

ST. XAVIER'S COLLEGE OF MANAGEMENT & TECHNOLOGY

NAAC Accredited with B++ Grade (1st Cycle)
Digha-Aashiyana Road, Digha, Patna-800011, Bihar



BACHELOR OF COMPUTER APPLICATION
SESSION-2020-2023

PROJECT REPORT

Attendance Management System

Submitted to ARYABHATTA KNOWLDGE UNIVERSITY

For the partial fulfilment for the award of the degree of Bachelor of computer application (BCA)

Under the Guidance - **Mr. Prakash Upadhyay**
Asst.prof., Dept. of Computer Science

BY:

Akash Kumar [REG NO. 20303302010]
Apurv Kumar [REG NO. 20303302044]
Archana Kumari [REG NO. 20303302048]

Table of Content

<i>S. No.</i>	<i>Title</i>	<i>Page No.</i>
1.	Introduction 1.1 About 1.2 Purpose 1.3 Scope	10
2.	System Analysis 2.1 Introduction 2.2 Existing System 2.3 Proposed System 2.4 Feasibility Study	11-13
3.	Project Description 3.1 Perspective 3.2 Technologies Used for Development 3.3 Technologies Used for Deployment 3.4 Product Features and User Classifications 3.5 Operating Environment 3.6 Hardware Requirements 3.7 Constraints	14-30
4.	Interface Requirements 4.1 User Interfaces 4.2 Hardware Interfaces 4.3 Software Interfaces	31
5.	System Design 5.1 System Design of AMS 5.2 Admin 5.3 Teacher 5.4 Student 5.5 End User Diagram 5.6 3-tier architecture 5.7 Database Design and Implementation 5.8 Module Hierarchy 5.9 ER Diagram 5.10 Gantt Chart 5.11 Pert Chart	32-50

6.	System Testing	51-59
	6.1 Introduction	
	6.2 Testing Methodologies	
	6.3 Test Cases	
7.	Screenshots and Source code	60-95
8.	Future Scope	96
9.	Limitations	97
10.	Conclusion	98
11.	Bibliography	99

ACKNOWLEDGEMENT

We thank almighty God who has provided us the faculty of learning and understanding the phenomenon occurring around me. We wish to express our sincere thanks to our parents for their tremendous contribution and support both morally and financially towards the completion of this project. It gives us immense pleasure to express our sincere gratitude to our faculty guide **Mr. Prakash Upadhyay sir**, for helping us in shaping this project. His support and encouragement helped us to strive towards the best of our potentials.

We also take this opportunity to express our gratefulness to all the faculty members of the department of computer science at St. Xavier's College of Management & Technology, Patna for their help and support.

We gained immensely from doing this project. It has contributed greatly towards increasing our learning curve, both at the theoretical and practical level. we would also like to extend our gratitude to the principal **Fr. (Dr) Martin Poras S.J.**, of St. Xavier's College of Management & Technology, Patna for providing us an opportunity to study in this premier institute of learning.

By:

Akash Kumar

Apurv Kumar

Archana Kumari

CERTIFICATE

This is certified that the project entitled **Attendance Management System** submitted to Aryabhatta Knowledge University that has been carried out by Akash Kumar, Apurv Kumar & Archana Kumari under my guidance to fulfillment of the requirements for the award of Degree of Bachelors in Computer Application (BCA). These students are partially completed this project under my guidance.

.....
Signature of the Principal
Fr. Dr. Martin Poras SJ
St. Xavier's College of
Management & Technology

.....
Signature of the Project
Guide Mr. Prakash Upadhyay
Asst. Professor
Dept. of Computer Science

.....
Signature of External Examiner
Aryabhatta Knowledge University
Patna, Bihar

DECLARATION

We hereby, declare that the work being presented in the project, titled “**Attendance Management System**” as a part of course curriculum of Bachelor of Computer Application (BCA), is an authentic record of our own work carried out under the guidance of Mr. Prakash Upadhyay, Asst. Prof. in the Department of Computer Science, St. Xavier’s College of Management & Technology, Patna.

Group Member:- **Akash Kumar** [REG NO. 20303302010]

Apurv Kumar [REG NO. 20303302044]

Archana Kumari [REG NO. 20303302048]

Course:- Bachelor of Computer Application

ABSTRACT

The purpose of Attendance Management System is to automate the existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Attendance Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather than concentrating on the record keeping. Thus, it will help organizations in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

Using various technologies, we have implemented the AMS application that provides the above functionalities. The software was also tested using the various testing methods and results were positive.

ABOUT COLLEGE

St. Xavier ‘s College of Management and Technology under Aryabhatta Knowledge University was established in 2012. It offers life-oriented and professional courses. St. Xavier ‘s College is a co-educational undergraduate college of Arts and Commerce under Magadh University, Bodh Gaya, started in July 2009 at its temporary College Campus at Digha. In June 2011, the college was shifted to its present location on a sprawling 36-acre campus at S.X.C.M.T on Digha Aashiyana Road.

Both colleges are managed by the Jesuits of Bihar. Jesuits are the members of a Christian religious Order called the Society of Jesus. Jesuit education is inspired by the life and teachings of Jesus Christ and on the principles of pedagogy elaborated by St. Ignatius of Loyola, the founder of the Society of Jesus.

The College aims to offer an all-round formation that is intellectual, cultural, social, emotional, physical, aesthetic, moral and spiritual. It further aims at promoting values such as respect for common Indian cultural heritage, egalitarianism, democracy, secularism, equality of sexes, protection of environment, removal of social barriers, responsible use of cybernetics and mass media, transparency and probity in private and public life, national unity and respect for religious and moral values.

The College Emblem contains the motto of the College: Pravahito Gyanganga Pravah: - Let the streams of Gyan Ganga keep on flowing. The College is situated near the river Ganga. Like the flowing river, the College is to ensure that the streams of Gyan keep on flowing and liberating people. The Sun with the letters IHS (first three letters for JESUS in Greek) is a symbol characterizing the Society of Jesus. The emblem within the emblem, containing crown, crescent moon, etc. is the coat of arms of the noble family of our patron, St. Francis Xavier.

ABOUT UNIVERSITY



The Aryabhatta Knowledge University Act 2008 provides for the establishment of university at Patna to conduct and facilitate affiliation of institutions in the conventional as well as frontiers of professional education. All colleges and institutions imparting professional education for example Engineering and Technology including Information Technology, Nano technology & Biotechnology, Management, Medicines, Health Technology, Public Health Pharmacy, Optometry, Nursing, Education, Law etc. is to be affiliated to this university.

Aryabhatta Knowledge University (AKU), Patna has been established by Government of Bihar for the Development and Management of Educational Infrastructure related to Technical, Medical, Management and allied professional education in the state. The objective of the university is to promote the professional education infrastructure to meet the national standard through well advanced Course, infrastructure and quality faculty. Bihar lacks in the infrastructure both on the term of number of intuitions and the quality of education. As a result Bihar State has become the major hub of student migrating to the other states for pursuing their education and carrier building for future growth.

The vision of this University is to mould the character, shape the career, and bring perfection in behavior and excellence in educating the young generation of today for future. Also, to bring up a vibrant knowledge university resonating with the mission of all round development of students in particular and the national and mankind in general by providing value-based creative, innovative quality education.

INTRODUCTION

1.1 About

The "Attendance Management System" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner. The application is reduced as much as possible to avoid errors while entering the data. It also provides an error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user-friendly. Attendance Management System, as described above, can lead to error free, secure, reliable and fast management systems. It can assist the user to concentrate on their other activities rather than concentrating on the record keeping. Thus, it will help organizations in better utilization of resources.

1.2 Purpose

The purpose is to design software for an Attendance database which contains up to date or accurate information of the Attendance. That should improve efficiency and flexibility of Attendance record management and to provide a common and or simple platform for everyone to access the student's information. Attendance Automation System consists of different modules such as student, faculty, admin etc. Our main purpose is to create a software which will manage the working of these different modules. The interconnectivity among modules reduces the time to perform different operational tasks.

1.3 Scope

The AMS will permit process impact gathering information about students from the admission cell and give it to Teacher. And another very important impact is taking classes, conducting exams, maintaining attendance records, fee details, assignments, marks etc.

SYSTEM ANALYSIS

2.1 INTRODUCTION

Analysis can be defined as breaking up of any whole so as to find out their nature, function etc. It defines design as to make preliminary sketches of; to sketch a pattern or outline for plan. To plan and carry out especially by artistic arrangement or in a skillful way. System analysis and design can be characterized as a set of techniques and processes, a community of interests, a culture and an intellectual orientation.

The various tasks in the system analysis include the following: -

- Understanding application.
- Planning.
- Scheduling.
- Developing candidate solution.
- Performing trade studies.
- Performing cost benefit analysis.
- Recommending alternative solutions.
- Selling of the system.
- Supervising, installing and maintaining the system

This system manages to the analysis of the report creation and develops manual entry of the student attendance. First design the student's entry form, staff allocation and time table allocation forms. This project will help the attendance system for the department calculate percentage and reports for eligibility criteria of examination. The application attendance entry system will provide flexible report for all students.

2.2 EXISTING SYSTEM

The Existing system is a manual entry for the students. Here the attendance will be carried out in the hand written registers. It will be a tedious job to maintain the record for the user. The human effort is more here. The retrieval of the information is not as easy as the records are maintained in the hand written registers. This application requires correct feed on input into the respective field. Suppose the wrong inputs are entered, the application resist to work. so the user finds it difficult to use.

2.3 PROPOSED SYSTEM

To overcome the drawbacks of the existing system, the proposed system has been evolved. This project aims to reduce the paper work and saving time to generate accurate results from the student's attendance. The system provides with the best user interface. The efficient reports can be generated by using this proposed system.

Advantages of Proposed System

- It is trouble-free to use.
- It is a relatively fast approach to enter attendance
- Is highly reliable, approximate result from user
- Best user Interface
- Efficient reports

2.4 FEASIBILITY STUDY

Feasibility analysis begins once the goals are defined. It starts by generating broad possible solutions, which are possible to give an indication of what the new system should look like. This is where creativity and imagination are used. Analysts must think up new ways of doing things- generate new ideas. There is no need to go into the detailed system operation yet. The solution should provide enough information to make reasonable estimates about project cost and give users an indication of how the new system will fit into the organization. It is important not to exert considerable effort at this stage only to find out that the project is not worthwhile or that there is a need significantly change the original goal.

Feasibility of a new system means ensuring that the new system, which we are going to implement, is efficient and affordable. There are various types of feasibility to be determined. They are:

2.4.1 Economically Feasibility

Development of this application is highly economically feasible. The only thing to be done is making an environment with an effective supervision. It is cost effective in the sense that has eliminated the paper work completely. The system is also time effective because the calculations are automated which are made at the end of the month or as per the user requirement.

2.4.2 Operational Feasibility:

The system working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system. Technical performance includes issues such as determining whether the system can provide the right information for the Department personnel student details, and whether the system can be organized so that it always delivers this information at the right place and on time using intranet services. Acceptance revolves around the current system and its personnel.

2.4.5 Technical Feasibility:

The technical requirement for the system is economic and it does not use any other additional Hardware and software. Technical evaluation must also assess whether the existing systems can be upgraded to use the new technology and whether the organization has the expertise to use it.

PROJECT DESCRIPTION

3.1 Perspective

The AMS is a system that can help maintain students and Teacher information easily and efficiently. A fully integrated web-based ERP will capture and create accurate, consistent and timely relevant data, and assist in intelligent business decision-making. The primary purpose of AMS is to provide mechanisms for automated processing and management of the entire institution. It reduces data error/ data redundancies, ensures that information is managed efficiently and is always up-to-date. Complete student histories for all years, can easily be searched, viewed and reported on the press of a button.

The AMS is a standalone product and doesn't depend upon the availability of other websites. The system will have an administrator who has full-fledged rights with regards to performing all actions related to control and management of the website and all relevant data.

3.2 Technologies Used for development

We have used MERN stack for our full stack project. MERN Stack is a collection of powerful technologies and robust, used to develop scalable master web applications comprising **backend**, **front-end**, and **database components**. It is JavaScript that is used for the faster and easier development of full-stack web applications. MERN Stack is a technology that is a user-friendly full-stack JavaScript framework for building applications and dynamic websites.

MERN Stack consists of four main components or can say four main technologies:

1. **M** stands for **MongoDB (Database)**, mainly used for preparing document database and is a NoSQL (Non-Structured Query Language) Database System
2. **E** stands for **Express**, mainly used for developing Node.js web framework
3. **R** stands for **React**, mainly used for developing a client-side JavaScript framework
4. **N** stands for **js**, mainly used for developing the premier JavaScript web server

Each of these four technologies plays an important role in providing an end-to-end framework for the developers. Even these four technologies play an important role in the development process of web applications.

Before MERN stack, it was earlier named as MEAN stack, MERN Stack is one of the variations of MEAN, here MEAN is also comprises of four components or say four different technologies, i.e., M is for MongoDB, ' E ' is for Express, ' A ' is for Angular.js, and ' N ' is for Node, here if you will observe, then you can identify that in MEAN, ' A ', i.e., Angular.js is replaced by ' R ', i.e., React.js in MERN, the main reason behind is - MERN Stack is mainly used for faster development of smaller applications as compared with MEAN, MEAN stack is a mainly better option for large-scale applications. Still, it takes more time for the development of smaller applications. They also both have different structures comparatively.

3.2.1 Why we chose MERN Stack for building our Project

1. **Cost-effective:** All the four technologies that are mentioned above, MERN (MongoDB, Express.js, React.js, and Node.js) are used in MERN Stack is built on JavaScript that makes it cost-effective and within less cost investment user will able to get the better results or output.
2. **SEO friendly:** Here, **SEO (Search Engine Optimization)** friendly means that Google, Yahoo and other search engines can search each page on the website efficiently and easily, interpret and correlate the content effectively with the searched text and easily index it in their database. As whenever websites are created using MERN technologies, then it is always SEO friendly.
3. **Better performance:** Better performance refers to the faster response between backend and front-end and database, which ultimately improves the website speed and yields better performance, thus providing a smooth user experience.
4. **Improves Security:** It mainly concerns the security of applications generated using MERN; her web application security refers to various processes, methods or technologies used for protecting web servers and various web applications, such as APIs (**Application user interface**) from the attack by internet-based threats. Generally, secured hosting providers can easily integrate applications created using the MERN stack. For more or better security Mongo DB and Node.js security tools are also used.

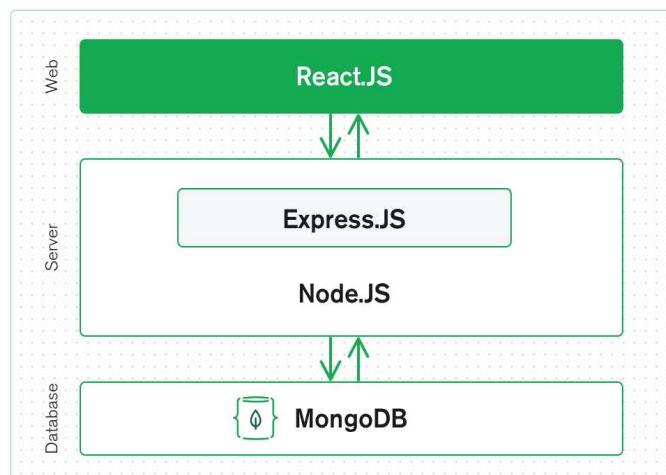
5. **Provide the fastest delivery:** Any Web applications and mobile applications created by using MERN Stack are built much faster, which also helps to provide faster delivery to our clients.
6. **Provides faster Modifications:** MERN stack technologies supports quick modifications as per the client's request in the mobile and web applications.
7. **Open Source:** All the four technologies that are involved in MERN are open-source. This feature allows developers to get solutions to queries that may evolve from the open portals during development. As a result, it will be ultimately beneficial for a developer.
8. **Easy to switch between client and server:** MERN is very simple and fast because it is written in only one language. And also, it is very easy to switch between client and server.

3.2.2 Architecture of MERN stack and its Working

MERN has a 3-tier Architecture system mainly consisting of 3 layers -

These layers are as follows:

1. Web as front-end tier
2. Server as the middle tier
3. Database as backend tier

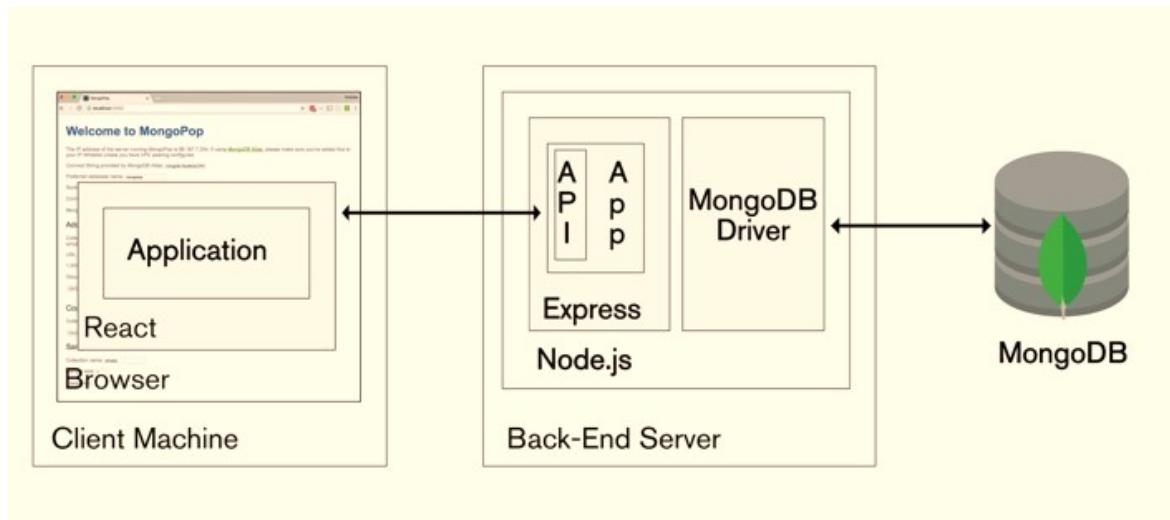


Web or front-end tier - The top tier of the MERN stack is mainly handled by React.js. It is one of the most prominent used open-source front-end JavaScript libraries used for building Web applications. It is famous for creating **dynamic client-side applications**. React will help you construct complex interfaces by using single components. It also connects those complex interfaces to data available on the backend server. React is used to create mobile applications (React Native) and web applications. React allows the reusability of code and can easily support it, which has many benefits and is much time saver. It permits users to create large web applications that can easily change the data of the page even without reloading the page.

Server or middle-tier - It is just next level from the top layer and is mainly handled by two components of the MERN stack, i.e., **Express.js** and **Node.js**. These two's components handle it simultaneously because Express.js maintained the Server-side framework, running inside the Node.js server. Express.js is one of the widely used backend development JavaScript Frameworks. It allows developers to spin up robust APIs (Application Programming Interface) and web servers much easier and simpler. It also adds helpful functionalities to Node.js HTTP (**HyperText Transfer Protocol**) objects. Whereas on the other hand, Node.js plays a very important role in itself. It is an open-source server environment, and it is a cross-platform runtime environment for executing JavaScript code outside a browser. Node.js continuously uses JavaScript; thus, it's ultimately helpful for a computer user to quickly create any net service or any net or mobile application.

Database as backend tier - It is one of the most important levels of the MERN Stack and is mainly handled by MongoDB; the main role of a database is to store all the data related to your application, for example - **content, statistics, information, user profiles, comments** and so on. It mainly stores all the data for **safety purposes**. It maintains a proper record, which usually returns the data to the user whenever required. It mainly stores the data in the database. It generates two or more replica files of the data so that whenever the system fails, it can retrieve the exact information or data that the user wanted earlier. It implies that MongoDB is not based on the table-like relational database structure. On the other hand, it provides an altogether different mechanism for the retrieval and storage of data. **Mongo DB** is the most popular NoSQL (NoSQL or Non Structured Query Language) database, an open-source document-oriented database. The term 'NoSQL' typically means a non-relational database that does not require a fixed schema or proper relational tables to

store the necessary data in it. MongoDB stores the data in a different format other than the relational tables, consisting of rows and columns.



3.2.3 Major Components of MERN Stack

i. MongoDB

- **Mongo DB** is the most popular NoSQL (NoSQL or Non Structured Query Language) database, an open-source document-oriented database.
- The term 'NoSQL' typically means a non-relational database that does not require a fixed schema or proper relational tables to store the necessary data in it. MongoDB stores the data in a different format other than the relational tables, consisting of rows and columns.
- It implies that MongoDB is not based on the table-like relational database structure. On the other hand, it provides an altogether different mechanism for the retrieval and storage of data.
- The storage format in which the data is stored is known as BSON, which stands for Binary JavaScript Object Notation; its binary structure encodes length and type of information, which allows it to be parsed much more quickly.
- MongoDB uses BSON when storing documents in collections.
- It allows a highly scalable and flexible document structure.
- It is very faster as compared to RDBMS due to its efficient storage and indexing techniques.
- In MongoDB, complex join operations are not available; hence, it cannot support complex transactions.
- MongoDB uses JavaScript for coding as a language which is one of the great advantages.
- It is Schemaless as any data stored which is stored in a separate document.

- In MongoDB, there is no concept of relationships or table formations, as this is happening in RDBMS (Relational Database Management System), in which tables have a certain relation between them.
- It also supports a flexible document model, which is very fast for any developer to create.
- MongoDB is one of the important types of NoSQL Databases. It is more scalable and provides excellent performance if we notice that it will reach its scaling limit whenever a database runs on a single server.
- MongoDB is a NoSQL database that scales by adding more and more servers and increases productivity with its flexible document model.

Some important features of MongoDB -

- **Schema-less Database:** MongoDB has this one of the great features, which means that one collection can hold different types of documents in it. Due to this extraordinary feature, MongoDB provides great flexibility to databases. In the MongoDB database, a single collection comprises multiple documents, and these documents may further comprise the different numbers of values, fields, and so on. One document doesn't need to be a must to relate with the other documents, as it happens in relational databases.
- **Indexing:** In the MongoDB database, one can easily fetch out the necessary data from the data pool due to this indexing feature. In MongoDB, every data item has provided a particular index, categorized as primary and secondary indices. With this indexing, data retrieval is easier for the user; it saves a lot of time. If the data is not indexed, the database searches each document with the specified query, which takes lots of time and is inefficient.
- **Document Oriented:** In MongoDB, all the data has been stored in documents instead of tables like SQL. Also, these documents have their unique object id. In these documents, the informative data is stored in fields, i.e., **key-value** pairs instead of columns and rows, making the data much more flexible and easier to fetch out rather than applying queries for every data compared to RDBMS.
- **Faster** - MongoDB is very fast compared with relational database (**RDBMS**), which is document-oriented. Each data item has its index value, making it easier for us to retrieve any data without wasting time writing queries and making logic accordingly.
- **Scalability:** MongoDB is more scalable with the help of sharding. It provides horizontal scalability. Here the term sharding means distributing data on multiple

servers; in this, a large amount of data has been divided into multiple small data chunks with the help of shard key. These types of **data chunks** are evenly distributed across shards that reside across many physical servers.

- **High Performance:** MongoDB has very high performance and has data persistency as compared to other databases due to the presence of its great features like **indexing, scalability, replication**, etc.
- **Replication and Highly Available** - MongoDB increases the availability of data due to creating multiple copies of data on different servers. Providing redundancy or data replication ultimately protects the database from any hardware failure and protects the data from being lost in the future. I suppose if one server was not working or clashes due to error, and then data can easily be retrieved from other active servers, who are currently working at that time, this will all be due to redundancy of data.
- **Aggregation:** This feature of MongoDB is quite similar to the SQL GROUPBY clause. This GROUPBY clause performs various operations on the grouped data to get the unique or computed
- **Simple Environment Setup** - MongoDB has a very simple environment setup. One can easily set up MongoDB in their system without applying much effort.

ii. Express.js

- Express is a JavaScript server-side framework that runs within **js**.
- It is one of the best backend development JavaScript Frameworks.
- It provides the developer with a platform to create and maintain robust servers.
- Express is used for building and designing web and mobile applications easily and quickly.
- Express is used to provide server-side logic for mobile and web applications, and as such, it is used all over the place.
- It allows developers to spin up robust APIs (**Application Programming Interface**) and web servers much easier and simpler.
- Express makes robust web servers easier to organize your application's functionality with routing and middleware.
- It also adds helpful functionalities to Node.js HTTP (**HyperText Transfer Protocol**) objects.

- It is an important component of the MERN and MEAN Stack and is used to build fast, maintainable, and robust production web applications.

Some important features of Express -

- Express makes Node.js web and mobile application development much easier and faster.
- Express has a very simple environment setup. One can easily set up Express in their system and configure it without applying much effort.
- Express is very easy to connect with Databases like MongoDB.
- Based on HTTP methods and URLs, Express allows you to define the routes of your application.
- Routing mainly aims to describe code that needs to be run in response to any request received by a server. Routing is generally done based on the sequence of URL patterns and the HTTP method, which is associated with the request.
- If you want to perform additional tasks and functions on any request and response, you can easily use various middleware modules present in Express.
- The request is a message that arrives at the server for requesting something, and a Response is a message sent by the server to a client in the form of the result of whatever the client asked for.
- If any error occurs and you want to handle it, you can easily handle it by using error handling middleware.
- **Middleware** is used somewhere during the lifecycle of request or response in the form of code. It is mainly used to add functionalities or augment the behaviour of the webserver.
- Express also facilitates you to create a **REST API** (Representational State Transfer Application Programming Interface)
- The REST APIs is also known as RESTful API, It mainly conforms to the constraints of REST architectural style, and it also allows for interaction with RESTful web services. The main advantage of REST API is that it provides great flexibility; it uses HTTP requests to access and use data.
- The data flow into a website structure can easily facilitate by using the two template engines, EJS and Jade, provided by Express.

- Express has a gigantic suite of third-party add-ons so that developers can use it to provide better functionality, helps to increase the security level, and improve speed.
- It is very efficient and scalable; one can easily access it from anywhere and use it simultaneously on different systems, and very fast.
- It is Single-threaded and Asynchronous.
- It also has the biggest community for Node.js.
- With its built-in router, it promotes code reusability.
- If we want to understand the architecture behind web servers and their working along with the organization, then learning Express is the best option.

iii. React.js

- React is one of the most popular open-source front-end JavaScript libraries used for building Web applications.
- Before using react, it has some prerequisites that one should follow, that you must download Node packages in your system with their latest versions. Also, you must have an understanding of HTML, CSS and JavaScript.
- It is used to build user interfaces, especially for a single page web application.
- It is not a JavaScript framework. It is just a JavaScript library developed by Facebook to solve problems we could not solve earlier using other libraries while building web and mobile applications.
- React is also used for making a grip over the view layer for mobile and web applications.
- It allows us to create reusable UI (User Interface) components.
- It was first created by software engineer Jordan Walke, who works for Facebook.
- React was first deployed in the Facebook news feed.
- It allows developers to create large web applications that can easily change the data of the page even without reloading the page.
- The main objective of reacting is that it only works on user interfaces in the application, whether mobile or web.
- It is very fast, simple and scalable.
- React is also used with a combination of other JavaScript libraries or frameworks.
- There are a lot of open-source platforms that are also used to make the front-end web and mobile applications easier, like Angular js in MVC, but still, React replaces the

Angular from the MEAN stack. Now, most developers are using the MERN stack in which react is used; the main reason is that it is very fast and has more advantages over other front-end frameworks.

Some Important features of React -

- **Easy to learn** - One of the great advantages of using react as it is very easier for a beginner to learn it and make web and mobile applications using this front-end framework. Anyone with a piece of previous basic knowledge in programming can easily understand React compared to Angular. Angular is referred to as a ' Domain Specific Language ', so it is implied that it is quite difficult to understand it. For Learning React, you need the basic knowledge of CSS and HTML.
- **Simple** - React is one of the simplest open-source JavaScript front-end frameworks for building web and mobile applications. It uses the component-based approach, uses plain and simple JavaScript, and a well-defined lifecycle, which makes react much simpler and easier. So that one can easily learn it and build professional mobile and web applications. It uses a simple syntax named JSX, which allows learners or developers to mix HTML with JavaScript to make it easier for them to apply and use it for making efficient web and mobile applications. However, it is not required to use JSX, you can either use plain JavaScript, but as compared to JSX, JSX is the much better option over it due to its simplicity and easier syntax.
- **Data Binding** - React uses an application architecture known as Flux to control data flow to components via one control point called the dispatcher. It uses **one-way data binding**, which is easier to debug self-contained components of large React applications.
- **Native Approach** - React is used to create mobile applications (React Native) and web applications. React allows the reusability of code and can easily support it, which has many benefits and is much time saver. So simultaneously, at the same time, we can make **IOS, Web applications** and Android.
- **Performance** - React has very fast performance due to the immutability of data. As the name suggests, we can predict that the immutable data structures never change and allows you to compare direct object references instead of doing deep-tree comparisons. The above reason ultimately affects the performance of reacting and makes it faster.

- **Testability** - React is very easy to test; whatever applications we are generating from the react, whether mobile or web applications, it is much easier for us to test it on react. There are some state functions in the react, where various react views are treated as these functions of the states, and we can easily manipulate with the state we pass to the react view. Also, we can take a look at the output and triggered actions, functions, events, etc.

iv. Node.js

- Node.js is an open-source server environment, and it is a cross-platform runtime environment for executing JavaScript code outside a browser.
- Node.js is not a programming language, and even it is not a framework.
- It is often used for building and developing numerous backend services like net applications, mobile applications.
- Massive corporations principally utilize it in production like **Uber, PayPal, Netflix**,
- It may be a free ASCII text file platform and may be utilized by anybody.
- It will run on numerous operative systems like **Windows, Mac, Linux, Unix**, etc.
- It is incredibly simple to start with it and may even be used for agile development and prototyping.
- It provides extremely scalable and really quick services to the users.
- It is incredibly consistent and may be used as an ASCII text file cleaner.
- It continuously uses JavaScript; thus, it's ultimately helpful for a computer user to quickly create any net service or any net or mobile application.
- It provides a massive system for any ASCII text file library.
- It contains a Non-blocking or, can say, Asynchronous nature.

Some Important features of Node.js –

- **Easy Scalability:** js is highly scalable because it uses a single-threaded model with event looping. The server usually responds in a non-blocking way due to the help of the event mechanism. It also makes the server very scalable instead of traditional servers that create limited threads to handle requests. Node.js uses a single-threaded program, and this program will be able to provide service to many requests.

- **Fast:** The event loop in Node.js handles all asynchronous operations, so Node.js acts like a fast suite, and all the operations in Node.js are performed quickly like network connection, reading or writing in the database, or file system. It runs on the V8 engine developed by Google.
- **Easy to learn and debug code:** js is quite easy to learn and debug because it uses JavaScript for running code of web-based projects and various web and mobile applications. If you have excelled in front-end developing and have a good command of JavaScript, you can easily build and run the application on Node.js and explore more as much you can; it depends on your capability.
- **Real-time web apps:** js plays a key role in making real-time web applications. And If you are building a mobile or a web application, you can also use PHP, although it will take the same time duration as when you use Node.js. Still, if someone wants to build gaming apps and chat applications, then Node.js is a much better option because of its faster synchronization.
- **Caching Advantage:** js provides the caching property in which a single module is cached. Sometimes you do not need to re-execute the same lines of code because it has already been cached using Node.js.
- **Data Streaming:** In Node.js, hypertext transfer protocol (HTTP) requests and responses area unit thought-about as 2 separate events. They're knowledge streams, thus once you method a file at the time of loading, it'll scale back the time and create it quicker once the info is given within the style of transmissions. It additionally permits you to stream audio and video files at lightning speed.
- **Object-Oriented Approach:** A huge complaint against Node.js was its JavaScript heritage, which frequently involved many procedural spaghetti codes. Frameworks like Coffee Script and Typescript solved these issues but came as a bolt-on for those who seriously cared about coding standards. With the release and general adoption of ES6, Classes are built into the framework, and the code looks syntactically similar to C#, Java and SWIFT.
- **Event-Driven and Asynchronization-** All APIs of the Node.js library area unit asynchronous, that is, non-blocking. It suggests that a Node.js based mostly server ne'er waits for associate API to come back knowledge. The server moves to the consequent API once line it, and a notification mechanism of Events of Node.js helps the server to urge a response from the previous API decision.

- **Corporate Support:** There are a lot of famous companies like PayPal, Wal-Mart, Microsoft, Google that are using Node.js for building the applications. Node.js uses JavaScript, so most companies are combining front-end and backend Teams into a single unit.

3.3 Technologies used for deployment

3.3.1 GitHub

GitHub plays a role in hosting websites through its GitHub Pages feature. GitHub Pages is a static site hosting service that takes HTML, CSS, and JavaScript files straight from a repository on GitHub and publishes a website. This allows users to host their static websites directly from their GitHub repository. It provides an easy and convenient way for users to create and host their websites for free. We have used GitHub for storing our repository so that we can access it on other websites like Cyclic.sh to host our websites.

- It is a free hosting service provided by GitHub that allows you to host your static websites directly from your GitHub repository.
- GitHub provides the flexibility to make and reflect changes on the website in real-time which reduces the overhead of uploading of same source code with minor changes again and again.
- You can create a website by simply publishing static HTML, JavaScript, and CSS files from GitHub's repository.
- It provides unlimited storage access.
- However, it is important to note that GitHub Pages is not intended for or allowed to be used as a free web hosting service to run your online business, e-commerce site, or any other website that is primarily directed at either facilitating commercial transactions or providing commercial software as a service (SaaS).

3.3.2 Cyclic

Cyclic.sh is a platform for deploying and hosting full-stack JavaScript applications. It allows you to connect your GitHub repository and build, deploy, and manage the hosting of your application. We used cyclic to host our backend of project online.

- Some of its features include a simple key-value inspired SDK, write-time indexing, flexible queries, JSON schema discovery, and local development. We have used this platform for hosting our web application
- It also offers a free tier with features such as no inactivity timeout/delay, unlimited hourly limit, 1024 MB RAM compute with autoscaling up to 10 x 1024 MB, 100,000 API requests per month per app, support for JavaScript, TypeScript and Python (beta), an integrated database with 1 GB storage and integrated object storage
- Cyclic apps are serverless which gives them the ability to scale up or down quickly.

3.3.3 Vercel

Vercel is a platform for frontend developers that provides fully-managed infrastructure, automated CI/CD, and integrated collaboration. We use Vercel to host our frontend of project online.

- It supports any type of application, from traditional HTML/CSS/JS sites to server-side rendered React or Next.js applications, as well as GraphQL-powered backends.
- Some of its features include Git provider integrations, preview deployments, edge functions, serverless functions, edge middleware, static file caching, web analytics and speed insights.
- The platform also provides powerful DevOps features like automatic scaling, deploy previews and canary deployments.

3.3.4 MongoDB Atlas

MongoDB Atlas is a fully managed cloud database service that offers robust data management paired with the leading NoSQL database. We used MongoDB Atlas for hosting our database online.

- It supports multiple cloud providers including AWS, Azure, and Google Cloud Platform.
- The free version of MongoDB Atlas allows you to deploy a free cluster (M0) with some limitations. For example, M0 free clusters limit the total data transferred into or out of the cluster to 10 GB in and 10 GB out per rolling seven-day period.

- MongoDB Atlas simplifies deploying and managing your databases while offering the versatility you need to build resilient and performant global applications on the cloud providers of your choice.

3.4 Product Features and User Classifications

There are several types of end users for the AMS. They are broadly divided as Students, Teacher and the Administrator. Each of these classes have their own set of features

- **ADMIN** who can view and edit the details of any students/Teacher. Can add/edit departments, courses, classes and time-tables.
- **TEACHER** who can view students' details, add/update assignments, marks and attendance of a particular student. They can see the time-table of a particular class also.
- **STUDENT** who can update profile/ add solution to assignments and see marks/attendance.

The features that are available to administrator are:

- An administrator can login into the system and perform any of the available operations:
 - add a student
 - view the student details
 - modify student data
 - delete student record
 - add Teacher
 - view teacher details
 - modify teacher data
 - delete teacher record
 - add department
 - modify department
 - add/modify courses
 - add/modify classes
 - Can view the record of all the students and teacher of a particular class/course/department.

The features that are available to Teachers are:

- A Teacher can login into the system and perform any of the available operations:
 - view his/her personal details (name, dob, age, address, joining date)
 - edit his/her details
 - view the department (s)he belongs to.
 - the classes they teach.
 - generate class report
 - mark attendance of a particular student
 - update the attendance
 - see the time-table of a class.

The features that are available to students are:

- A student can login into the system and perform any of the following operations:
 - view his/her personal details (name, dob, age, address, joining date)
 - edit his/her details
 - view the department (s)he belongs to.
 - the classes they belong
 - view his/her attendance
 - add solution to the assignment
 - view his/her marks

3.5 Operating Environment

The operating env. for AMS application are listed below: -

- Operating System: Windows 11
- Database: MongoDB
- Frontend: HTML/CSS/JS/Bootstrap/React.js
- Backend: Node.js
- Deployment: GitHub

3.6 Hardware Requirements

The Hardware Req. for AMS application is Listed below: -

- OS: - Windows 7 (min)
- Processor: - i3 5th Gen (min)
- RAM: - 2GB. (min)
- HDD: - 80GB (min)
- Browser: - Chrome, Safari, Linux.

3.7 Constraints

- Users should be aware of the internet.
- Users must have a valid id and password.
- Only Admin, Teacher and Students can access.
- Every user must be comfortable using a computer.
- All operations are in English so users must have basic knowledge of English.

INTERFACE REQUIREMENT

4.1 User Interfaces

The User Interfaces are made using HTML / CSS /JS and React.js. Each type of end user has different types of UI as per requirements. These are user friendly and anyone can use it. No extra efforts or manual work is needed for marking attendance, assigning assignments, submitting assignments etc.

4.2 Hardware Interfaces

Since neither the mobile application nor the web portal have any designated hardware, it does not have any direct hardware interfaces. Any browser can be used to access the web app. Though some features of JavaScript may not work in old versions of browsers such as internet explorer, so it is best recommended to use this on the latest browsers.

4.3 Software Interfaces

The following is a list of software used in the making of the project.

- Operating System: We have chosen Windows operating system for its best support and user-friendliness.
- Backend: We have made our backend using node.js as it offers an easy scalability (both in horizontal as well as vertical directions.)
- Database: Our main motive is to use schema-less document oriented databases and so we are using MongoDB as a database for our project. It provides Data Security, On Demand Scalability and comes with a very user-friendly MongoDB Compass application which can be used to see our data more easily through GUI and run our script

SYSTEM DESIGN

5.1 System Design of AMS

Various Design concepts and processes were applied to this project on the basis of which a complete logical system is built which fulfils the given requirements. There were two steps while design analysis:

1. Primary Design Phase:

The system is designed at block level. We first decided the end users that will be using our application. On finalizing that and following the concepts like separation of concerns, the software is divided into individual modules based upon the end users that are functionally independent and incorporates information hiding. There are 3 types of end users (modules) - students, Teacher and the administrator.

2. Secondary Design Phase:

Each module/block was studied in detail during this phase.

5.2 Admin

The administrator will have access to all the information in the different tables in the database. They will be able to add an entry in any table and also edit them. They will be responsible for maintaining the data. The main aim here is to provide a user-friendly environment for the admin so that they can easily adapt themselves with the environment. They will be responsible for the creation/deletion of student / Teacher / department / course / class information. Only authorized users will be made admin and thus we have protected the routes. They will be provided with search and filter features so that they can access the data efficiently. Different Views of Admin are described below.

- **Dashboard**

An overview for admin where they can easily navigate to other submodules for performing its task.

- **Profile**

Admin will be able to see its own information. They will be able to see a complete overview of total students / Teacher in the Attendance and various courses / departments.

- **Settings**

Admin will be able to change it; s own information like name, email, password, add other information such as address, contact etc.

- **Department**

The admin is responsible for creating new departments and updating the already existing ones. Here they will be provided with search and filter operations so that they can work efficiently. Necessary information will be needed while adding new departments.

- **Course**

The admin is responsible for creating new courses and updating the already existing ones. Here they will be provided with search and filter operations so that they can work efficiently. Necessary information will be needed while adding new courses (like which department the course is associated with).

- **Class**

The admin is responsible for creating new classes and updating the already existing ones. Here they will be provided with search and filter operations so that they can work efficiently. Necessary information will be needed while adding new classes.

- **Student Section**

The admin will be able to see the complete information of all the students. They are also responsible for adding new students and updating the already existing ones. They will be provided with a user-friendly UI to list down all the necessary information of students and if they want, they can see the complete details by clicking on the relevant student's name.

- **Teacher Section**

The admin will be able to see the complete information of all the Teacher. They are also responsible for adding new students and updating the already existing ones. They will be provided with a user-friendly UI to list down all the necessary information of Teacher

and if they want, they can see the complete details by clicking on the relevant Teacher's name.

- **Forget Password / Reset Password**

In case an admin forgets his password or wants to reset it, he can do so by clicking on a reset password button. A reset link will be sent in the mail which will be active for only 20 min.

5.3 Teacher

Each teacher belongs to a department and is assigned to classes with a course. Teachers will also have a username and password to login (which will be provided by the admin). The different views for teachers are described below.

- **Dashboard**

An overview for Teacher where they can easily navigate to other submodules for performing tasks.

- **Profile**

Each Teacher member will be able to see its own information. They will be able to see a complete overview of the classes they teach, number of students that belong to those classes etc.

- **Class**

Teacher can see a list of all the classes he teaches. He can navigate into any class to see the list of students of that class.

- **Attendance**

The teacher has the ability to add and also edit the attendance of each student. For entering the attendance, they will be given the list of students in each class and they can enter the attendance of the whole class on a day-to-day basis.

- **Time Table**

The teacher can see the time-table of the classes he teaches (so that if he wants to take an extra class he can do so without any clashes).

- **Forget Password / Reset Password**

In case a teacher forgets his password or wants to reset it, he can do so by clicking on a reset password button. A reset link will be sent in the mail which will be active for only 20 min

5.4 Students

Each student belongs to a class identified by semester and section. Each class belongs to a department and are assigned a set of courses. Therefore, these courses are common to all students of that class. The students are given a unique username and password to login. Each of them will have a different view. These views are described below.

- **Dashboard**

An overview for students where they can easily navigate to other submodules for performing tasks.

- **Profile**

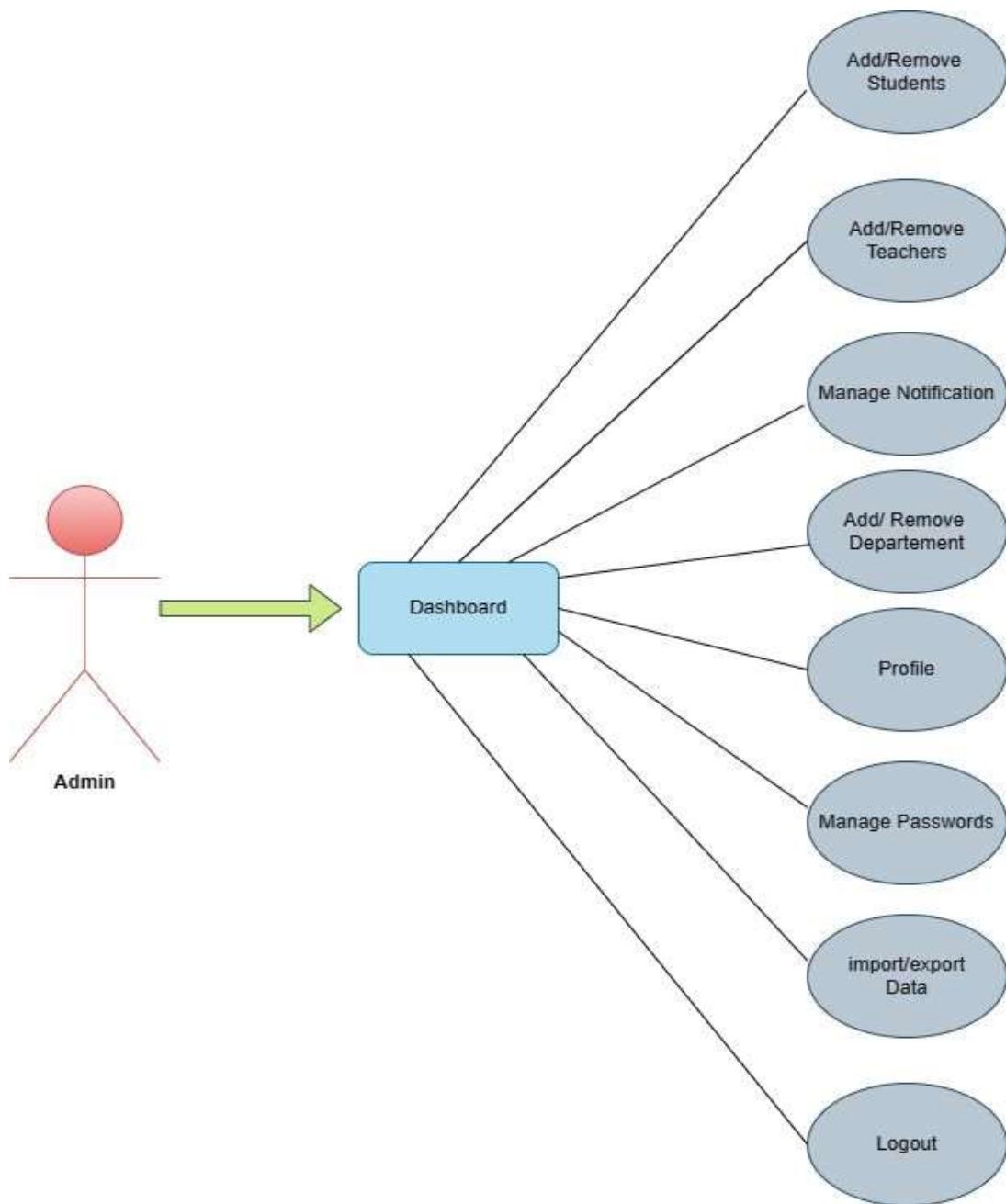
Each student will be able to see its own information. They will be able to see the department they belong to, current semester registered courses, their personal information etc.

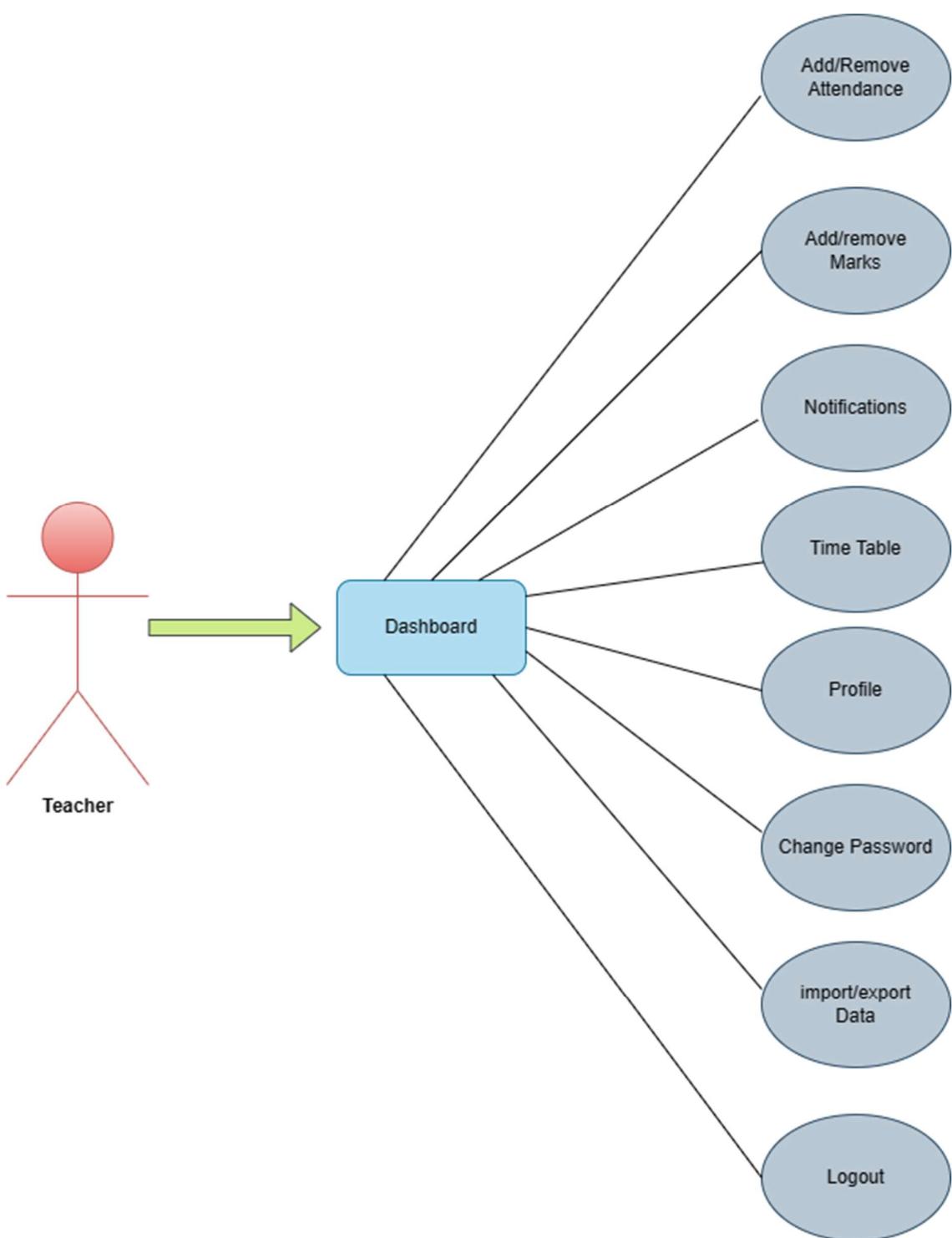
- **Attendance Information**

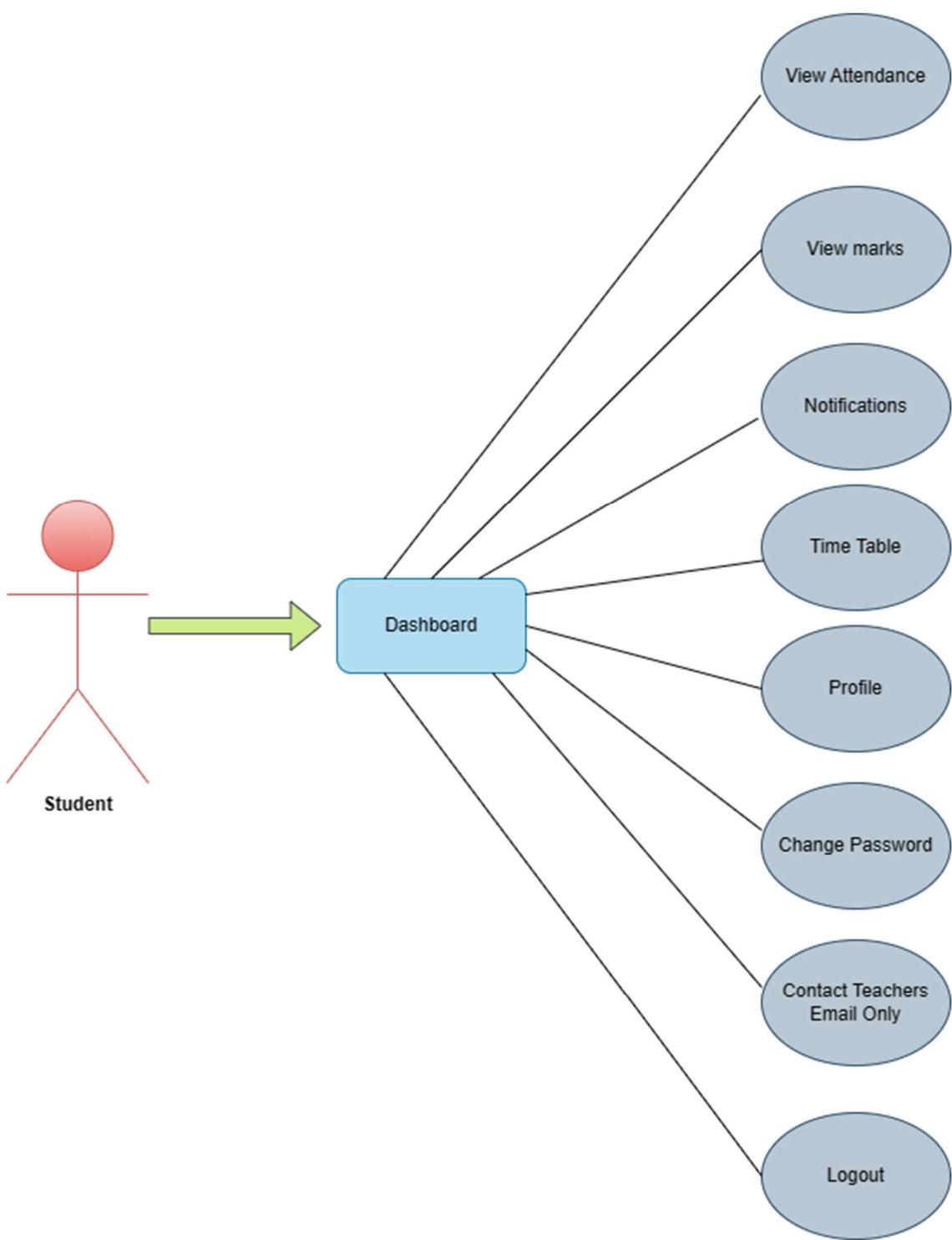
Attendance for each course will be displayed. This includes the number of attended classes and the attendance percentage. If the attendance percentage is below a specified threshold, say 75%, It will be marked in red otherwise it will be in green.

5.5 End User Features (Diagram)

This is the use case diagram which depicts the user's interaction with the system. It also shows the relationship between the user and the different use cases in which the user is involved.







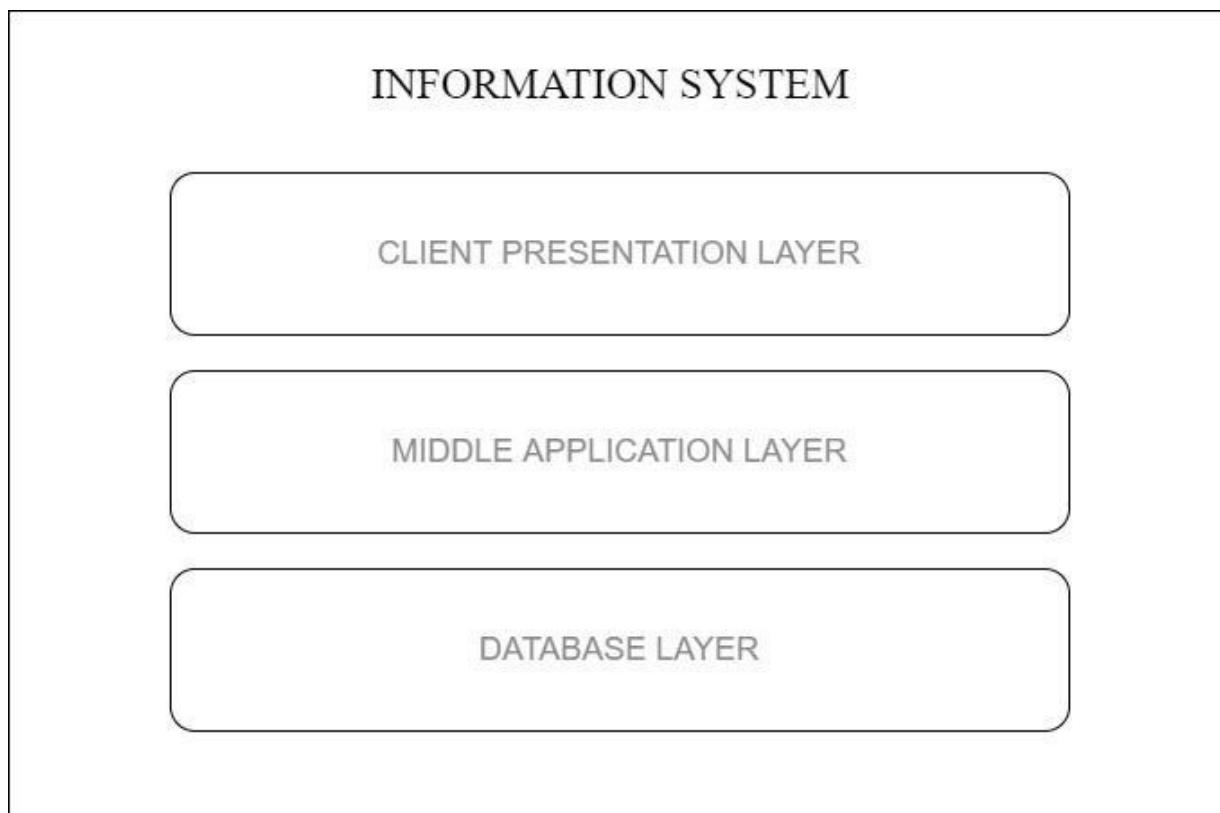
5.6 3 tier Architecture

Generalizing AMS architecture in 3 tiers. The 3 tiers consist of **presentation layer**, **application logic layer and data layer**. Any Information System needs to communicate with external entities, human users or other computers. Presentation layer allows these entities to interact with the system; it can also be implemented as a GUI interface and can be referred to as the client of the IS. Application layers do more than information delivery, they perform data processing (Business Logic and calculation) behind the results being delivered.

This tier is often referred to as

1. Services
2. Business rules
3. Business logic
4. Servers

The database layer is implemented using a Database Management System which in our case is MongoDB.



5.7 Database Design and Implementation

We are using MongoDB as our database. The main objective of this project is to use a NoSQL document-oriented database and hence MongoDB is the best choice for that.

5.7.1 Key Features of MongoDB

- All data are represented in JavaScript Object Notation (JSON) format and hence called Data Model
- Availability of Cloud database service through MongoDB Atlas
- It is suitable for hierarchical data storage
- In terms of performance, it is much faster than Relational Database
- It has dynamic Schema.
- Every object in data model has a unique id assigned to it
- A data model is linked to another by including the other table's object id.
- Centers around the CAP theorem (Consistency, Availability, and Partition tolerance).

5.7.2 Qualities of our Database Design

- Reflects real-world structure of the problem
- Can represent all expected data over time
- Avoids redundant storage of data items
- Provides efficient access to data
- Supports the maintenance of data integrity over time
- Clean, consistent, and easy to understand
- No data redundancies (Reduced to 3NF).

Structure of Admin Schema

```
const adminSchema = new Schema({  
  name: {  
    type: String,  
    required: true  
  },  
  email: {  
    type: String,  
    required: true,  
    unique: true  
  },  
  password: {  
    type: String  
  },  
  registrationNumber: {  
    type: String  
  },  
  department: {  
    type: String  
  },  
  dob: {  
    type: String  
  },  
  joiningYear: {  
    type: String  
  },  
  avatar: {  
    type: String  
  },  
  contactNumber: {  
    type: Number  
  }  
, { strict: false })
```

Structure of attendance Schema

```
const attendanceSchema = new Schema({
  student: {
    type: Schema.Types.ObjectId,
    ref: 'student'
  },
  subject: {
    type: Schema.Types.ObjectId,
    ref: 'subject'
  },
  totalLecturesByFaculty: {
    type: Number,
    default: 0
  },
  lectureAttended: {
    type: Number,
    default: 0
  },
})
```

Structure of datewise attendance Schema

```
const datewiseSchema = new Schema({
  student: {
    type: Schema.Types.ObjectId,
    ref: 'student'
  },
  subject: {
    type: Schema.Types.ObjectId,
    ref: 'subject'
  },
  date: {
    type: String,
    required: true
  },
  status: {
    type: String,
  }
})
```

Structure of faculty Schema

```
const facultySchema = new Schema({  
    name: {  
        type: String,  
        required: true  
    },  
    email: {  
        type: String,  
        required: true,  
        unique: true  
    },  
    avatar: { type: String },  
    password: { type: String },  
    registrationNumber: { type: String },  
    gender: { type: String },  
    designation: {  
        type: String,  
        required: true  
    },  
    department: {  
        type: String,  
        required: true  
    },  
    facultyMobileNumber: {  
        type: Number  
    },  
    aadharCard: {  
        type: Number  
    },  
    dob: {  
        type: String,  
        required: true  
    },  
    joiningYear: {  
        type: Number,  
        required: true  
    },  
    subjectsCanTeach: [{  
        type: String  
    }]  
})
```

Structure of marks Schema

```
const markSchema = new Schema({  
    student: {  
        type: Schema.Types.ObjectId,  
        ref: 'student'  
    },  
    subject: {  
        type: Schema.Types.ObjectId,  
        ref: 'subject'  
    },  
    exam: {  
        type: String,  
        required:true  
    },  
    marks: {  
        type: Number,  
        default: 0  
    },  
    totalMarks: {  
        type: Number,  
        default: 100  
    },  
    department: {  
        type:String  
    },  
    semester: {  
        type:Number  
    },  
    section: {  
        type:String  
    }  
})
```

Structure of student Schema

```
const studentSchema = new Schema({  
    name: {  
        type: String,  
        required:true  
    },  
    email: {  
        type: String,  
        required: true,  
        unique: true  
    },  
    avatar: {  
        type: String  
    },  
    password: {  
        type: String,  
        required: true,  
    },  
    year: {  
        type: Number,  
        required: true  
    },  
    subjects: [  
        {  
            type: Schema.Types.ObjectId,  
            ref: 'subject'  
        }  
    ],  
    fatherName: {  
        type: String  
    },  
    aadharCard: {  
        type: Number  
    },  
    gender: {  
        type: String  
    },  
    registrationNumber: {  
        type: String  
    },  
    department: {  
        type: String,  
        required: true  
    },  
    section: {  
        type: String,  
    },  
    batch: {  
    }  
})
```

```

        type: String
    },
    dob: {
        type: String,
        required: true
    },
    studentMobileNumber: {
        type: Number
    },
    fatherMobileNumber: {
        type: Number
    },
    fatherName: {
        type: String
    },
    otp: {
        type: String
    }
})
}
)

```

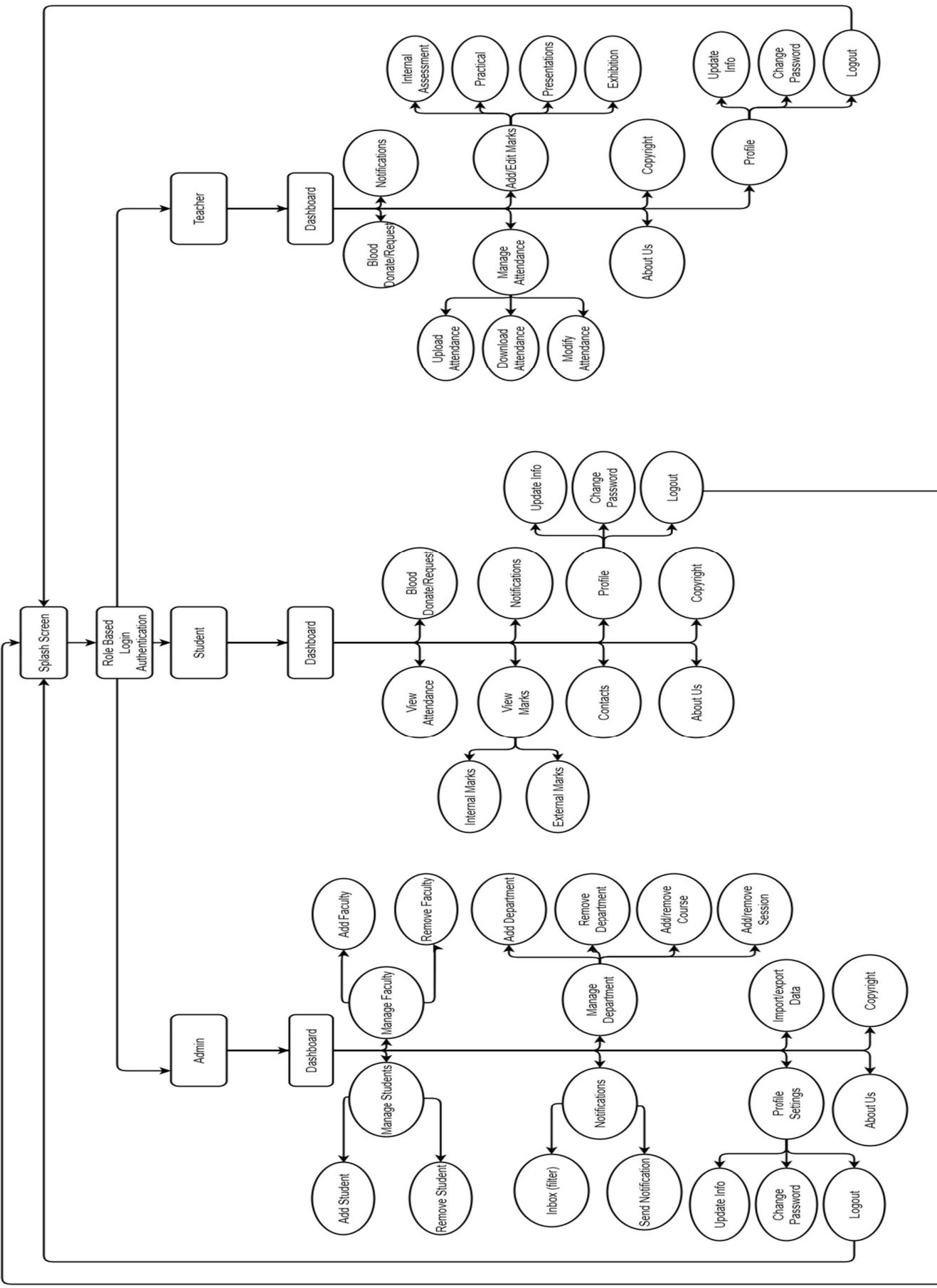
Structure of subject Schema

```

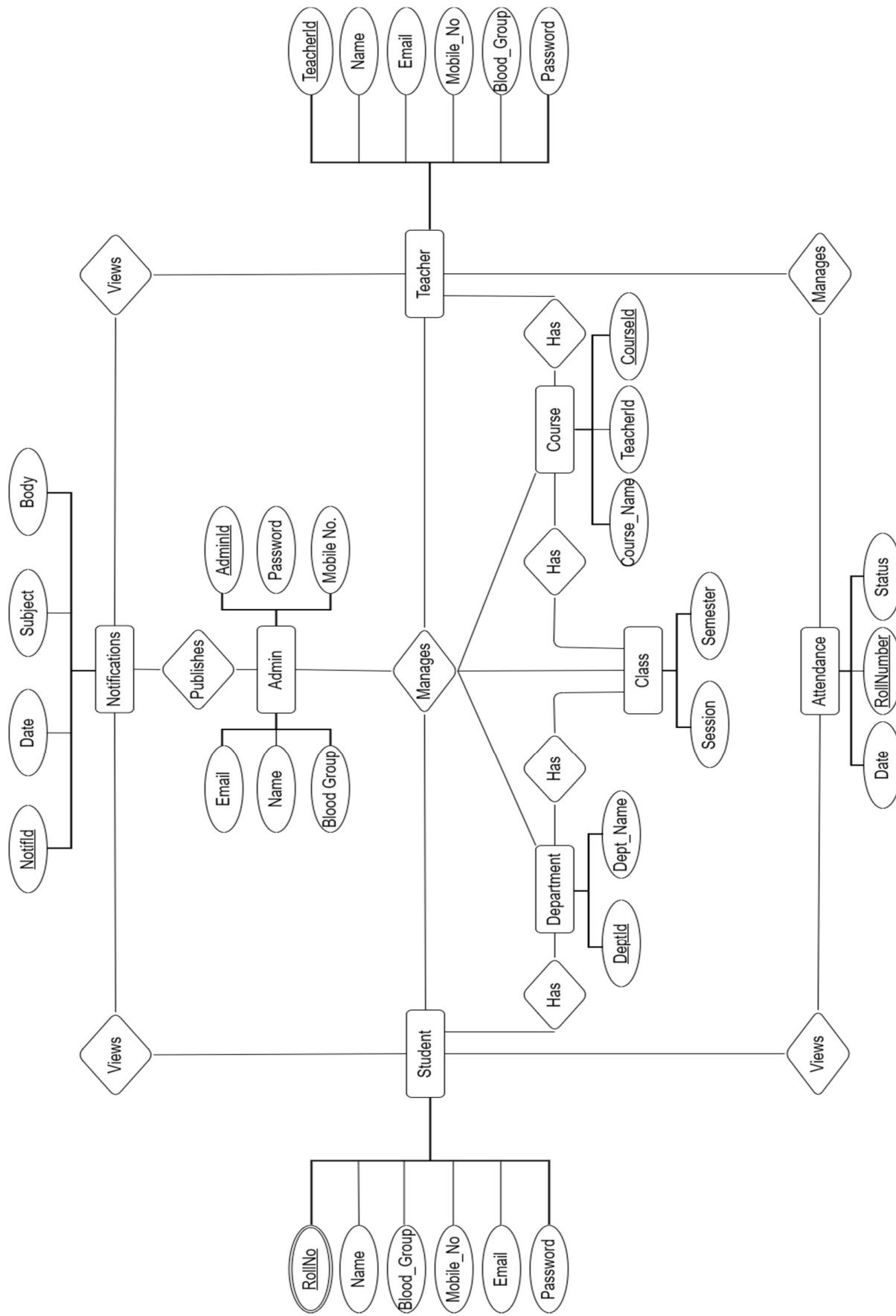
const subjectSchema = new Schema({
    department: {
        type: String,
        required: true
    },
    subjectCode: {
        type: String,
        required: true
    },
    subjectName: {
        type: String,
        required: true,
        trim: true
    },
    totalLectures: {
        type: Number,
        default: 30
    },
    year: {
        type: String,
        required: true
    },
    attendence: {
        type: Schema.Types.ObjectId,
        ref: 'attendence'
    }
})

```

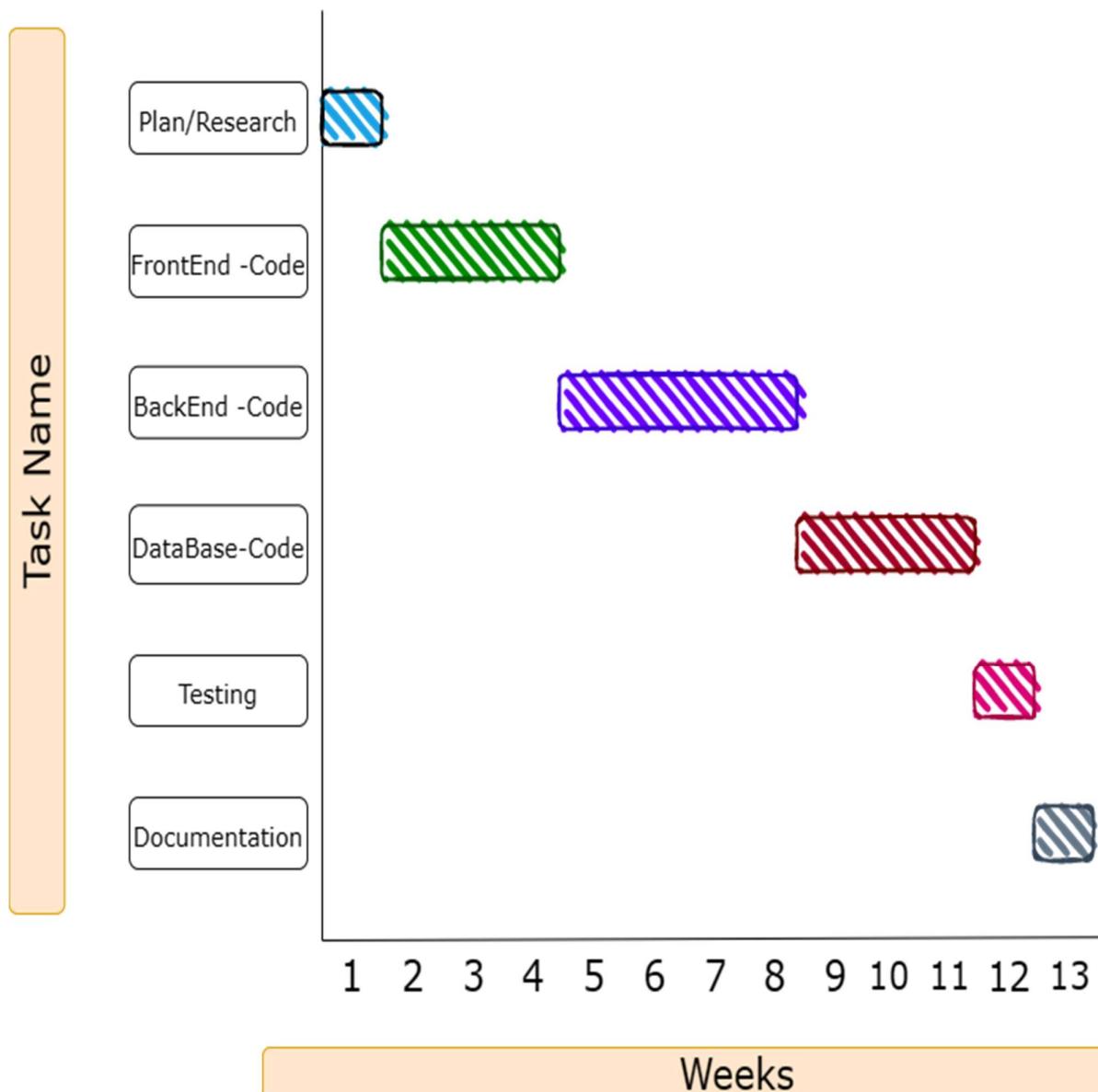
5.8 Module Hierarchy Diagram



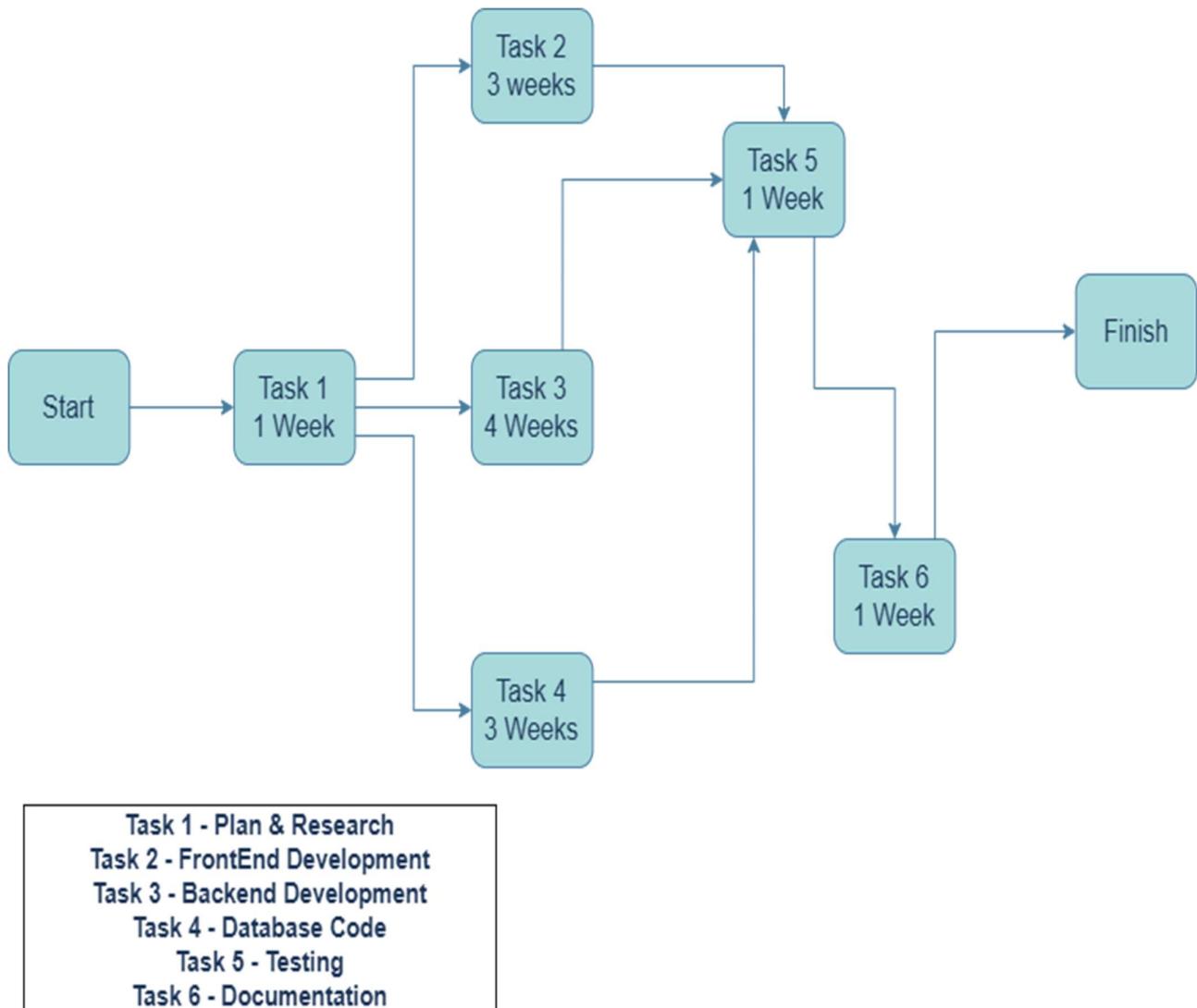
5.9 ER MODEL



5.10 GANTT CHART



5.11 PERT CHART



SYSTEM TESTING

6.1 Introduction

Once source code has been generated, software must be tested to uncover (and correct) as many errors as possible before delivery to customer. Our goal is to design a series of test cases that have a high likelihood of finding errors. To uncover the errors software techniques are used. These techniques provide systematic guidance for designing test that

- 1) Exercise the internal logic of software components, and
- 2) Exercise the input and output domains of the program to uncover errors in program function, behaviour and performance.

6.1.1 Steps: Software is tested from two different perspectives:

- 1) Internal program logic is exercised using —White box|| test case design Techniques.
- 2) Software requirements are exercised using —black box test case Design techniques.

In both cases, the intent is to find the maximum number of errors with the Minimum amount of effort and time.

6.2 Testing Methodologies:

A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements. A strategy must provide guidance for the practitioner and a set of milestones for the manager. Because the steps of the test strategy occur at a time when deadline pressure begins to rise, progress must be measurable and problems must surface as early as possible. Following testing techniques are well known and the same strategy is adopted during this project testing.

6.2.1 Unit testing:

Unit testing focuses verification effort on the smallest unit of software design the software component or module. The unit test is white-box oriented. The unit testing implemented in every module of student attendance management System. by giving correct manual input to the system, the datas are stored in database and retrieved. If you want required module to access input or get the output from the End user. any error will accrue the time will provide handler to show what type of error will be accrued.

6.2.2 System testing:

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Below we have described the two types of testing which have been taken for this project. it is to check all modules worked on input basis, if you want change any values or inputs will change all information. so specified input is must.

6.2.3 Performance Testing:

Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process. Even at the unit level, the performance of an individual module may be assessed as white-box tests are conducted.

This project reduces attendance table, codes. it will generate report fast.no have extra time or waiting of results entered correct data will show result few milliseconds just used only low memory of our system. Automatically do not getting access at another software. Get user permission and access to other applications.

6.3 Test cases:

Test case is an object for execution for other modules in the architecture does not represent any interaction by itself. A test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases: - manual and automated. A manual test case is executed manually while an automated test case is executed using automation.

In system testing, test data should cover the possible values of each parameter based on the requirements. Since testing every value is impractical, a few values should be chosen from each equivalence class. An equivalence class is a set of values that should all be treated the same.

Ideally, test cases that check error conditions are written separately from the functional test cases and should have steps to verify the error messages and logs. Realistically, if functional test cases are not yet written, it is ok for testers to check for error conditions when performing normal functional test cases. It should be clear which test data, if any is expected to trigger errors.

6.3.1 Admin form

Sno	Test case id	Test case name	Test case desc	Step	Expected result	Actual Result	Test case status pass/fail
1	Login Admin	Validate login	To verify that login name on login page	Enter the login name and password and click on submit button	Login successful or an error message “In valid login or password” must be displayed	Login successful	Pass
2	Add Teaches	Validate to add teachers	To verify that to add teachers on add teachers page	Enter the teachers detail and click on add button	Add successful or an error message “email-id already exist” must be displayed	Add successful	Pass
3	Add Student	Validate to add students	To verify that to add students on add student page	Enter the students detail and click on add button	Add successful or an error message “email-id already exist” must be displayed	Add successful	Pass
4	Add Admin	Validate to add admin	To verify that to add admin on add admin page	Enter the admins detail and click on add button	Add successful or an error message “email-id already exist” must be displayed	Add successful	Pass
5	Add Subject	Validate to add subject	To verify that to add subject on add subject page	Enter the subject and click on add button	Add successful	Add successful	Pass
6	Teachers Data	Validate teachers data	To verify that to see teachers data on teachers data page	Enter the department and year and click on search button	Display successful	Display successful	Pass
7	Students Data	Validate students data	To verify that to see students data on student data page	Enter the department and year and click on search button	Display successful	Display successful	Pass

8	Subjects Data	Validate subjects data	To verify that to see students data on student data page	Enter the department and year and click on search button	Display successful	Display successful	Pass
9	Update Teachers Data	Validate teachers data	To verify that to update teachers data on update teacher data page	Select the teachers data and then update	Teachers data updated successfully	An error message “Sorry, the page you are looking for does not exit” must be displayed	Fail
10	Update Students Data	Validate students data	To verify that to update students data on update student data page	Select the department and year for update the students data	Students data updated successfully	An error message “Sorry, the page you are looking for does not exit” must be displayed	Fail
11	Update Admin Data	Validate admins data	To verify that to update admins data on update admin data page	Select the admin data and then update	Students data updated successfully	An error message “Sorry, the page you are looking for does not exit” must be displayed	Fail
12	Update Subject Data	Validate subject data	To verify that to update subjects data on update admin data page	Select the department and year for update the subject data	Subjects data updated successfully	An error message “Sorry, the page you are looking for does not exit” must be displayed	Fail

6.3.2 Teacher form

Sno	Test caseid	Test case name	Test case desc	Step	Expected result	Actual Result	Test case status pass/fail
1	Login Teacher	Validate login	To verify that loginname on login page	Enter the registration number and password and click submit button	Login successful or an error message "In valid login or password" must be displayed	Login successful	Pass
2	Forgot Password	Validate forgot password	To verify the forgot password on login page	Enter the new password and click submit button	Password changed successfully	An error message "Sorry, the page you are looking for does not exit" must be displayed	Fail
3	Mark Attendance	Validate attendance	To verify the attendance on attendance page	Enter your department, year and subject and then mark students status and click on submit button	Attendance Marked successful	Attendance Marked successful	Pass
4	Update attendance	Validate to update attendance	To verify that to update attendance on update attendance Page	Select the department, year and subject for update attendance	Attendance updated successfully	An error message "Sorry, the page you are looking for does not exit" must be displayed	Fail
5	Generate Reports	Validate to generate reports	To verify that to generate student reports	Select the department, year and subject for generate reports	Reports generated successfully	An error message "Sorry, the page you are looking for does not exit" must be displayed	Fail

6	Upload Marks	Validate to upload marks	To verify that to upload marks on upload marks page	Enter department, year and subject and then upload marks and click on submit button	Marks uploaded successfully	Marks uploaded successful	Pass
7	Update Marks	Validate to update marks	To verify that to update marks on update marks page	Change the marks of the selected department, year and subject	Marks updated successfully	An error message “Sorry, the page you are looking for does not exit” must be displayed	Fail
5	Update Profile	Validate Profile	To verify that to update profile on Profile page	Change your details and click on submit button	Profile updated successfully and show your updated profile	Profile updated successfully and show your updated profile	Pass
6	Change Password	Validate Password	To verify that to change password on Change password page	Enter your new password and click on submit button	Password changed successfully and now you can login with your new password	Password changed successfully and now you can login with your new password	Pass
7	Donate Blood	Validate the donate blood	To verify that to see the donate blood request	Fill the form and click on submit button	Blood donation form filled successfully	An error message “Sorry, the page you are looking for does not exit” must be displayed	Fail

6.3.3 Student form

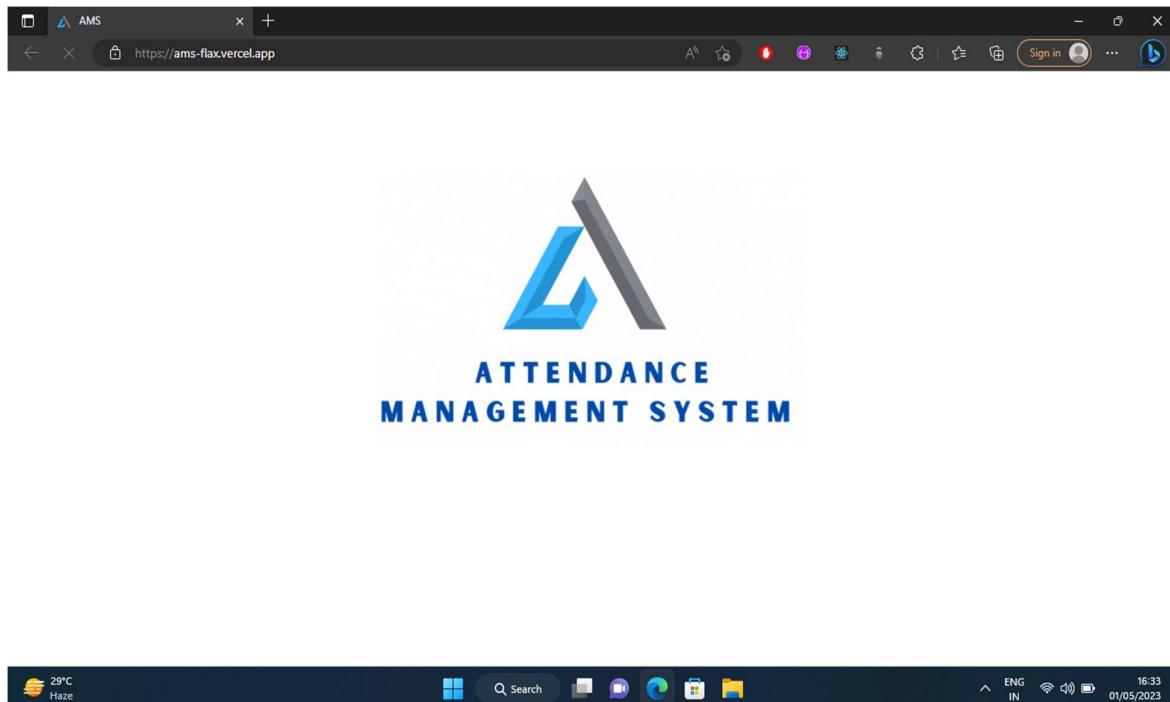
Sno	Test case id	Test case name	Test casedesc	Step	Expected result	Actual Result	Test case status pass/fail
1	Login Student	Validate login	To verify that login name on login page	Enter the registration number and password and click submit button	Login successful or an error message "In valid login or password" must be displayed	Login successful	Pass
2	Forgot Password	Validate forgot password	To verify the forgot password on login page	Enter the new password and click submit button	Password changed successfully	An error message "Sorry, the page you are looking for does not exist" must be displayed	Fail
3	View Attendance	Validate attendance	To verify the attendance on attendance page	Enter your department, year and subject and click on search button	Search successful and show your attendance	Search successful and show your attendance	Pass
4	View Marks	Validate Marks	To verify the marks on marks page	Enter your department, year and subject and click on search button	Search successful and show your marks	Search successful and show your marks	Pass
5	Update Profile	Validate Profile	To verify that to update profile on Profile page	Change your details and click on submit button	Profile updated successfully and show your updated profile	Profile updated successfully and show your updated profile	Pass

6	Change Password	Validate Password	To verify that to change password on Change password page	Enter your new password and click on submit button	Password changed successfully and now you can login with your new password	Password changed successfully and now you can login with your new password	Pass
7	View Subject List	Validate the subject list	To verify that to see the subject list	Enter your department and year and click on search button	Subject list will be displayed successfully	Subject list will be displayed successfully	Pass
8	Donate Blood	Validate the donate blood	To verify that to fill the form for donate blood request	Fill the form and click on submit button	Blood donation form filled successfully	An error message "Sorry, the page you are looking for does not exist" must be displayed	Fail

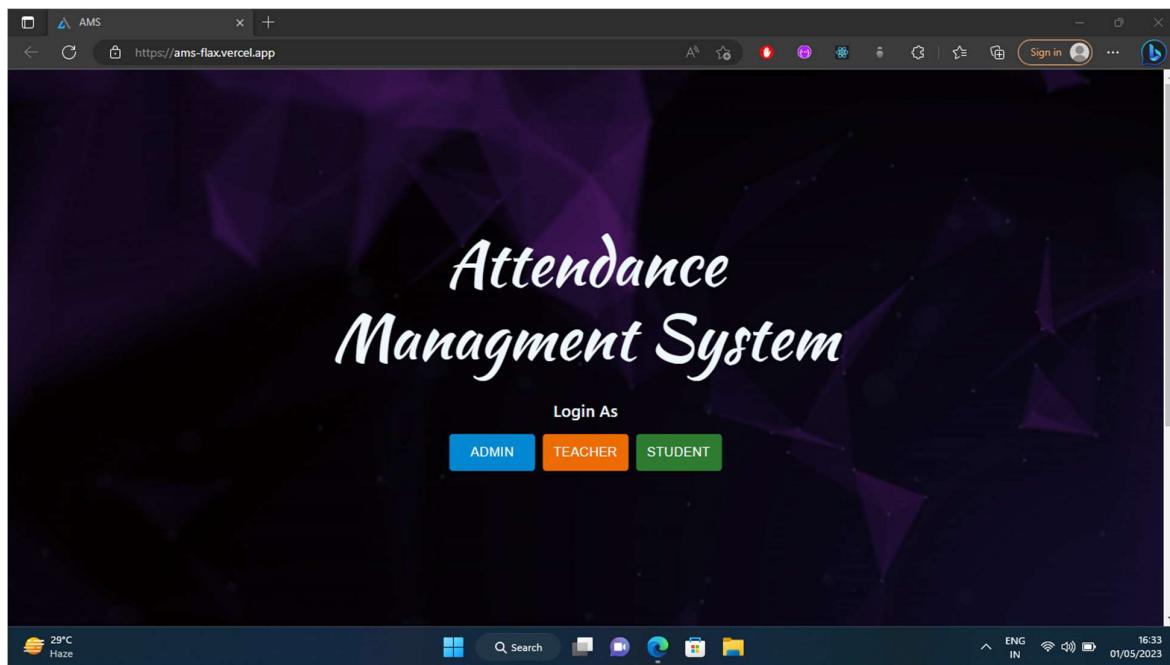
SCREENSHOTS

7. Desktop View:

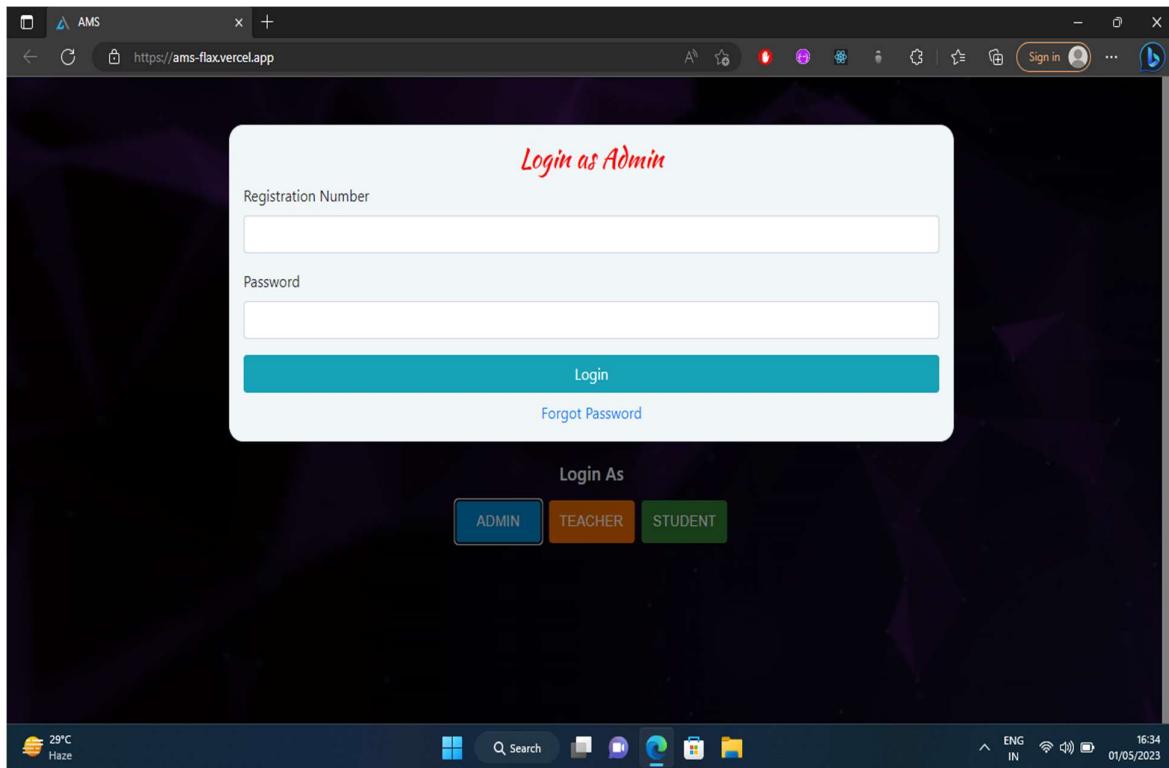
Splash Screen:



Front Page:



Admin Login Page:



Admin Dashboard:

The screenshot shows the 'AMS ADMIN' dashboard. On the left is a sidebar with a dark blue background containing various administrative links: Profile, Add Teachers, Add Students, Add Subjects, Add Admin, Post Notice, Remove Teacher, Remove Student, Remove Subject, Update Admin, Update Teachers, Update Students, Update Subjects, About Us, and a red 'LOGOUT' button. The main area is titled 'Welcome : AKASH'. It features four colored boxes: green for 'TEACHERS' (0), red for 'STUDENTS' (0), teal for 'SUBJECTS' (0), and yellow for 'ADMIN-NAME' (AKASH). Below this is a user profile card for 'akash' with a placeholder profile picture. To the right is a table with the following data:

Full Name	akash
Email	admin@mail.com
Phone	7895412360
Registration-Number	ADM202301000
Joining year	2023
Department	BCA

At the bottom left of the main area, the URL 'https://ams-ebon.vercel.app/admin' is visible.

Add Teachers Page:

AMS ADMIN

Welcome : AKASH

TEACHERS STUDENTS SUBJECTS ADMIN-NAME

AKASH

Faculty Name

Email

Designation

Department

DOB dd/mm/yyyy

Gender

Contact Number

Aadhar Card Number

Add Faculty

<https://ams-ebon.vercel.app/admin/addFaculty>

Add Students Page:

AMS ADMIN

Welcome : AKASH

TEACHERS STUDENTS SUBJECTS ADMIN-NAME

AKASH

Student Name

Gender

Email

Contact Number

Department

Father Name

Year

Father Contact Number

Section

College Roll Number

DOB dd/mm/yyyy

Add Student

<https://ams-ebon.vercel.app/admin/addStudent>

Add Subjects Page:

AMS ADMIN

Welcome : AKASH

TEACHERS STUDENTS SUBJECTS ADMIN-NAME

0 0 0 AKASH

Subject Name:

Subject Code:

Total Lectures:

Department:

Year:

Add Subject

Add Admin Page:

AMS ADMIN

Welcome : AKASH

TEACHERS STUDENTS SUBJECTS ADMIN-NAME

0 0 0 AKASH

Admin Name:

DOB: dd/mm/yyyy

Email:

Contact Number:

Department:

Add Admin

Logout

Remove Teachers Page1:

The screenshot shows the AMS Admin dashboard with a dark blue sidebar on the left containing various administrative options. The main area displays four colored boxes: green for Teachers (0), red for Students (0), teal for Subjects (0), and yellow for Admin-NAME (AKASH). Below these boxes is a search interface with a dropdown menu labeled 'Select' and a 'Search' button.

Remove Teachers Page2:

The screenshot shows the AMS Admin dashboard with a dark blue sidebar on the left containing various administrative options. The main area displays four colored boxes: green for Teachers (4), red for Students (0), teal for Subjects (0), and yellow for Admin-NAME (AKASH). Below these boxes is a search interface with a dropdown menu set to 'BCA' and a 'Search' button. A table below lists four users with their details and a delete icon.

S.No	Registration Number	Name	Email	Joining Year	Delete
1	FAC202301000	Mr. Prakash Upadhyay	upadhyay@gmail.com	2023	
2	FAC202301001	Mr. Rakesh Kumar Pathak	pathak@gmail.com	2023	
3	FAC202301002	Mrs. Supriya Shree	shree@gmail.com	2023	
4	FAC202301003	Mr. Mukesh Kumar	kumar@gmail.com	2023	

Remove Students Page1:

The screenshot shows the AMS Admin dashboard. On the left, a sidebar lists various administrative functions: Profile, Add Teachers, Add Students, Add Subjects, Add Admin, Post Notice, Remove Teacher, Remove Student, Remove Subject, Update Admin, Update Teachers, Update Students, Update Subjects, and About Us. A red 'LOGOUT' button is at the bottom. The main area is titled 'AMS ADMIN' and 'Welcome : AKASH'. It features four colored boxes: a green 'TEACHERS' box with '0', a red 'STUDENTS' box with '0', a teal 'SUBJECTS' box with '0', and a yellow 'ADMIN-NAME' box with 'AKASH'. Below these are dropdown menus for 'Department' (Select) and 'Year' (Select), and a blue 'Search' button.

Remove Students Page2:

The screenshot shows the AMS Admin dashboard after performing a search. The sidebar and top navigation are identical to the first screenshot. The main area now displays a table of student records. The table has columns: S.No, Registration Number, Name, Email, Section, and Delete. The data is as follows:

S.No	Registration Number	Name	Email	Section	Delete
1	STU202301020	Ravi Shankar	ravi@gmail.com	A	
2	STU202301021	Aditya Prakash	aditya12@gmail.com	A	
3	STU202301022	Esha Solomon	esha@gmail.com	A	
4	STU202301023	Sourav Kumar	sourav@gmail.com	A	
5	STU202301024	Preety Kumari	preety@gmail.com	A	
6	STU202301025	Ved Prakash	ved@gmail.com	A	

At the bottom left, a URL is visible: <https://ams-ebon.vercel.app/admin/allStudents>.

Remove Subjects Page1:

AMS ADMIN

Welcome : AKASH

TEACHERS STUDENTS SUBJECTS ADMIN-NAME

0 0 0 AKASH

Department: Select

Year: Select

Search

Profile
Add Teachers
Add Students
Add Subjects
Add Admin
Post Notice
Remove Teacher
Remove Student
Remove Subject
Update Admin
Update Teachers
Update Students
Update Subjects
About Us

LOGOUT

Remove Subjects Page2:

AMS ADMIN

Welcome : AKASH

TEACHERS STUDENTS SUBJECTS ADMIN-NAME

4 11 6 AKASH

Department: BCA

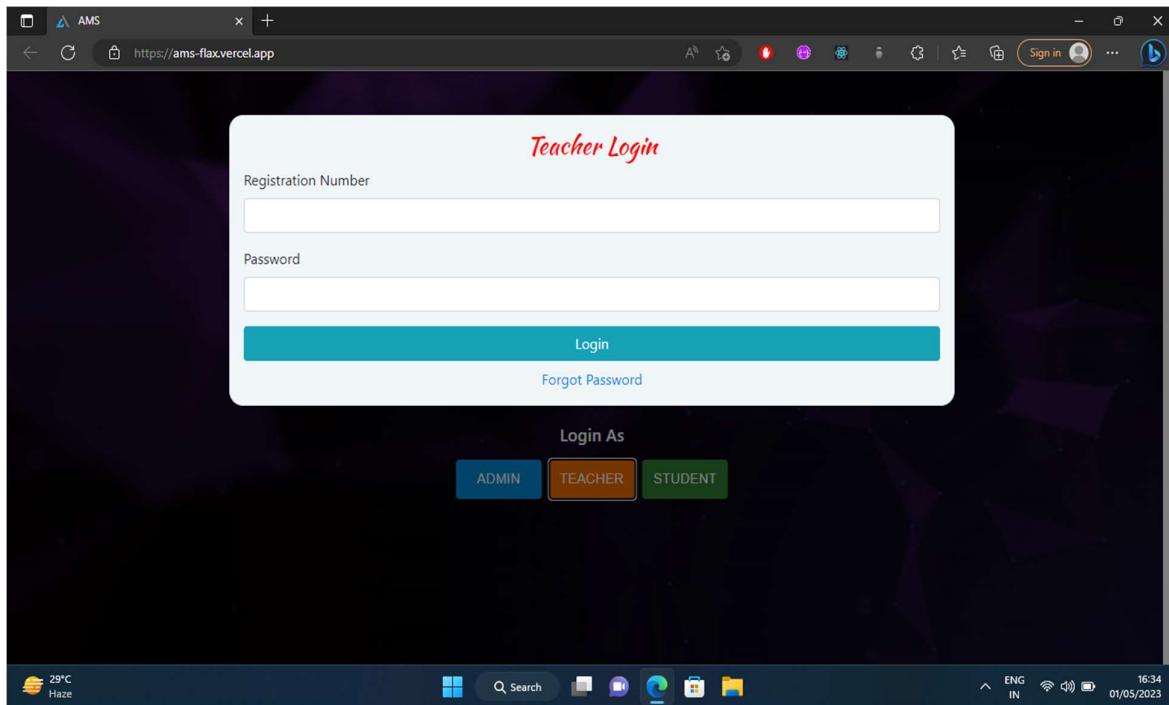
Year: 3

Search

S.No	Subject Code	Subject Name	Total Lectures	Delete
1	BCA-501	VB.net	20	
2	BCA-502	Graphics	20	
3	BCA-503	Computer Network	25	
4	BCA-504	Accounts	20	
5	BCA-601	Web Technology	25	
6	BCA-603	E-Commerce	20	

<https://ams-ebon.vercel.app/admin/allSubject>

Teachers Login Page:



Teacher Dashboard:

Full Name	Mr. Prakash Upadhyay
Email	upadhyay@gmail.com
Phone	NA
Registration-Number	FAC202301000
Joining year	2023
Department	BCA

<https://ams-ebon.vercel.app/faculty>

Update Profile Page:

The screenshot shows the AMS Teacher Dashboard with a dark blue sidebar on the left containing various navigation links. The main content area displays a welcome message "Welcome : MR. PRAKASH UPADHYAY". On the right side, there are several input fields for updating profile information: "Profile Picture" (with a "Choose File" button and a note "No file chosen"), "Gender" (a dropdown menu set to "Select"), "Contact Number" (an empty input field), "Aadhar Card Number" (an empty input field), and a "Update" button at the bottom.

Mark Attendance Page1:

The screenshot shows the AMS Teacher Dashboard with a dark blue sidebar on the left containing various navigation links. The main content area displays a welcome message "Welcome : MR. PRAKASH UPADHYAY". On the right side, there are three dropdown menus for selecting "Department", "Year", and "Section", followed by a "Search" button at the bottom.

Mark Attendance Page2:

AMS Teacher Dashboard

Welcome : MR. PRAKASH UPADHYAY

Subject Code
Select

Select Date --> dd/mm/yyyy

Status	Roll Number	Student Name
□	BCA2020001	Ravi Shankar
□	BCA2020002	Aditya Prakash
□	BCA2020003	Esha Solomon
□	BCA2020005	Sourav Kumar
□	BCA2020006	Preety Kumari
□	BCA2020007	Ved Prakash
□	BCA2020008	Kunal Kumar
□	BCA@020009	Mohit Kumar
□	BCA2020010	Rohit Kumar
□	BCA2020054	Archana Kumari
□	BCA2020060	Akash Kumar

Submit Back

<https://ams-ebon.vercel.app/attendanceFaculty>

Upload Marks Page1:

AMS Teacher Dashboard

Welcome : MR. PRAKASH UPADHYAY

Department
Select

Year
Select

Section
Select

Search

<https://ams-ebon.vercel.app/faculty/uploadMarks>

Upload Marks Page2:

AMS Teacher Dashboard

Welcome : MR. PRAKASH UPADHYAY

Roll Number	Student Name	Marks
BCA2020001	Ravi Shankar	
BCA2020002	Aditya Prakash	
BCA2020003	Esha Solomon	
BCA2020005	Sourav Kumar	
BCA2020006	Preeti Kumari	
BCA2020007	Ved Prakash	
BCA2020008	Kunal Kumar	

<https://ams-ebon.vercel.app/faculty/uploadMarks>

Update Password Page:

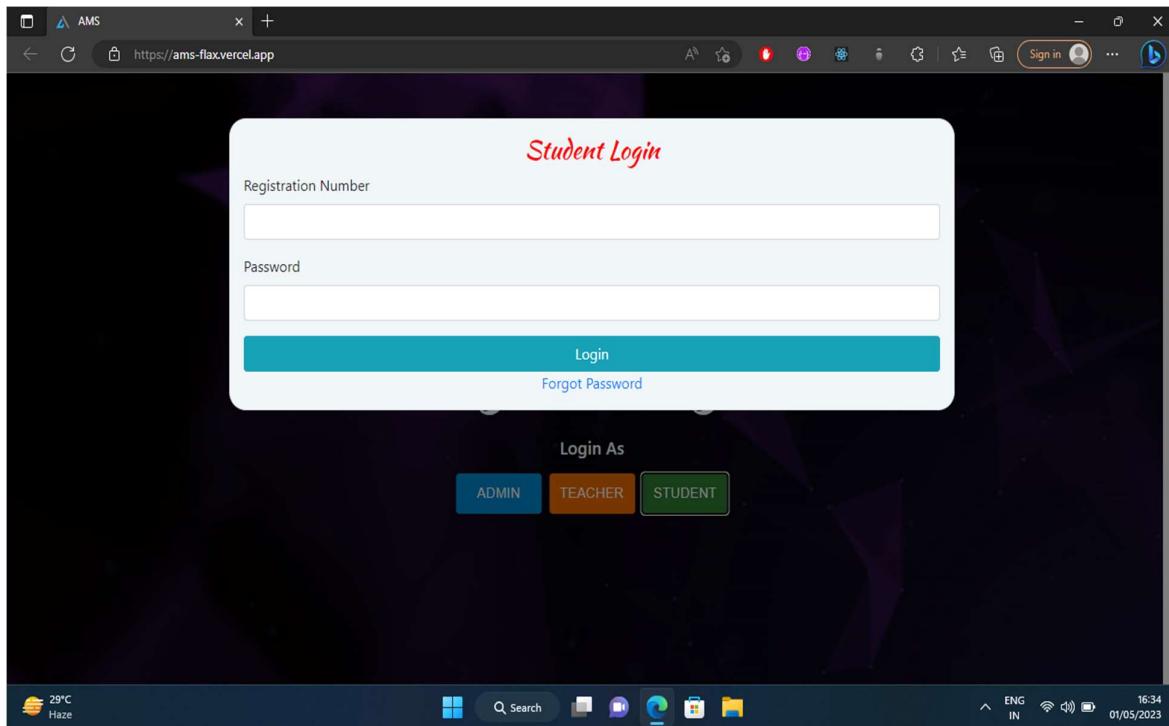
AMS Teacher Dashboard

Welcome : MR. PRAKASH UPADHYAY

Old Password	<input type="text"/>
New Password	<input type="text"/>
Confirm New Password	<input type="text"/>

<https://644ec8be1e322e0b613bf845--bespoke-kulfi-bb7bcf.netlify.app/faculty/updatePassword>

Student Login Page:



Student Dashboard Page1:

A screenshot of the 'AMS Student Dashboard'. The title 'AMS Student Dashboard' is at the top. On the left, a vertical sidebar menu lists several options: 'Profile', 'Update profile', 'Results', 'Attendance', 'Subject List', 'Chat (0)', 'Donate Blood', 'Update Password', 'Contact Us', 'About Us', and a red 'LOGOUT' button. The main content area displays a welcome message 'Welcome : AKASH KUMAR' above a circular profile picture of a man with glasses. Below the profile picture, the name 'AKASH KUMAR' and the ID 'BCA2020060' are shown. At the bottom left of the dashboard, the URL https://ams-ebon.vercel.app/home is visible.

Student Dashboard Page2:

The screenshot shows the AMS Student Dashboard. On the left is a sidebar with the following menu items:

- Profile
- Update profile
- Results
- Attendance
- Subject List
- Chat (0)
- Donate Blood
- Update Password
- Contact Us
- About Us

A red "LOGOUT" button is at the bottom of the sidebar.

The main content area has a dark header bar with the text "AMS Student Dashboard". Below it, a welcome message reads "Welcome : AKASH KUMAR". In the center is a circular profile picture of a man with glasses and a beard. To the right of the picture, the user's name "AKASH KUMAR" is displayed, along with their roll number "BCA2020060", gender "GENDER: MALE", and date of birth "D.O.B. : 01-05-2023". Below this information are the email "EMAIL: AK65469@GMAIL.COM" and mobile number "MOBILE NUMBER: 8271400511". At the bottom of the central area, the text "FATHER NAME:" is visible.

Update Profile Page:

The screenshot shows the AMS Student Dashboard with the "Update profile" option selected in the sidebar. The main content area displays the same welcome message and user information as the previous screenshot. However, the central area now contains form fields for updating personal details:

- Profile Picture: A file input field labeled "Choose File" with the placeholder "No file chosen".
- Gender: A dropdown menu labeled "Select".
- Contact Number: An input field.
- Father Name: An input field.
- Father Contact Number: An input field.
- College Roll Number: An input field containing the value "BCA2020060".

A blue "Update" button is located at the bottom of the form.

Results Page:

AMS Student Dashboard

Welcome : AKASH KUMAR

InternalAssessment

S.No	Subject Code	Subject Name	Obtained Marks	Total Marks	Percentage
1	BCA-501	VB.net	38	40	95.00%
2	BCA-502	Graphics	38	40	95.00%
3	BCA-503	Computer Network	36	40	90.00%
4	BCA-504	Accounts	37	40	92.50%

Practical 1

S.No	Subject Code	Subject Name	Obtained Marks	Total Marks	Percentage
1	BCA-501	VB.net	21	40	52.50%
2	BCA-502	Graphics	22	40	55.00%

Attendance Page1:

AMS Student Dashboard

Welcome : AKASH KUMAR

WEB TECHNOLOGY
ATTENDED - 2 50.0%
TOTAL CLASSES - 4

E-COMMERCE
ATTENDED - 3 75.0%
TOTAL CLASSES - 4

Attendance Page2:

AMS Student Dashboard

Welcome : AKASH KUMAR

CLOSE

S. no.	Status	Date
1	Present	2023-05-28
2	Present	2023-05-29
3	Absent	2023-05-29
4	Absent	2023-05-31

Subject List Page:

AMS Student Dashboard

Welcome : AKASH KUMAR

S.No	Subject Code	Subject Name	Year	Total Lectures
1	BCA-501	VB.net	3	20
2	BCA-502	Graphics	3	20
3	BCA-503	Computer Network	3	25
4	BCA-504	Accounts	3	20
5	BCA-601	Web Technology	3	25
6	BCA-603	E-Commerce	3	20

Update Password Page:

The screenshot shows the AMS Student Dashboard interface. On the left, there is a dark sidebar menu with various options: Profile, Update profile, Results, Attendance, Subject List, Chat (0), Donate Blood, Update Password (highlighted in blue), Contact Us, About Us, and Logout. The main content area is titled "AMS Student Dashboard" and "Welcome : AKASH KUMAR". It contains three input fields for password update: "Old Password", "New Password", and "Confirm New Password", each with a corresponding text input box. Below these fields is a teal-colored "Update Password" button.

AMS Student Dashboard

Welcome : AKASH KUMAR

Old Password

New Password

Confirm New Password

Update Password

Mobile View:

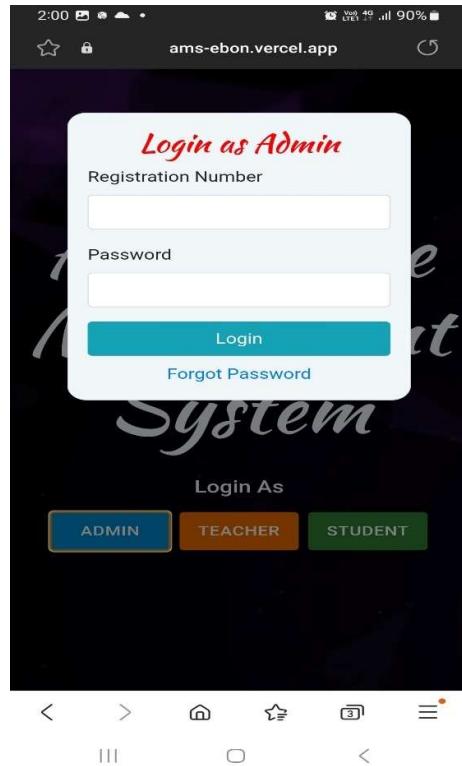
Splash Screen:



Front Page:



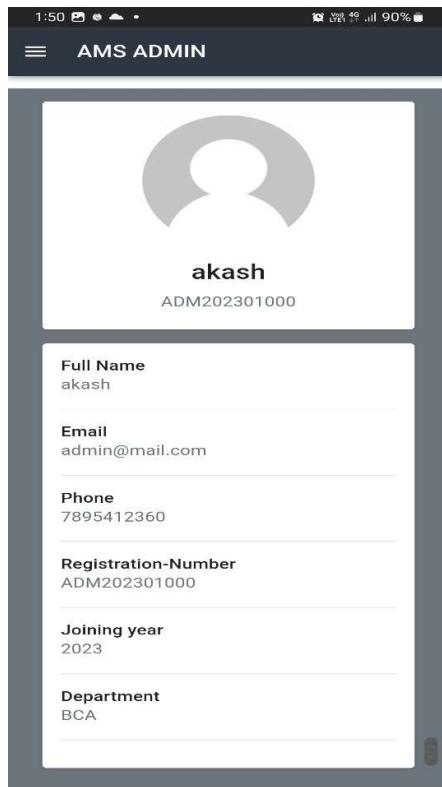
Admin Login Page:



Admin Dashboard:



Admin Dashboard:



Add Teachers Page:

This screenshot displays the "Add Faculty" form within the AMS Admin app. It includes the following input fields:

- Faculty Name:** (Text input field)
- Email:** (Text input field)
- Designation:** (Select dropdown menu with option "Select")
- Department:** (Select dropdown menu with option "Select")
- DOB:** (Text input field)
- Gender:** (Select dropdown menu with option "Select")
- Contact Number:** (Text input field)
- Aadhar Card Number:** (Text input field)
- Add Faculty:** (A teal-colored button at the bottom left)

Add Students Page:

1:53 4G 90% AMS ADMIN

Student Name
Email
Department
Select
Year
Select
Section
Select
DOB
Gender
Select
Contact Number
Father Name
Father Contact Number
College Roll Number
Add Student

Add Subjects Page:

1:54 4G 90% AMS ADMIN

Subject Name
Subject Code
Total Lectures
Department
Select
Year
Select
Add Subject

Add Admin Page:

1:55 90% ■

☰ AMS ADMIN

Admin Name
[Text Input]

Email
[Text Input]

Department
[Select]

DOB
[Text Input]

Contact Number
[Text Input]

Add Admin

||| □ <

Remove Teachers Page1:

1:55 90% ■

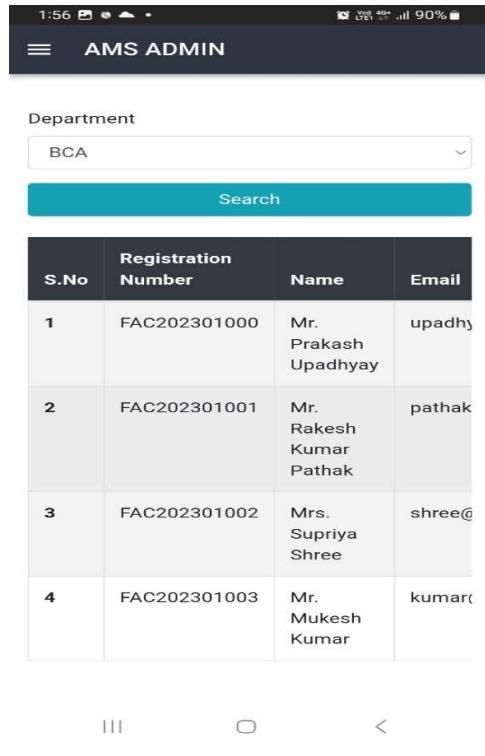
☰ AMS ADMIN

Department
[Select]

Search

||| □ <

Remove Teachers Page2:

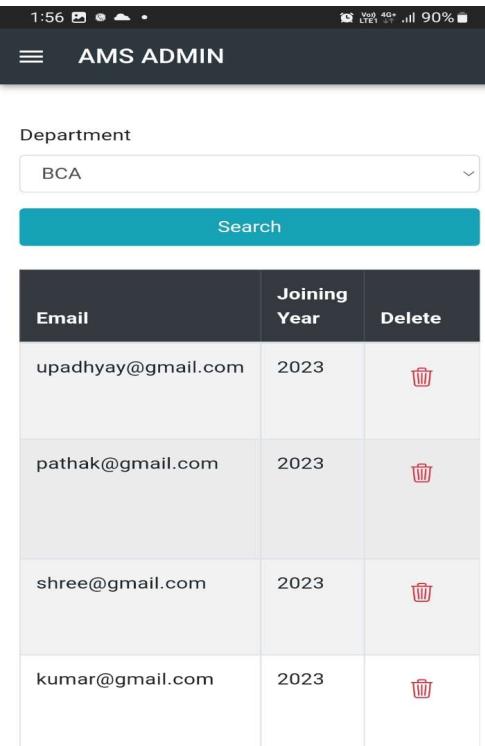


The screenshot shows a mobile application interface titled "AMS ADMIN". At the top, there is a header bar with the title and some icons. Below it, a dropdown menu labeled "Department" is set to "BCA". A teal-colored "Search" button is positioned below the dropdown. The main content area displays a table with four columns: "S.No", "Registration Number", "Name", and "Email". There are four rows of data in the table.

S.No	Registration Number	Name	Email
1	FAC202301000	Mr. Prakash Upadhyay	upadhy
2	FAC202301001	Mr. Rakesh Kumar Pathak	pathak
3	FAC202301002	Mrs. Supriya Shree	shree@
4	FAC202301003	Mr. Mukesh Kumar	kumar

||| ○ <

Remove Teachers Page2:

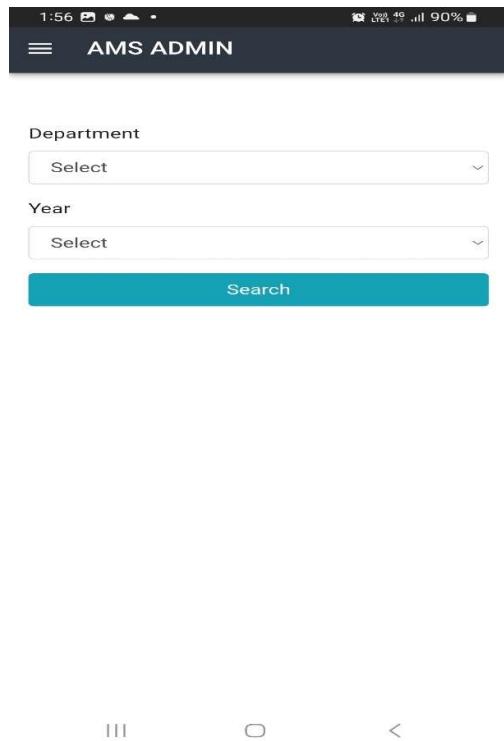


The screenshot shows a mobile application interface titled "AMS ADMIN". At the top, there is a header bar with the title and some icons. Below it, a dropdown menu labeled "Department" is set to "BCA". A teal-colored "Search" button is positioned below the dropdown. The main content area displays a table with three columns: "Email", "Joining Year", and "Delete". There are four rows of data in the table, each with a red trash icon in the "Delete" column.

Email	Joining Year	Delete
upadhyay@gmail.com	2023	trash
pathak@gmail.com	2023	trash
shree@gmail.com	2023	trash
kumar@gmail.com	2023	trash

||| ○ <

Remove Students Page1:



1:56 100% 90% 9:49 AM

AMS ADMIN

Department
Select

Year
Select

Search

||| ○ <

Remove Students Page2:



1:59 100% 90% 9:49 AM

AMS ADMIN

Department
Select

Year
Select

Search

S.No	Registration Number	Name	Email
1	STU202301020	Ravi Shankar	ravi@gr
2	STU202301021	Aditya Prakash	aditya1
3	STU202301022	Esha Solomon	esha@cs
4	STU202301023	Sourav Kumar	sourav@
5	STU202301024	Preety Kumari	preety@
6	STU202301025	Ved Prakash	ved@gr
7	STU202301026	Kunal Kumar	kunal@
8	STU202301027	Mohit Kumar	mohit@
9	STU202301028	Rohit Kumar	rohit@c
10	STU202301031	Archana Kumari	archan
11	STU202301029	Akash	ak6546

Remove Students Page2:

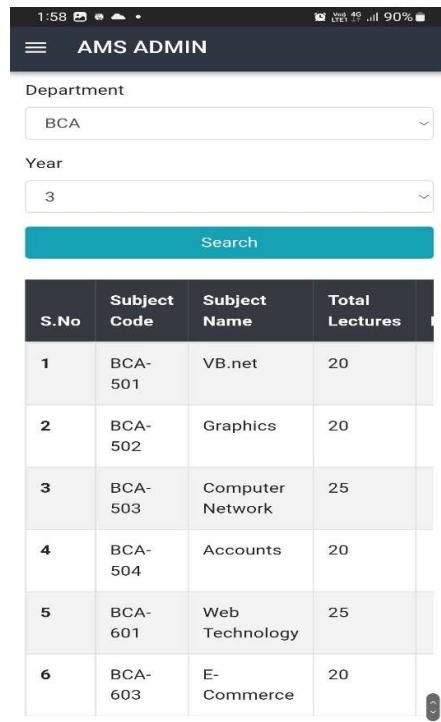
The screenshot shows a mobile application interface titled "AMS ADMIN". At the top, there are dropdown menus for "Department" (set to "Select") and "Year" (set to "Select"). Below these is a teal-colored "Search" button. The main area is a table with three columns: "Email", "Section", and "Delete". The "Email" column lists various student email addresses, and the "Section" column shows all entries as "A". The "Delete" column contains a red trash can icon in each row. A vertical scroll bar is visible on the right side of the table.

Email	Section	Delete
vi@gmail.com	A	trash
litya12@gmail.com	A	trash
sha@gmail.com	A	trash
xurav@gmail.com	A	trash
eety@gmail.com	A	trash
id@gmail.com	A	trash
inal@gmail.com	A	trash
ohit@gmail.com	A	trash
hit@gmail.com	A	trash
chana3647@gmail.com	A	trash
.....@gmail.com	A	trash

Remove Subjects Page1:

The screenshot shows a mobile application interface titled "AMS ADMIN". At the top, there are dropdown menus for "Department" (set to "Select") and "Year" (set to "Select"). Below these is a teal-colored "Search" button. At the bottom of the screen, there are three small navigation icons: three vertical lines, a square, and a left arrow.

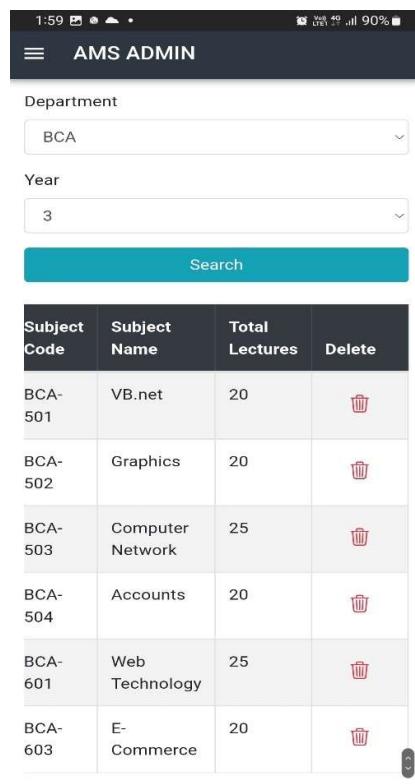
Remove Subjects Page2:



The screenshot shows a mobile application interface titled "AMS ADMIN". At the top, there are dropdown menus for "Department" (set to "BCA") and "Year" (set to "3"). Below these is a teal-colored "Search" button. The main area displays a table with the following data:

S.No	Subject Code	Subject Name	Total Lectures	Action
1	BCA-501	VB.net	20	
2	BCA-502	Graphics	20	
3	BCA-503	Computer Network	25	
4	BCA-504	Accounts	20	
5	BCA-601	Web Technology	25	
6	BCA-603	E-Commerce	20	

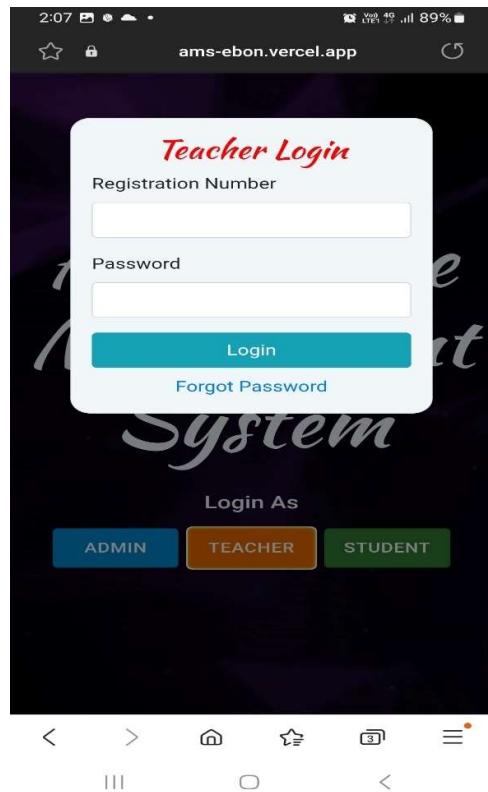
Remove Subjects Page2:



The screenshot shows the same mobile application interface as the previous one. The "Department" and "Year" dropdowns are set to "BCA" and "3" respectively. The "Search" button is present. The table data is identical to the first screenshot, but now includes a "Delete" column with a red trash can icon in each row.

Subject Code	Subject Name	Total Lectures	Delete
BCA-501	VB.net	20	
BCA-502	Graphics	20	
BCA-503	Computer Network	25	
BCA-504	Accounts	20	
BCA-601	Web Technology	25	
BCA-603	E-Commerce	20	

Teachers Login Page:



Teacher Dashboard:



Update Profile Page:

The screenshot shows the 'AMS Teacher Dashboard' interface. At the top, it displays 'Welcome : MR. PRAKASH UPADHYAY'. Below this, there are fields for updating profile information: 'Profile Picture' (with a 'Choose file' button and 'No file chosen' message), 'Gender' (a dropdown menu set to 'Select'), 'Contact Number' (an empty input field), 'Aadhar Card Number' (an empty input field), and a large blue 'Update' button at the bottom.

||| O <

Mark Attendance Page1:

The screenshot shows the 'AMS Teacher Dashboard' interface. At the top, it displays 'Welcome : MR. PRAKASH UPADHYAY'. Below this, there are three dropdown menus for filtering attendance data: 'Department' (set to 'Select'), 'Year' (set to 'Select'), and 'Section' (set to 'Select'). At the bottom, there is a large blue 'Search' button.

||| O <

Mark Attendance Page2:

Subject Code
BCA-603

Select Date --> 16/05/2023

Status	Roll Number	Student Name
<input checked="" type="checkbox"/>	BCA2020001	Ravi Shankar
<input checked="" type="checkbox"/>	BCA2020002	Aditya Prakash
<input checked="" type="checkbox"/>	BCA2020003	Esha Solomon
<input checked="" type="checkbox"/>	BCA2020005	Sourav Kumar
<input type="checkbox"/>	BCA2020006	Preety Kumari
<input type="checkbox"/>	BCA2020007	Ved Prakash
<input type="checkbox"/>	BCA2020008	Kunal Kumar
<input checked="" type="checkbox"/>	BCA2020010	Rohit Kumar
<input checked="" type="checkbox"/>	BCA2020054	Archana Kumari
<input type="checkbox"/>	BCA2020060	Akash Kumar

Submit Back

Upload Marks Page1:

Department
Select

Year
Select

Section
Select

Search

Upload Marks Page2:

2:08 Vol 4G all 89%

☰ AMS Teacher Dashboard

Welcome : MR.
PRAKASH UPADHYAY

Subject Code
Select

Exam
Select

Total Marks

Roll Number	Student Name	Marks
BCA2020001	Ravi Shankar	<input type="text"/>
BCA2020002	Aditya Prakash	<input type="text"/>
BCA2020003	Esha Solomon	<input type="text"/>
BCA2020005	Sourav Kumar	<input type="text"/>
BCA2020006	Preety Kumari	<input type="text"/>
BCA2020007	Ved Prakash	<input type="text"/>
BCA2020008	Kunal Kumar	<input type="text"/>
BCA2020010	Rohit Kumar	<input type="text"/>

Update Password Page:

2:15 Vol 4G all 88%

☰ AMS Teacher Dashboard

Welcome : MR.
PRAKASH UPADHYAY

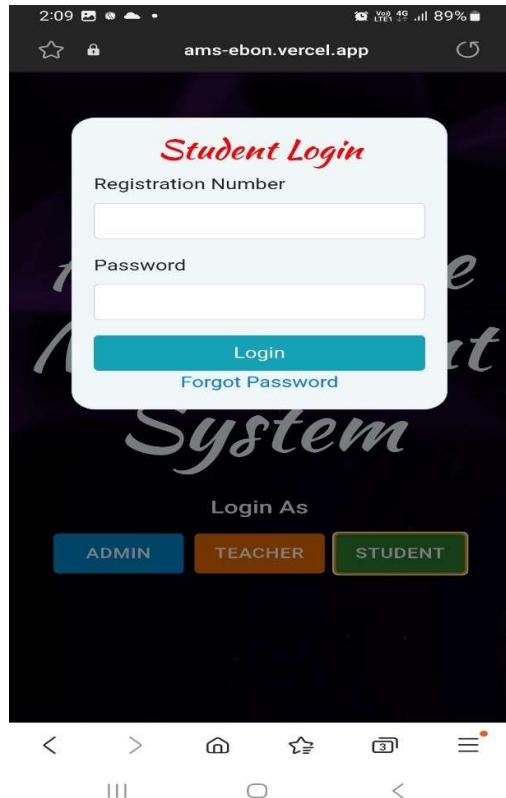
Old Password

New Password

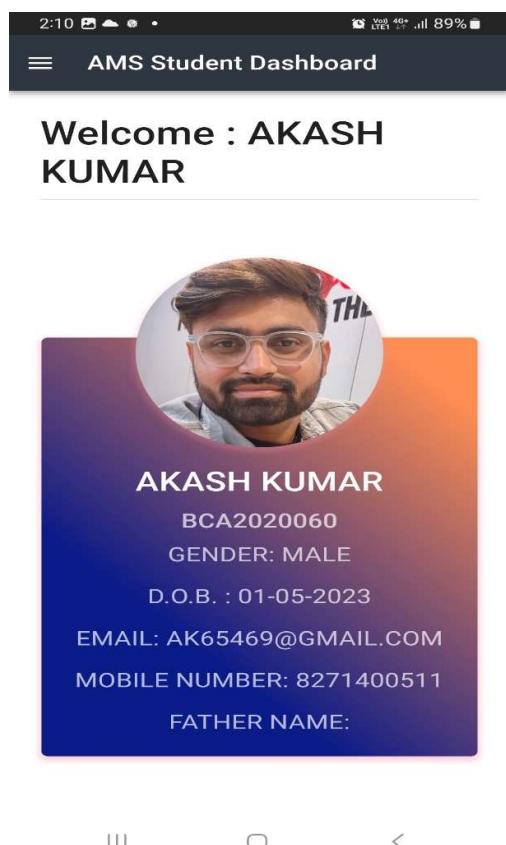
Confirm New Password

Update Password

Student Login Page:



Student Dashboard:



Update Profile Page:

AMS Student Dashboard

Welcome : AKASH KUMAR

Profile Picture

No file chosen

Gender

Select

Contact Number

Father Name

Father Contact Number

College Roll Number

BCA2020060

Results Page:

AMS Student Dashboard

Welcome : AKASH KUMAR

InternalAssessment

S.No	Subject Code	Subject Name	Obtained Marks	Total Marks
1	BCA-501	VB.net	38	40
2	BCA-502	Graphics	38	40
3	BCA-503	Computer Network	36	40
4	BCA-504	Accounts	37	40

Practical 1

S.No	Subject Code	Subject Name	Obtained Marks	Total Marks
1	BCA-501	VB.net	21	40
2	BCA-502	Graphics	22	40

Results Page:

The screenshot shows the AMS Student Dashboard with a welcome message for 'AKASH KUMAR'. Below it, there are two tables: one for Internal Assessment and one for External Assessment.

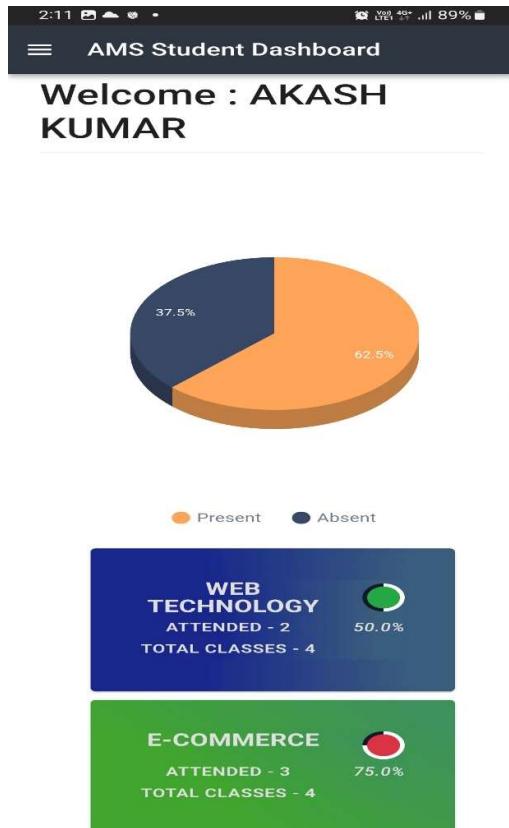
Internal Assessment

Subject Name	Obtained Marks	Total Marks	Percentage
VB.net	38	40	95.00%
Graphics	38	40	95.00%
Computer Network	36	40	90.00%
Accounts	37	40	92.50%

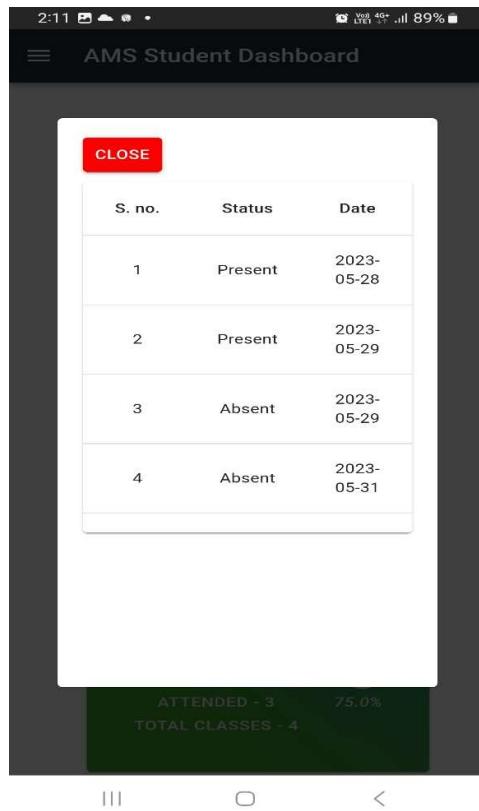
External Assessment

Subject Name	Obtained Marks	Total Marks	Percentage
VB.net	21	40	52.50%
Graphics	22	40	55.00%

Attendance Page1:



Attendance Page2:



Subject List Page:

The screenshot shows the 'AMS Student Dashboard' with a header bar at the top displaying the time '2:12', signal strength, battery level at 89%, and the text 'AMS Student Dashboard'.

Below the header, a welcome message reads 'Welcome : AKASH KUMAR'.

At the bottom of the screen is a table listing subjects:

S.No	Subject Code	Subject Name	Year	Total Lectures
1	BCA-501	VB.net	3	20
2	BCA-502	Graphics	3	20
3	BCA-503	Computer Network	3	25
4	BCA-504	Accounts	3	20
5	BCA-601	Web Technology	3	25
6	BCA-603	E-Commerce	3	20

Update Password Page:

The screenshot shows a mobile application interface. At the top, there is a dark header bar with the text "AMS Student Dashboard" and a menu icon. Below the header, the main content area displays a welcome message: "Welcome : AKASH KUMAR". Underneath this, there are three input fields labeled "Old Password", "New Password", and "Confirm New Password", each with a corresponding empty text input box. Below these fields is a teal-colored button labeled "Update Password".

Old Password

New Password

Confirm New Password

Update Password

||| ○ <

SOURCE CODE

For source code scan this QR code:



LINK TO AMS

Scan the Code or follow the Link



<https://ams-ebon.vercel.app>

FUTURE SCOPE

8. Scope for future development: -

The project has a very vast scope in future. The project can be implemented on intranet in future. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. With the proposed software of database Space Manager ready and fully functional the client is now able to manage and hence run the entire work in a much better, accurate and error free manner. The following are the future scope for the project.

- Discontinue of particular student eliminate potential attendance.
- Bar code Reader based attendance system.
- Individual Attendance system with photo using Student login.

LIMITATIONS

1. Inconsistency in data entry that could generate errors.
2. System is fully dependent on skilled individuals.
3. Time consuming and costly to produce reports.
4. Entry of false information.
5. Duplication of data entry.
6. Admin doesn't have the flexibility to use existing file to update the existing database.
7. Admin module lacks the features for insertion, deletion or updating of multiple entries.
8. Admin cannot monitor the actions of other users.
9. Attendance cannot be marked according to the timetable.
10. Unable to generate summarized reports for attendance and marks.
11. Teachers do not have the feature of updating or deleting attendance or marks.
12. User cannot generate new password if forgot the older one (they need to request the superuser for retrieval of old password.)
13. Lack of security features.

CONCLUSION

Being the Final year students of Computer Application , we were able to implement what we learnt in our web programming, System Analysis & Software engineering class into practice. We were able to attain our set objectives, and this helped us gain confidence in writing our own code and our own applications.

We worked as a team, and gained some experience on how professional programmers work in the industry.

There is always room for improvement, and this application we created can also be improved. This is especially because we had to create it within a limited time due to examination. With more time, the software can be vast and improved to include security and different types of users. Thanks a lot.

BIBLIOGRAPHY

1. www.stackoverflow.com
2. www.mui.com
3. www.getbootstrap.com
4. www.npmjs.com
5. www.gravatar.com
6. www.cyclic.sh
7. www.vercel.com
8. www.codesandbox.io
9. <https://www.udemy.com/course/the-complete-web-development-bootcamp/>
10. <https://youtube.com/@codewithyousaf>
11. <https://youtube.com/@DeepakSingh-sy4ws>

