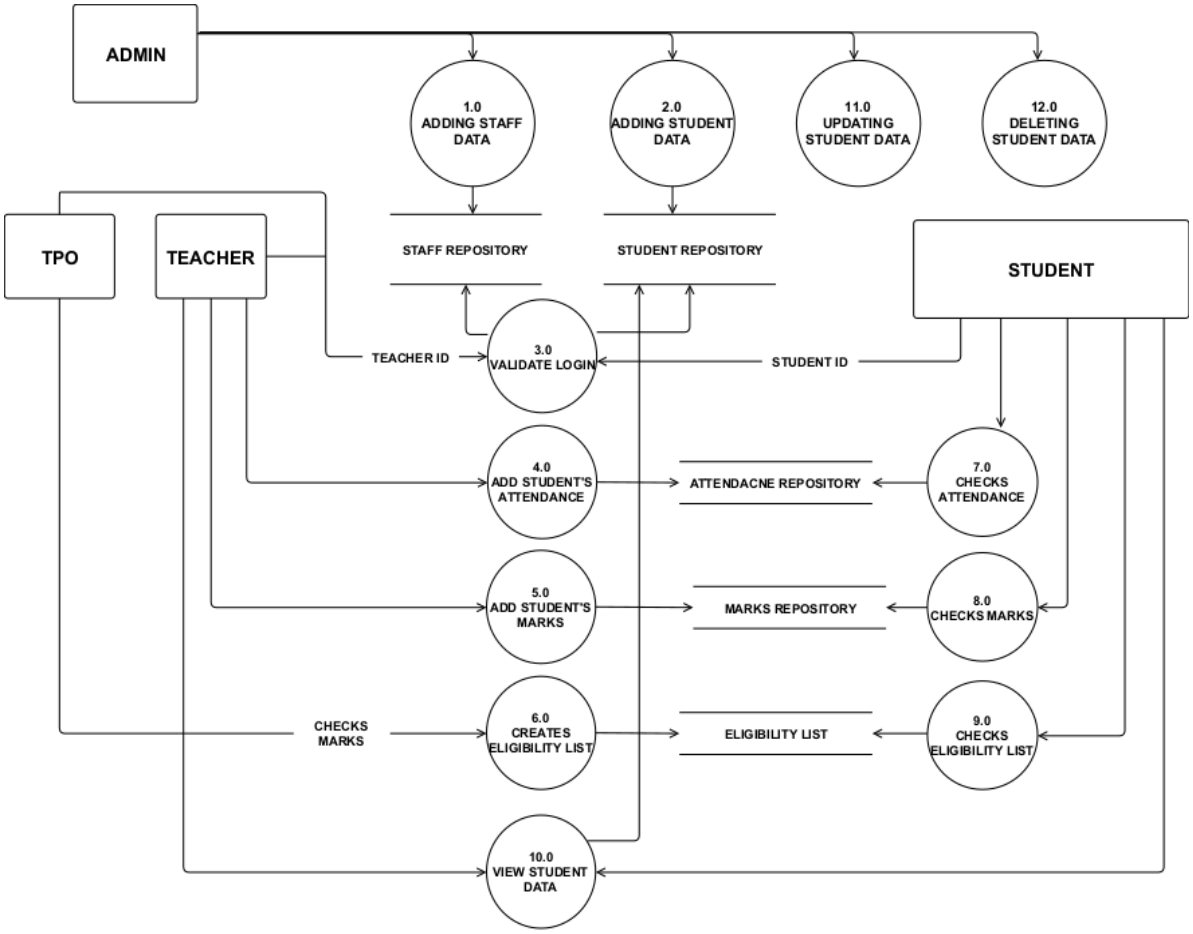
The entities are Administrator, Student, and Staff.

Administrator enters all the students and staff details through an user interface and then these details will be stored in student and staff repositories respectively.

Staff can log in to the application and then add student details like marks and attendance. This again is stored in another database.

Students can simply log in to the application and view the attendance and marks details.

2.1 DFD – Level 1

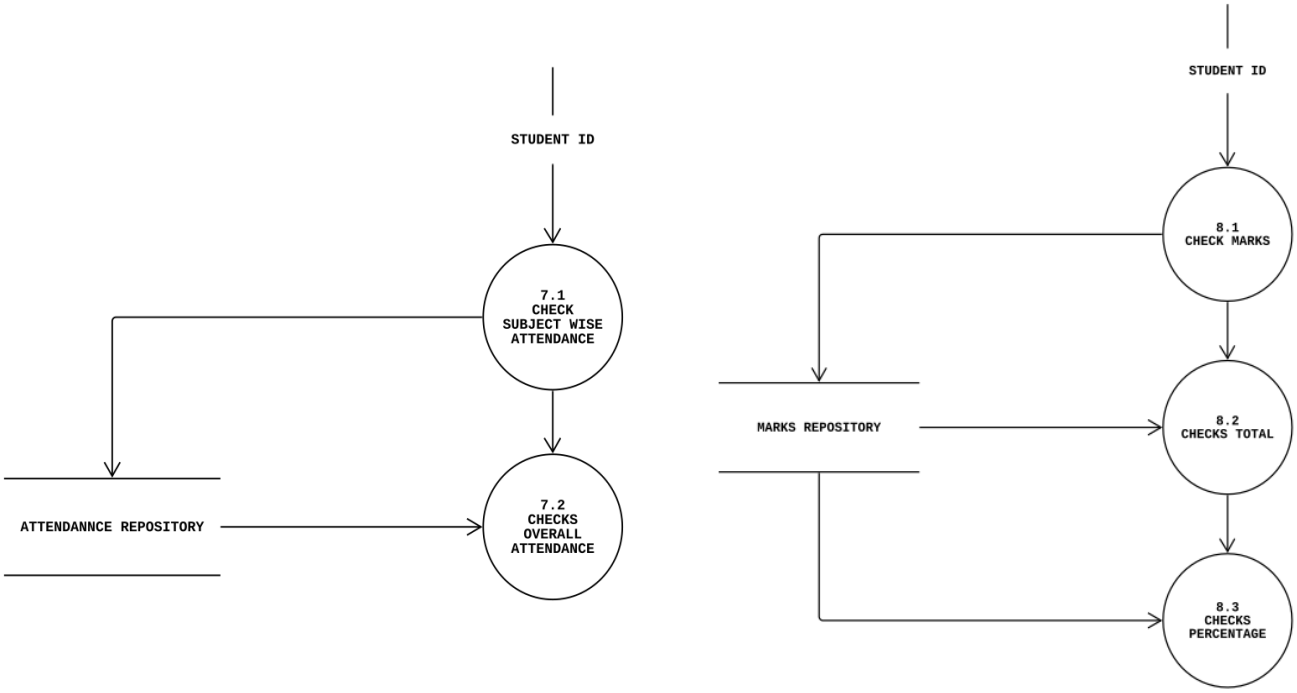


Level 1 DFD has following processes:

1. **Adding Staff data :** This process is used by administrator to add all the staff details which includes staff_name, staff_id, designation as the main fields into the Staff Details database.
2. **Adding Student data :** This process is used by administrator to add all the student details which includes student_name, student_id as the main fields into the Student Details database.
3. **Validate login:** This process is used when either of the student or the staff is trying to login using their respective user-id and password.

4. **Add Attendance details :** This process is used by the faculty to add the attendance of the student. The attendance is added based on the subject the staff teaches and the student id.
5. **Add Marks Details :** This process is used by the faculty to add the marks of the student. The attendance is added based on the subject the staff teaches and the student id.
6. **Creates Eligibility List :** This process is used by the TPO to select those students who are eligible for placements based on the marks of the student.
7. **Checks Attendance:** This process is used by the student to check upon the attendance, which is filled up by the faulty. The attendance is retrieved from attendance repository.
8. **Checks Marks :** This process is used by the student to check upon the marks, which is filled up by the faulty. The marks is retrieved from marks repository.
9. **Check Eligibility List:** This process is used by the student to view whether he/she is eligible for the placement drive or not.
10. **View Student data:** This process can either be used by student or staff to view sudent details like student name, id, even academic details like marks.
11. **Update Student Data:** This process is used by the administrator to update any details of the student. Administrator first verifies the details using student-id and then proceed for updation.
12. **Delete Student data :** This process is used by the administrator to delete the details of the student. Deletion will delete all the records of the student

2.2 DFD – Level 2



Process 7 is further divided into two more processes:

- 7.1 student can check attendance subject wise or
- 7.2 can view the total attendance directly.

Similarly, Process 8 is further divided into three more processes,

- 8.1 student can check marks either subject wise or
- 8.2 can view total marks or
- 8.3 can directly check for percentage.

Object-oriented Diagrams

1. Use case diagrams

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a "system" is something being developed or operated, such as a web site. The "actors" are people or entities operating under defined roles within the system. Use case diagrams are valuable for visualizing the functional requirements of a system that will translate into design choices and development priorities

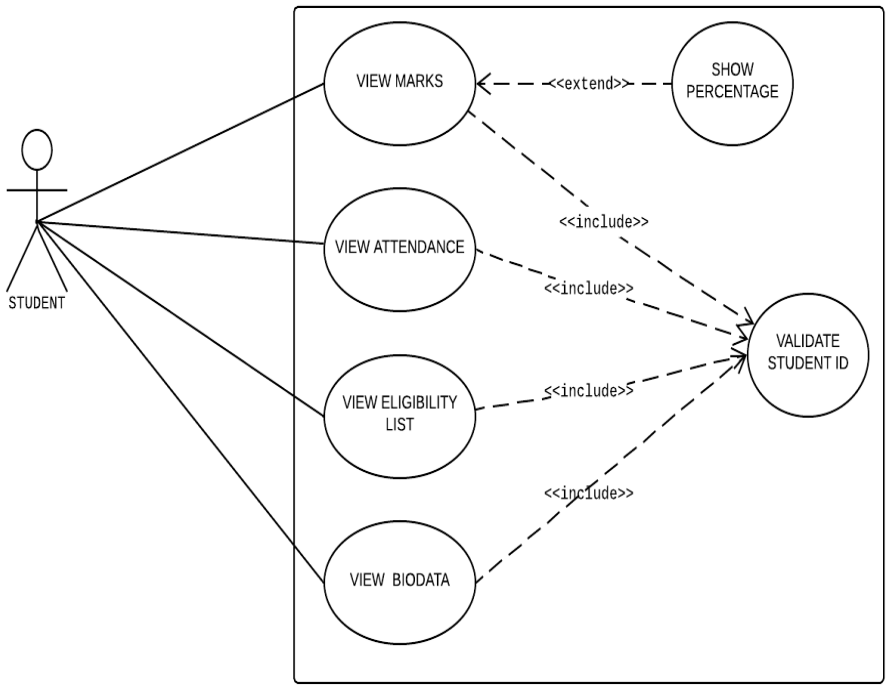


Fig : Student Use-Case

a) Actors

- Students

b) Use Cases

- View Marks
- View Attendance

- View Eligibility list
- View BioData

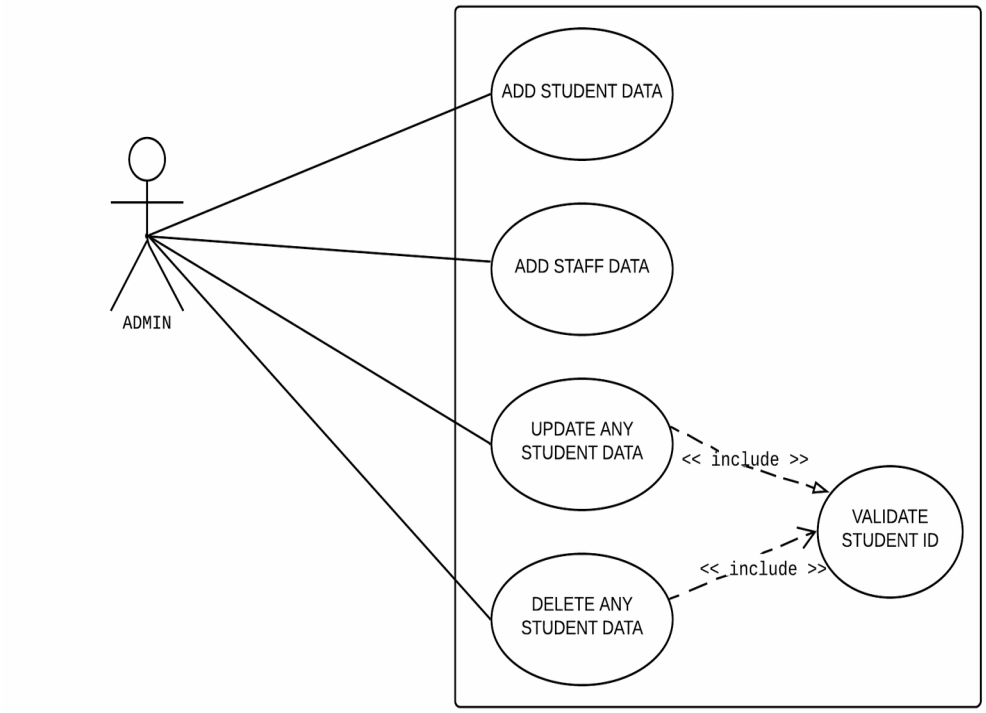


Fig : Admin Use-Case

a) Actors

- Administrator

b) Use Cases

- Add Student Data
- Add Staff Data
- Update any Student Data
- Delete any Student data

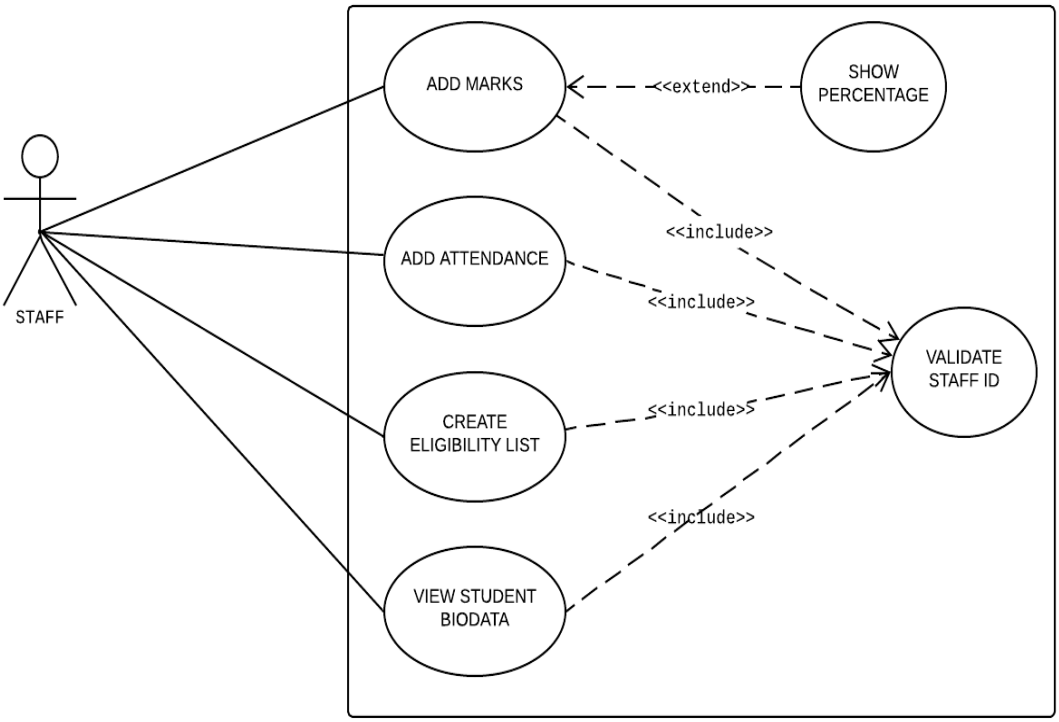


Fig : Staff Use-Case

a) Actors

- Staff

b) Use Cases

- Add Marks
- Add Attendance
- Create Eligibility List
- View Student BioData

2. State transition diagram

State-transition diagrams describe all of the states that an object can have, the events under which an object changes state (transitions), the conditions that must be fulfilled before the transition will occur (guards), and the activities undertaken during the life of an object (actions). State-transition diagrams are very useful for describing the behavior of individual objects over the full set of use cases that affect those objects.

2.1 State-transition diagram for View Eligibility list use-Case

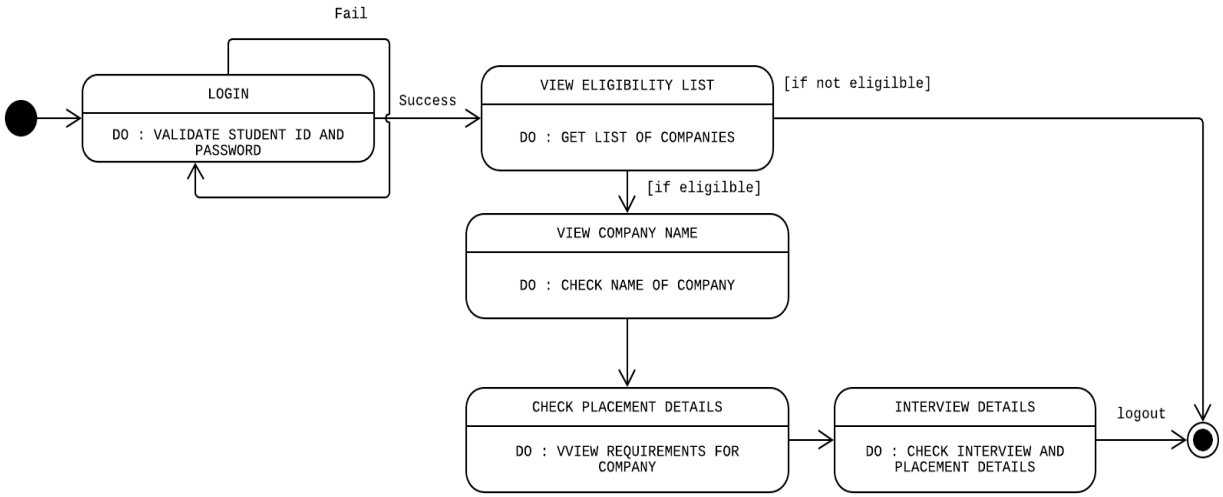


Fig : State-transition diagram for View Eligibility list use-Case

2.2 State-transition diagram for View Marks use-Case

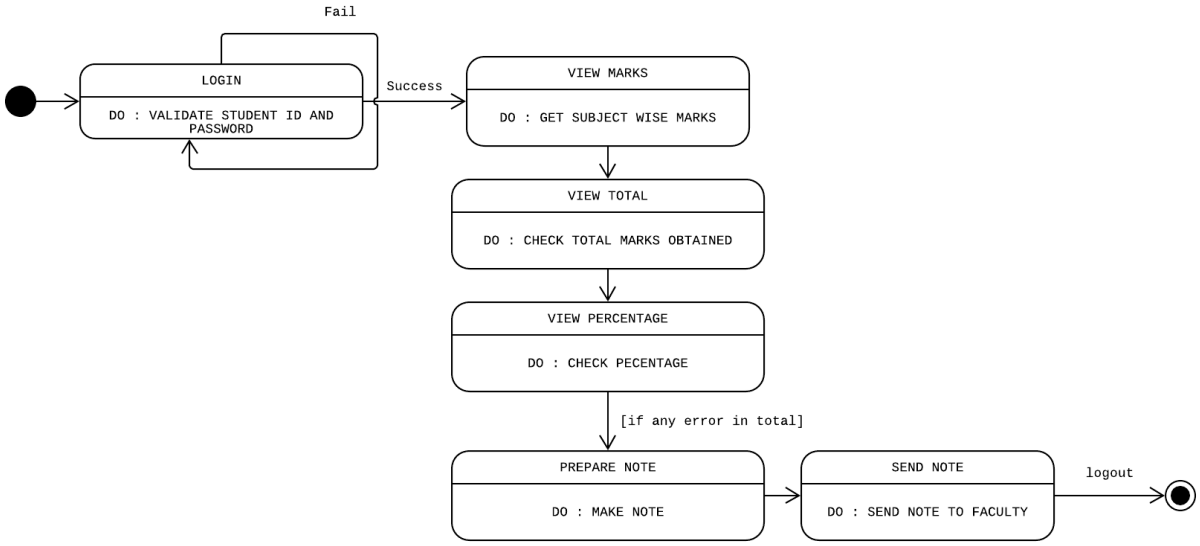


Fig: State transition diagram for View Marks Use-case

2.3 State-transition diagram for View Attendance use-Case

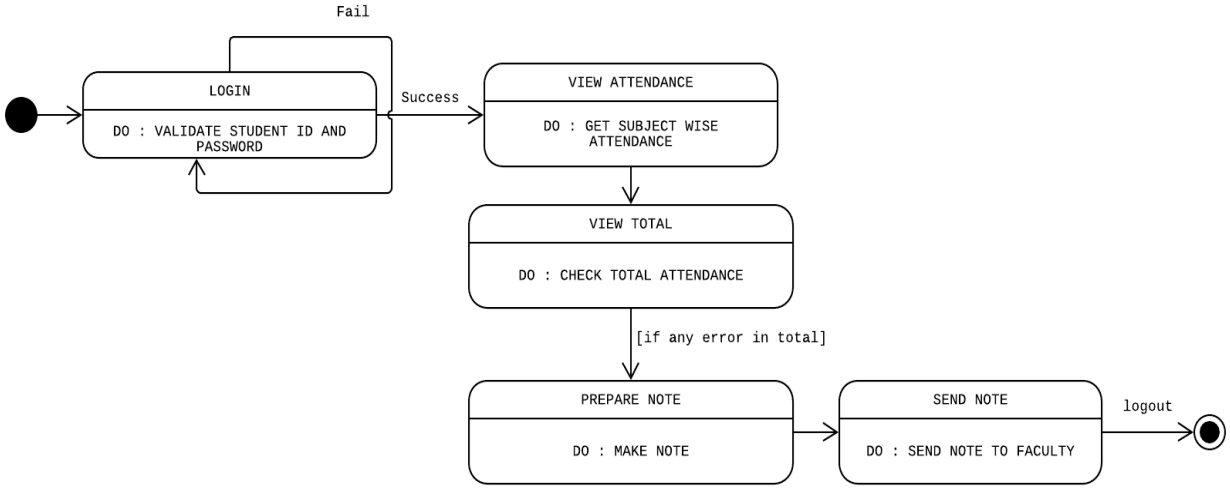


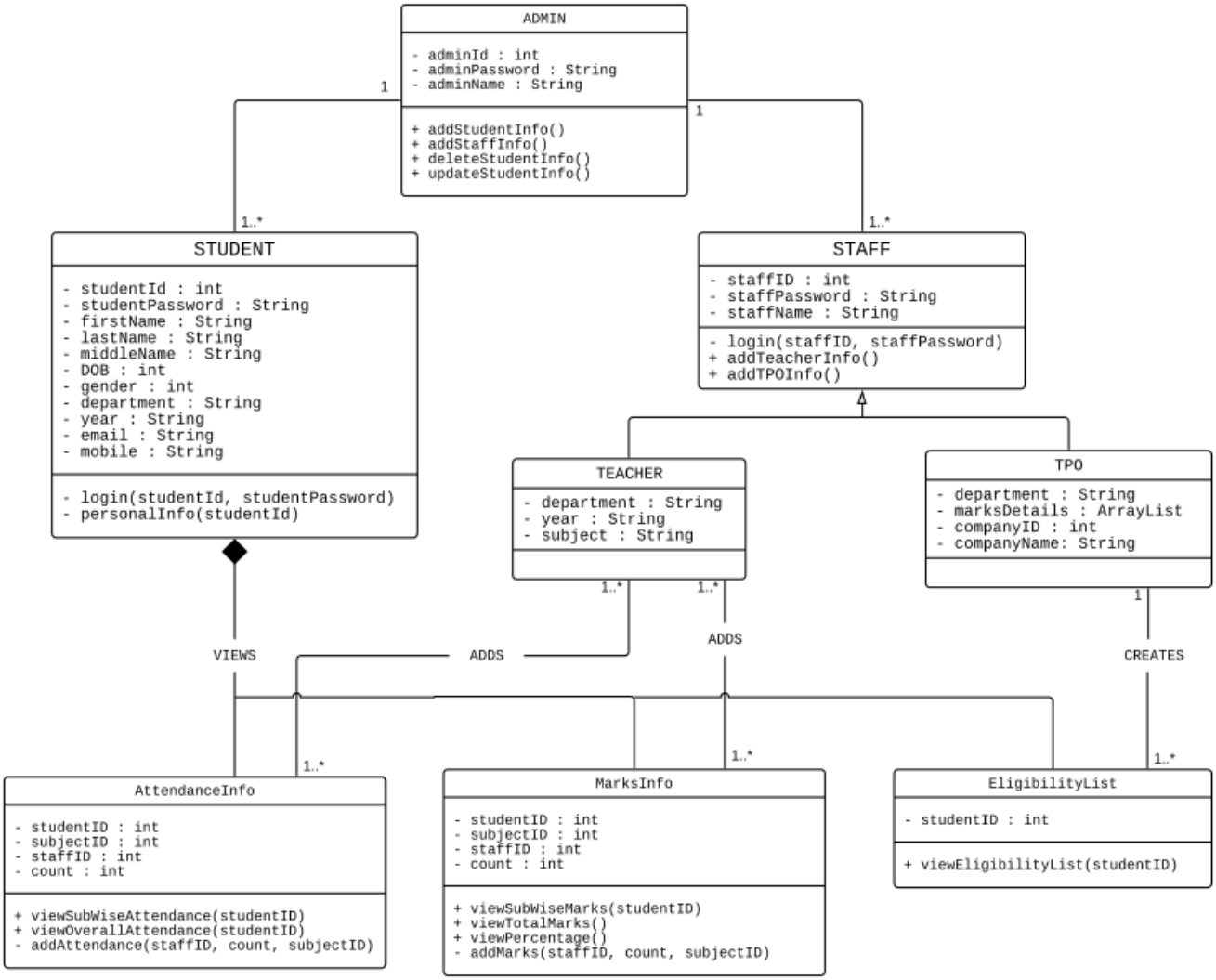
Fig : State-transition diagram for View Attendance use-Case

3. Class diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

A class is a description of a group of objects all with similar roles in the system, which consists of:

- **Structural features** (attributes) define what objects of the class "know"
  - Represent the state of an object of the class
  - Are descriptions of the structural or static features of a class
- **Behavioral features** (operations) define what objects of the class "can do"
  - Define the way in which objects may interact
  - Operations are descriptions of behavioral or dynamic features of a class



Class Data Models :

- **Student**
  student-id, student_password, firstName,lastName, middleName, DOB, email, gender, year, department, mobile
- **Staff**
  staff_id, staffName
- **Teacher**
  department,year,subject
- **TPO**
  department, marksDetails, companyName, companyID
- **AttendanceInfo**
  studentID, staffID, subjectName, count

- **MarksInfo**
  studentID, staffID, subjectName, count
- **EligibilityList**
  studentID

Class Process Models :

- **Student**
  login(studentID, password), personalInfo(studentId)
- **Staff**
  login(studentID, password)
- **AttendanceInfo**
  viewSubWiseAttendace(studentID),viewOverallAttendance(studentID), addAttendance(staffID,count, subjectID)
- **MarksInfo**
  viewSubWiseMarks(studentID),viewTotalMarks(),viewPercentage(), addMarks(staffID,count, subjectID)
- **EligibilityList**
  viewEligibilityList(studentID)

4. Activity diagram

4.1 Activity diagram to view Attendance

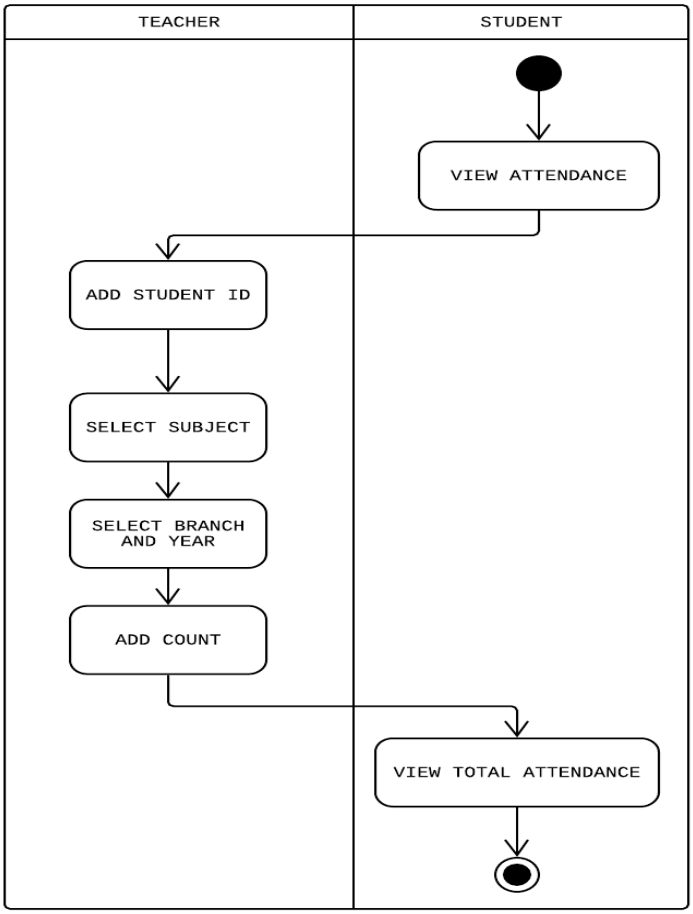


Fig: Activity diagram to view Attendance

Activities

- Teacher/Faculty is the one responsible for adding marks of the student. Faculty thus adds marks of the student studying in the respective subject of the teacher. Faculty selects the year, branch, and subject name whose marks details are to be filled in. Faculty simply write the student id and enter the marks.
- After logging in, student goes to Student Details page, where student can click on view Marks button. Upon clicking the button student can have choice either to view total marks, or subject wise marks or directly opt for percentage.

4.2 Activity diagram to view Marks

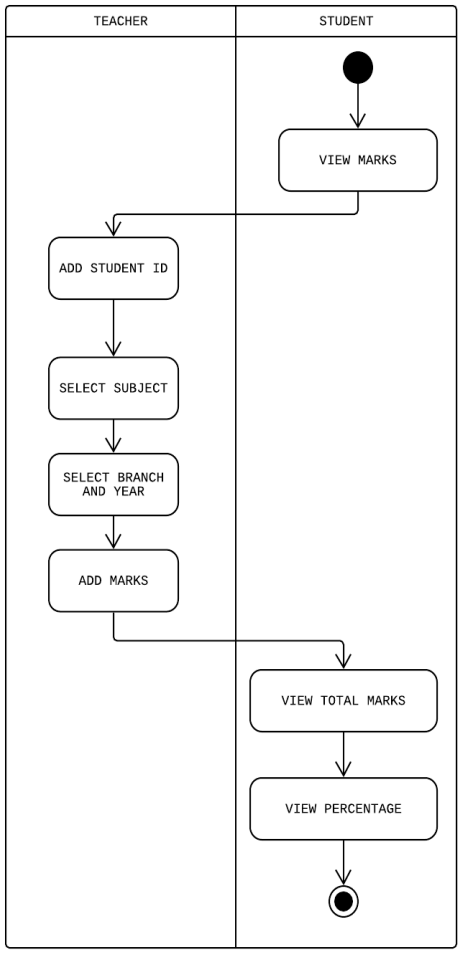


Fig : Activity diagram to view Marks

Activities

- Teacher/Faculty is the one responsible for adding marks of the student. Faculty thus adds marks of the student studying in the respective subject of the teacher. Faculty selects the year, branch, and subject name whose marks details are to be filled in. Faculty simply write the student id and enter the marks.
- After logging in, student goes to Student Details page, where student can click on view Marks button. Upon clicking the button student can have choice either to view total marks, or subject wise marks or directly opt for percentage.

4.3 Activity diagram to view Eligibility list

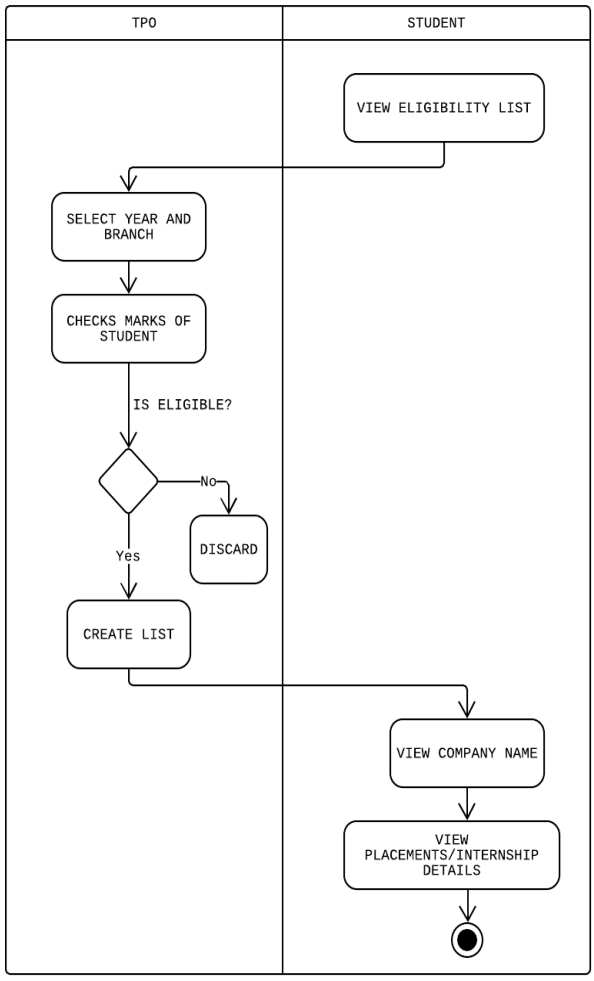


Fig : Activity diagram to view eligibility list

Activities

- TPO can view marks details of the student after faculty adds it. Now, TPO has a list of companies along with its criteria. TPO enters the name of the company who is bound to come for placements and enters the minimum criteria, On entering the criteria, a list of student id and student names is generated who are eligible for the drive. This list is stored into database.
- After logging in, student goes to Student Details page, where student can click on view Eligibility list. Upon clicking the button student can see whether or not he is eligible for placements for the company which is going to come.

5. Sequence diagram

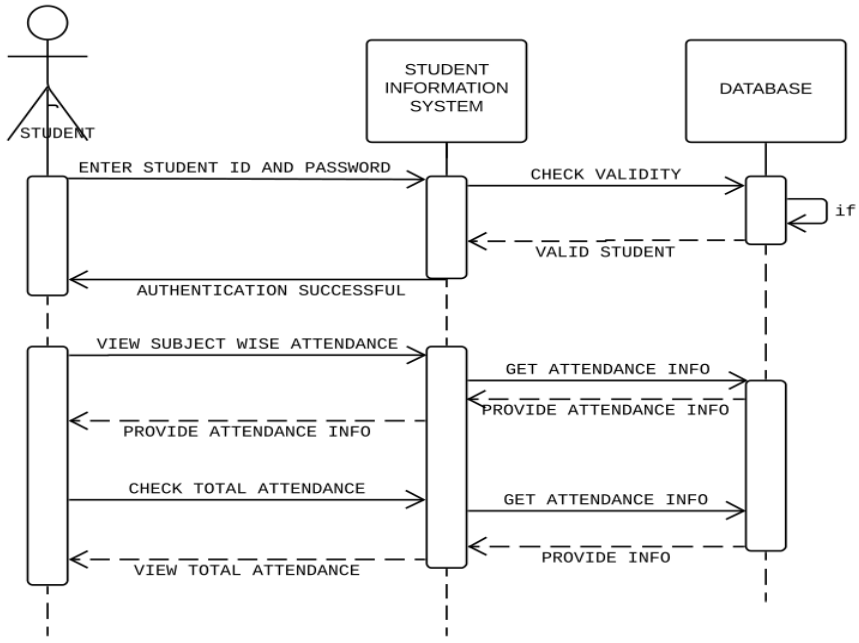


Fig : Sequence diagram to view attendance

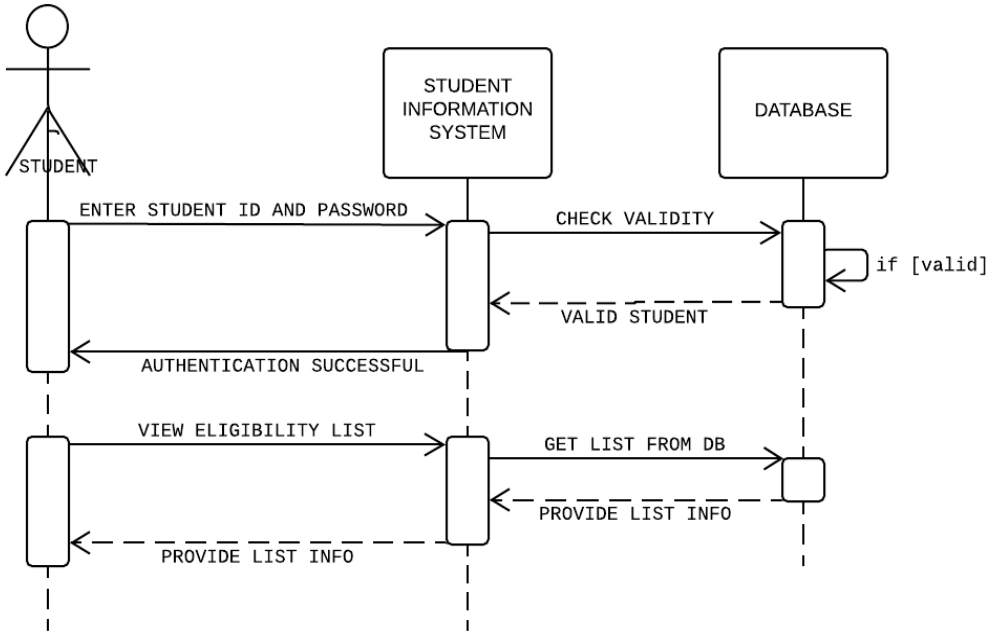


Fig : Sequence diagram to view eligibility list

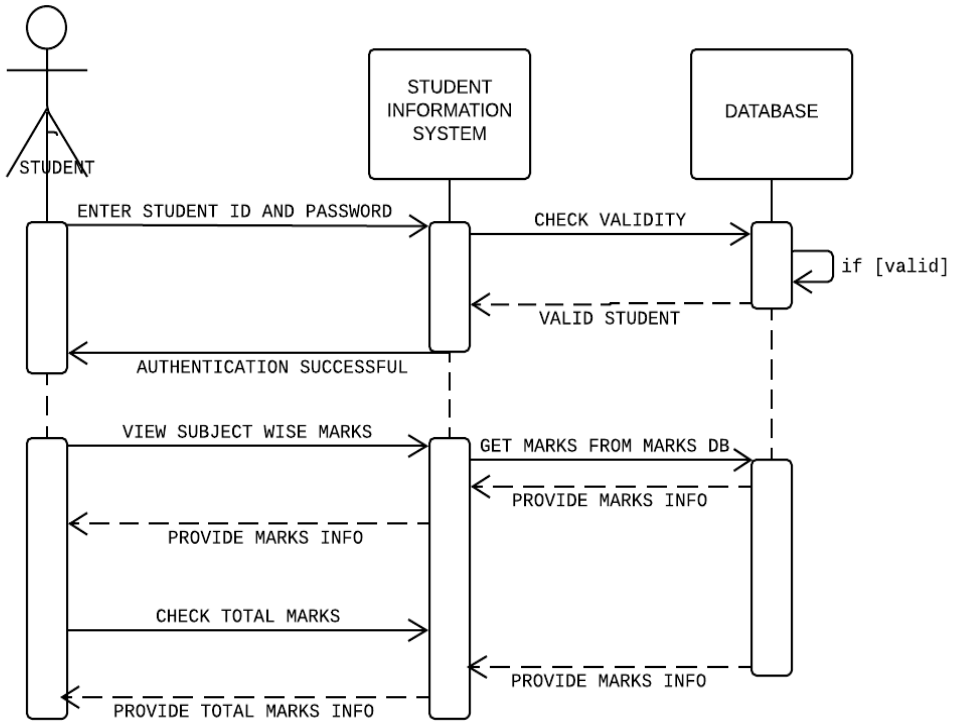
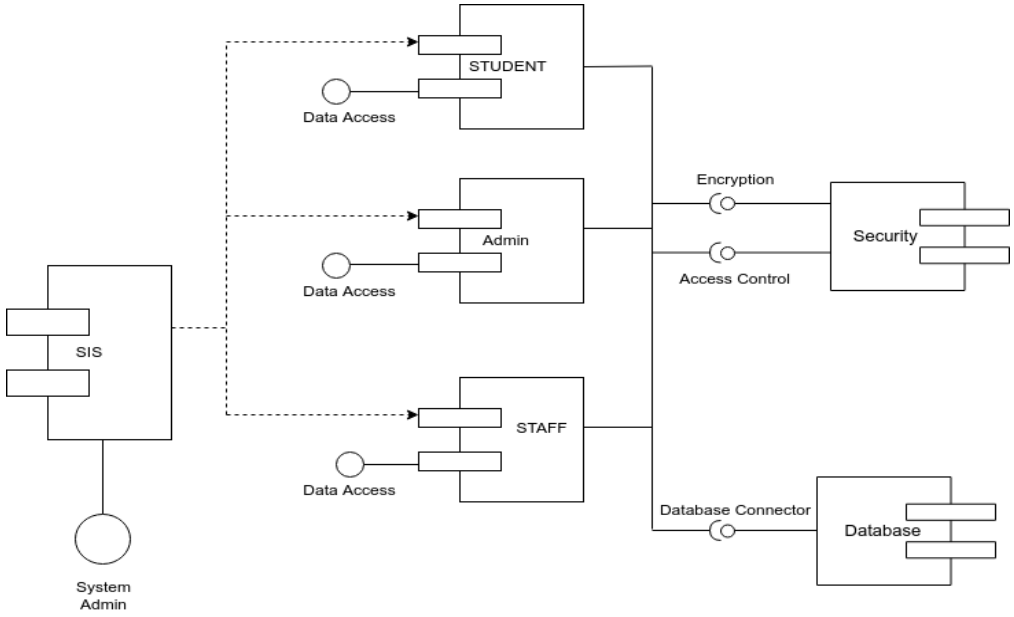
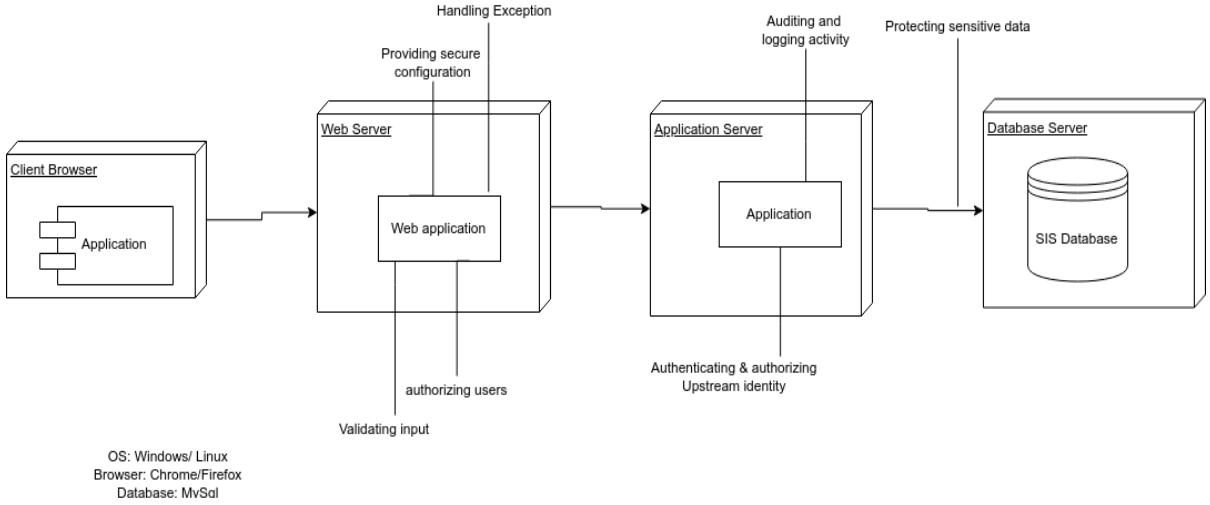


Fig : Sequence diagram to view Marks

6. Component diagram



7. Deployment diagram



Web Engineering:

Web engineering is basically all about designing and promoting web based systems. The application of systematic and quantifiable approaches to cost-effective analysis, design, implementation, testing, operation, and maintenance of high-quality Web applications.

Web engineering versus software engineering Developing Web-based systems is significantly different from traditional software development and cause many additional challenges. There are slight differences in the nature and life cycle of Web-based and software systems. Multidisciplinary Building a complex Web-based system calls for knowledge and expertise from many different disciplines and requires a team of various people with expertise in different areas.

1) LoginPage

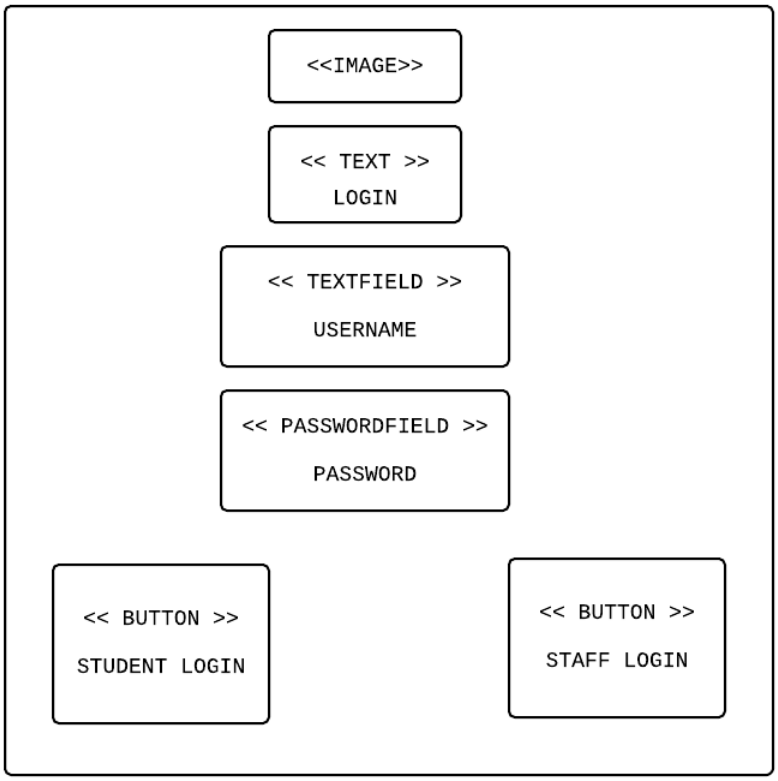


Fig : Login Page

2) Students Details

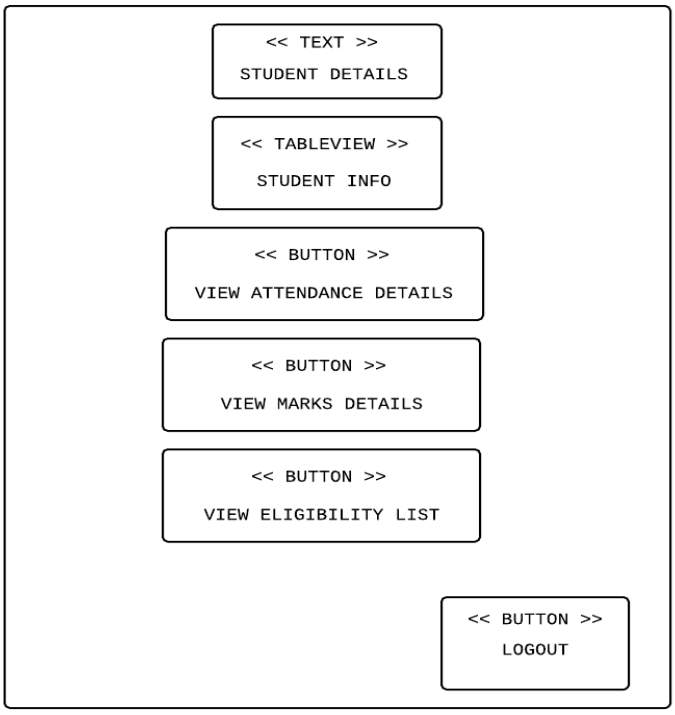


Fig : Student details page

3) Attendance Details page

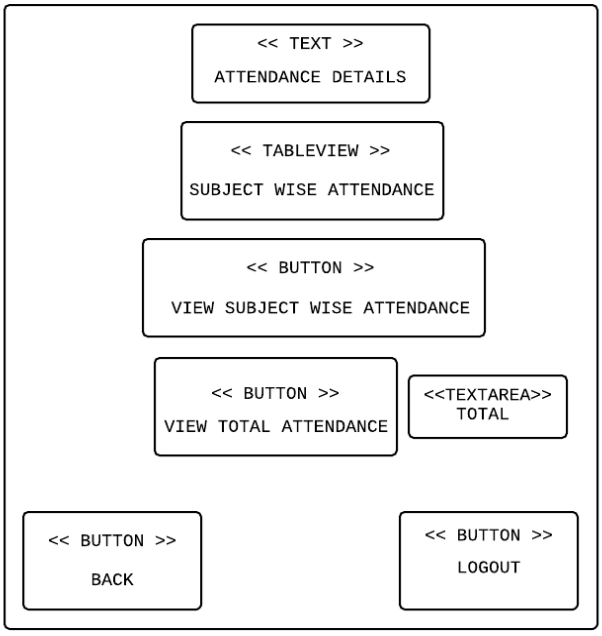


Fig : Attendance details page

4) Marks Details

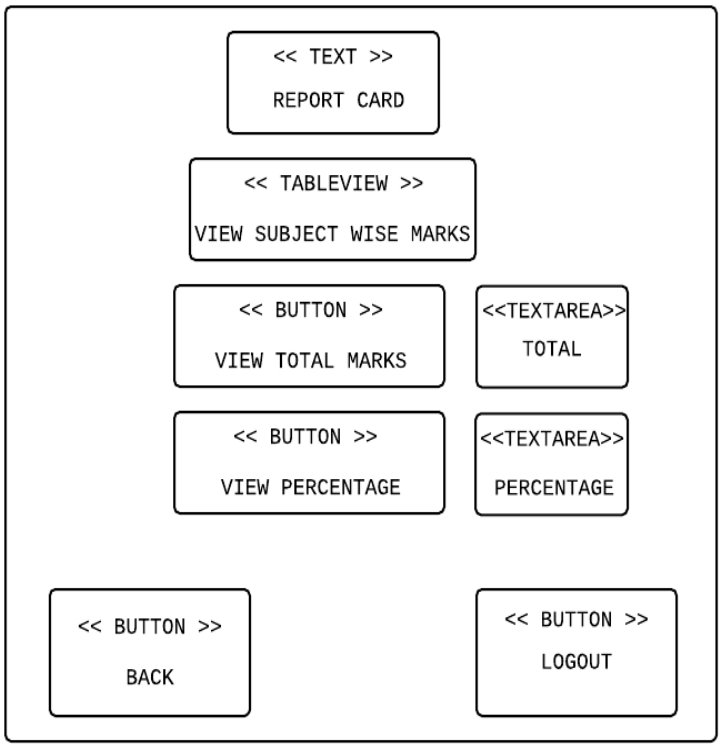


Fig : Marks details page

5) Eligibility List page

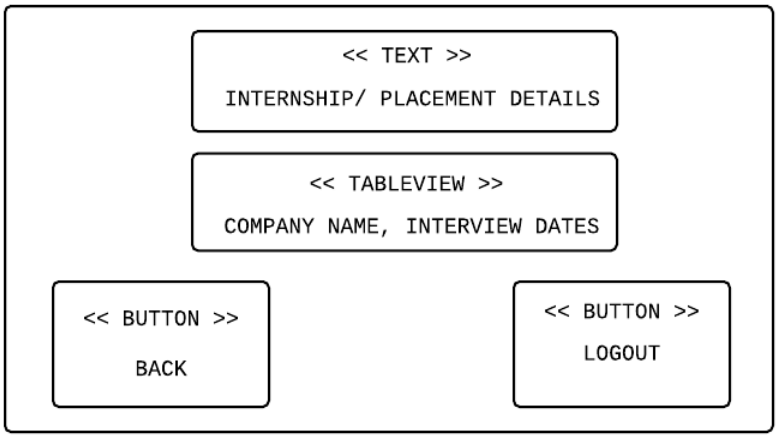
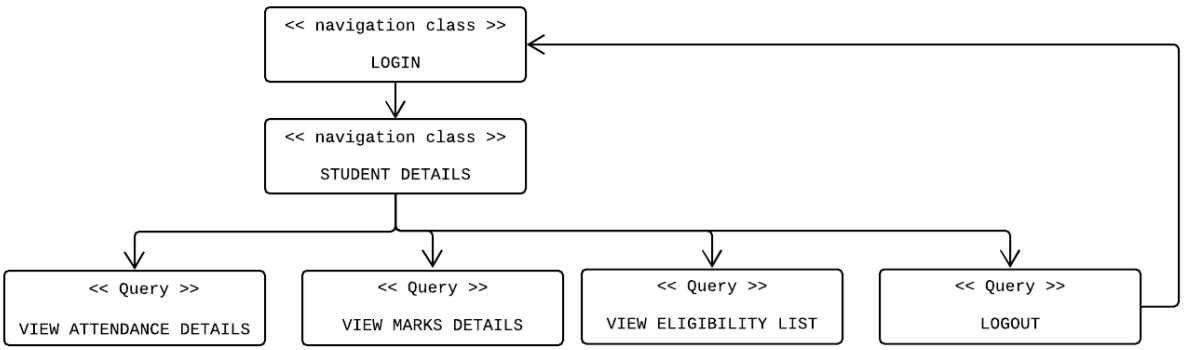
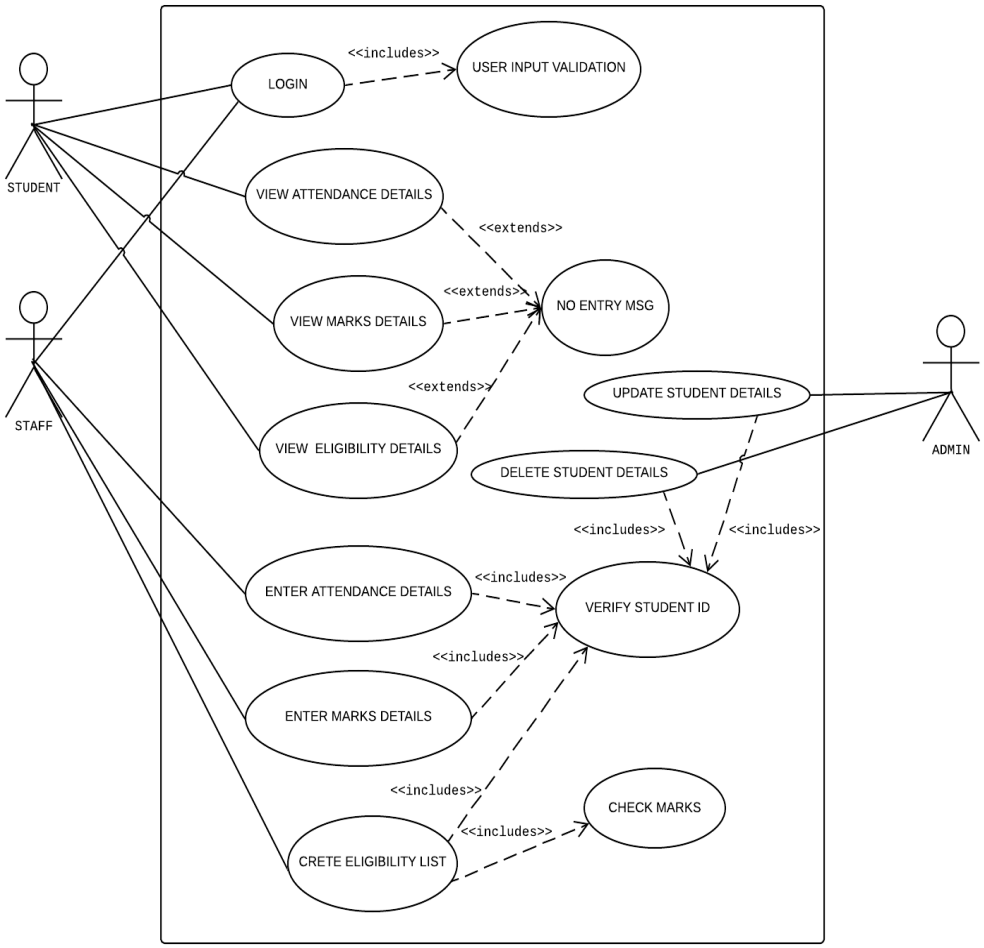


Fig :Eligibility list page

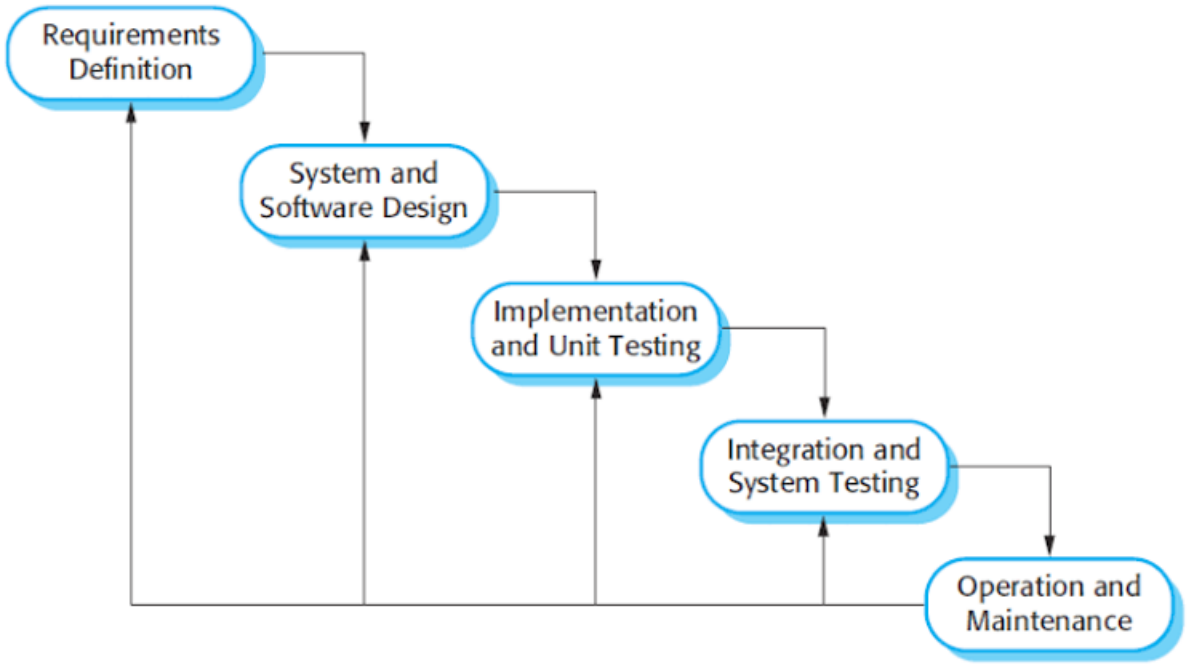
6) Hypertext Structural Model



7) Security Use case



The proposed system has used waterfall model for developing the Workflow Management System.
In "The Waterfall" approach, the whole process of software development is divided into separate process phases. The phases in Waterfall model are: Requirement Specifications phase, Software Design, Implementation and Testing & Maintenance. All these phases are cascaded to each other so that second phase is started as and when defined set of goals are achieved for first phase and it is signed off, so the name "Waterfall Model". All the methods and processes undertaken in Waterfall Model are more visible.



Requirements Definition:
All possible requirements of the system to be developed are captured in this phase. Requirements are set of functionalities and constraints that the end-user (who will be using the system) expects from the system. The requirements are gathered from the end-user by consultation, these requirements are analyzed for their validity and the possibility of incorporating the requirements in the system to be development is also studied. Finally, a Requirement Specification document is created which serves the purpose of guideline for the next phase of the model.
The proposed Student Information system is a system that is capable of Managing student details and retrieving it upon request. The system is developed with functional and non-functional requirements.

Functional Requirements: The following requirements are captured for the intended use of the system:

- Student upon receiving student id and password can directly log in to system.
- Staff upon receiving staff id and password can directly log in to system
- By Designation of staff the access is provided by the system.
- Only admin is able to add the new user to the system.
- Student can only view the profile and academic details.

Non-functional requirements: The proposed system is designed to fulfill the following non-functional requirements.

- Performance Requirements: Performance of the system is dependent on the bandwidth of the internet.
- Security Requirements: There is only one authorized person who can see the student's confidential information apart from the student himself. The information of the student is only available for the administrator.
- Software Quality Attributes: The system is user friendly, inter-operable and flexible.

System & Software Design:

The requirement specifications from first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture. The system design specifications serve as input for the next phase of the model.

Implementation & Unit Testing:

On receiving system design documents, the work is divided in modules/units and actual coding is started. The system is first developed in small programs called units, which are integrated in the next phase. Initially home page, login page and sign up page was developed. After that database was created and "view profile" and "view experience" was created. Each unit is developed and tested for its functionality and after testing each unit meet their specification needs.

Integration & System Testing:

As specified above, the system is first divided in units, which are developed and tested for their functionalities. These units or pages were integrated into a complete system or folder in the Integration phase and tested to check if all modules/units coordinate between each other and the system as a whole behaves as per the specifications. After successfully testing the software, it is hosted locally.

Unit testing for “Login form” ,”Add Marks Details”
LOGIN FORM:

Sr No	Test Case	Expected Result	Test Result
1	Enter valid user name and password & click on login button	Software should display main window	Successful
2	Enter invalid	Software should not display main window	Successful

ADD MARKS FORM:

Sr No	Test Case	Expected Result	Test Result
1	On the click of ADD MARKS button	At first the staff logs in using staff id and password, if staff is a faculty than only he can add marks of the student, if not he cannot.	Successful

Operations & Maintenance:
This phase of "The Waterfall Model" is virtually never ending phase. Generally, problems with the system developed (which are not found during the development life cycle) come up after its practical use starts, so the issues related to the system are solved after deployment of the system. Not all the problems come in picture directly but they arise time to time and needs to be solved; hence this process is referred as Maintenance.

Deployment Cost of the System:
Since the system is developed in Java, it must be uploaded and run on server. Hence cost for deployment is directly proportional to that of a server.

1- Cost to deploy a website
Here are the four things we need to do to deploy a website.

- Web hosting and domain name

  1. Domain Cost new: Rs700/- ~ Rs1000/-
  2. Web Host Cost Shared: Rs200/- ~ Rs1000/-
  3. VPS: Rs1000/- ~ Rs3500/-

- Web design (icons, logos, themes)
  Designer Fees:
  - Average: Rs2000/ hour
  - Highest: Rs6000/hour

- Web content (writing and relevant images / videos)
  Writer Fees:

- Average: Rs2000/hour

◆ Web development (special functions and backend systems)

Web developer fees:

- Average: Rs2000/hour

These are the primary areas where we spend our money on.

We can use WordPress as the foundation for running our website and use either free or premium designed templates.

2- Cost to host a website:

To host website the cost price for “GoDaddy.com” is Rs3000/month