

## **ABSTRACT**

Student Management System provides the benefits of streamlined management operations, enhanced administrations & control, strict cost control, and improved student development. SMS is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to Educational Institutions. More importantly, it is backed by reliable support.

The Project 'Student Management System' is focused on provide a facility of keeping a track of students' score and attendance throughout his course. A three user based configuration of the application allows an Admin to manage student, faculty and subject records while a faculty can manage attendance and score of the student they teach. Application all provides a feature of automatic calculation of average marks. User Interface and all the features off application have been particularly built to just fit requirements of the educational institution.

## **INTRODUCTION**

Student Management Systems are a necessity of today's educational institutions. Student Management System, as we all know is a massive integrated system that supports the comprehensive information of educational institutions including administrative, student, and faculty information such as student's USN, Name, Date of Birth, Department, Score, Attendance, etc. with other relevant records of faculty and subjects. The Functionality of the system differs with every user's needs and operations with less Human Error. The strength and Strain of manual Labour can be reduced. High Security and Data Redundancy can be avoided to some extent with data consistency and ease to handle.

This Student Management System provides the functionality to add students, faculty, and courses in the database by the database administrator. They can also assign subjects to faculties and students respectively. Faculties can update scores and attendance for students in assigned subjects. Students can track their attendance and scores for enrolled subjects.

In the Project, an entity relation diagram shows the entities in the database and relationships between the tables within the database. The ER diagram is then converted to a schema, to represent the logical view of the entire database. The student database consists of tables, namely administrator, assigned, attendance, enrolled, faculty, marks, student, subject and users, and marksTrigger (Trigger). Relations are made between the entities using primary key constraints and foreign key constraints.

### **1.1 Database Management System (DBMS)**

Following the technology progress in the areas of processors, computer memory, computer storage, and computer networks, the sizes, capabilities, and performance of databases and their respective DBMSs have grown in orders of magnitude. The development of database technology can be divided into three eras based on data model or structure: navigational, SQL relational, and post-relational. The two main early navigational data models were the hierarchical model, epitomized by IBM's IMS system, and the CODASYL model (network model), implemented in a number of products such as IDMS.

The relational model employs sets of ledger-style tables, each used for a different type of entity. Only in the mid-1980s did computing hardware become powerful enough to allow the wide deployment of relational systems (DBMSs plus applications). By the early 1990s, however, relational systems dominated in all large-scale data processing applications, and as of 2015 they remain dominant:

IBM DB2, Oracle, MySQL, and Microsoft SQL Server are the top DBMS. The dominant database language, standardized SQL for the relational model, has influenced database languages for other data models.

### 1.1.1 JAVA

**Java** was developed by James Gosling at **Sun Microsystems Inc.** in the year **1991**, later acquired by Oracle Corporation. It is a simple programming language. Java makes writing, compiling, and debugging programming easy. It helps to create reusable code and modular programs.

Java is a class-based, object-oriented programming language and is designed to have as few implementation dependencies as possible. A general-purpose programming language made for developers to *write once run anywhere* that is compiled Java code can run on all platforms that support Java. Java applications are compiled to byte code that can run on any Java Virtual Machine. The syntax of Java is similar to C/C++.

### 1.1.2 Swing

Java Swing is a part of Java Foundation Classes (JFC) which was designed for enabling large-scale enterprise development of Java applications. Java Swing is a set of APIs that provides the graphical user interface (GUI) for Java programs. Java Swing is also known as the Java GUI widget toolkit. Java Swing or Swing was developed based on earlier APIs called Abstract Windows Toolkit (AWT). Swing provides richer and more sophisticated GUI components than AWT. The GUI components are ranging from a simple label to a complex tree and table. Besides emulating the look and feel of various platforms, Swing also provides a *pluggable look and feel* to allow the look and feel of Java programs independent from the underlying platform.

### 1.1.3 Java Database Connectivity (JDBC)

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database. There are four types of JDBC drivers:

- JDBC-ODBC Bridge Driver,
- Native Driver,
- Network Protocol Driver, and
- Thin Driver

We can use JDBC API to access tabular data stored in any relational database. With the help of JDBC API, we can save, update, delete and fetch data from the database. It is like Open Database Connectivity (ODBC) provided by Microsoft.

Before JDBC, ODBC API was the database API to connect and execute the query with the database. But ODBC API uses ODBC driver which is written in C language (i.e. platform-dependent and unsecured). That is why Java has defined its own API (JDBC API) that uses JDBC drivers (written in Java language). We can use JDBC API to handle database using Java program and can perform the following activities:

1. Connect to the database
2. Execute queries and update statements to the database
3. Retrieve the result received from the database.

#### **1.1.4 MySQL**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed, and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons –

- MySQL is released under an open-source license. So, you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

## **OBJECTIVES**

- ❖ The objective of the project is to develop a system that automates the processes and activities of the Education Institution.
- ❖ The purpose is to design a system using which one can perform all operations related to student admission, attendance, and score records.
- ❖ Design a system for better student progress tracking.
- ❖ Reduce educational institutions' operating costs.
- ❖ Better coordination within different departments.
- ❖ Provide management information system report on demand to management for better decision making.

## **2. REQUIREMENTS**

### **Functional Requirements**

- Register new students.
- Record the attendance of students.
- Record the marks of students.
- Register a new faculty.
- Record the course details and subject information.
- Generate CSV files for all result sets in the system.

### **Non-Functional Requirements**

In this system, the authentication of the user is an important factor. In this system, user authentication will be done by login by user name and password and classified by user type. Users will get access to the system as permissions are classified for that type of user.

The system has a consistent interface so that the system is easy to use and in the interface of our system buttons and forms are used to enter data related to a specific module.

### **Software Requirements**

- Operating System: Windows Vista/7/8/10
- Front End: Java, Java Swing
- Rear End: MySQL Other
- Tools Used: IntelliJ Idea, JDBC Connector, MySQL Workbench

## **3. SYSTEM DESIGN**

### **3.1 Introduction**

Systems design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development.

This Project is implemented using IntelliJ Idea, which is proven to be a very efficient tool in the implementation of Java Projects. And styling is done using Java programming language and swing tools. It is done under Windows10 platform. Database Interface to the program is provided with the help of MySQL Database. JDBC connectors are used to provide connectivity between Java Code and MySQL.

### **SOFTWARE DESIGN**

#### **3.2 ER Diagram**

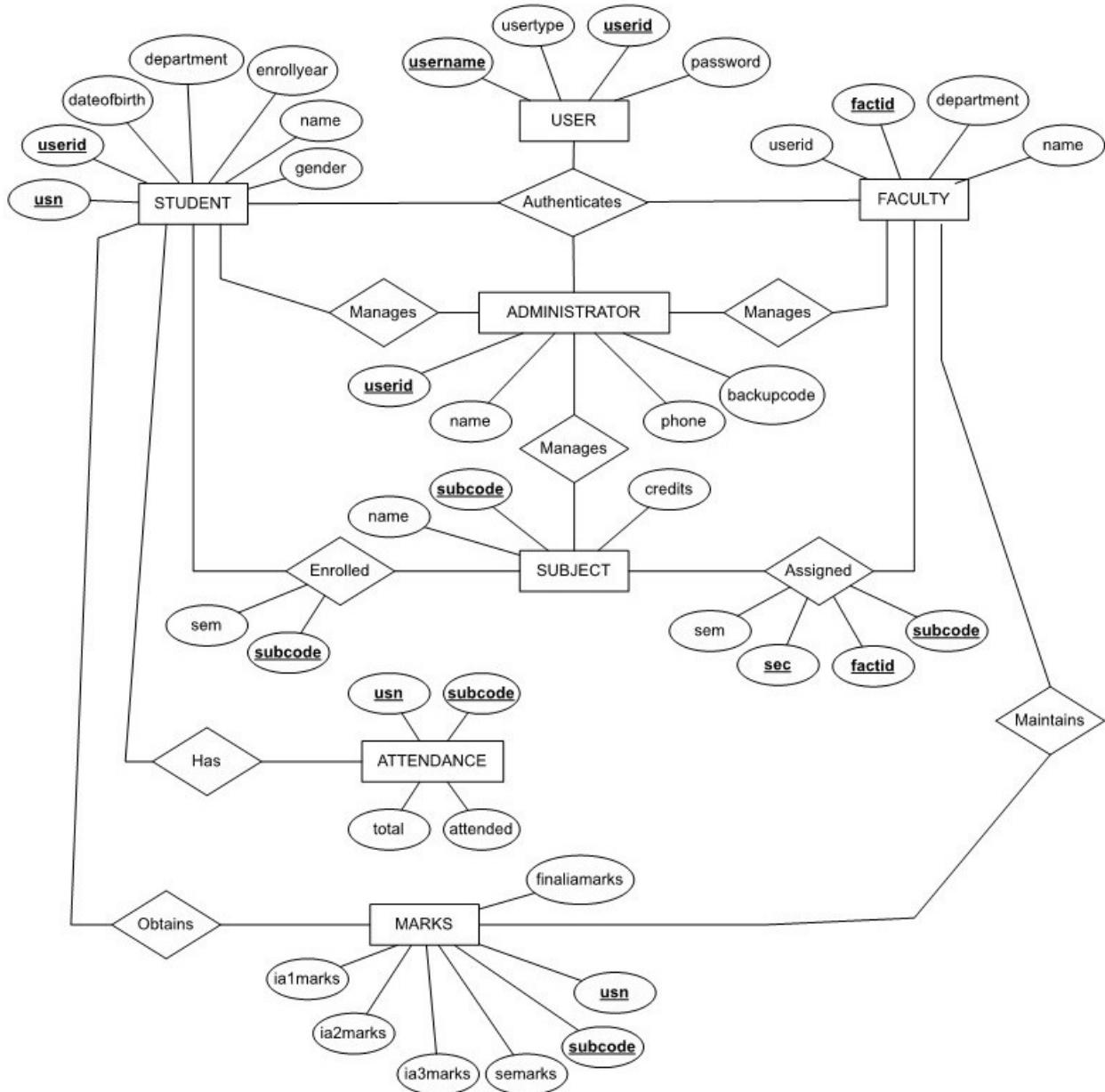
An entity–relationship model or the ER Diagram describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types and specifies relationships that can exist between instances of those entity types.

In software engineering an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure that can be implemented in a database, typically a relational database.

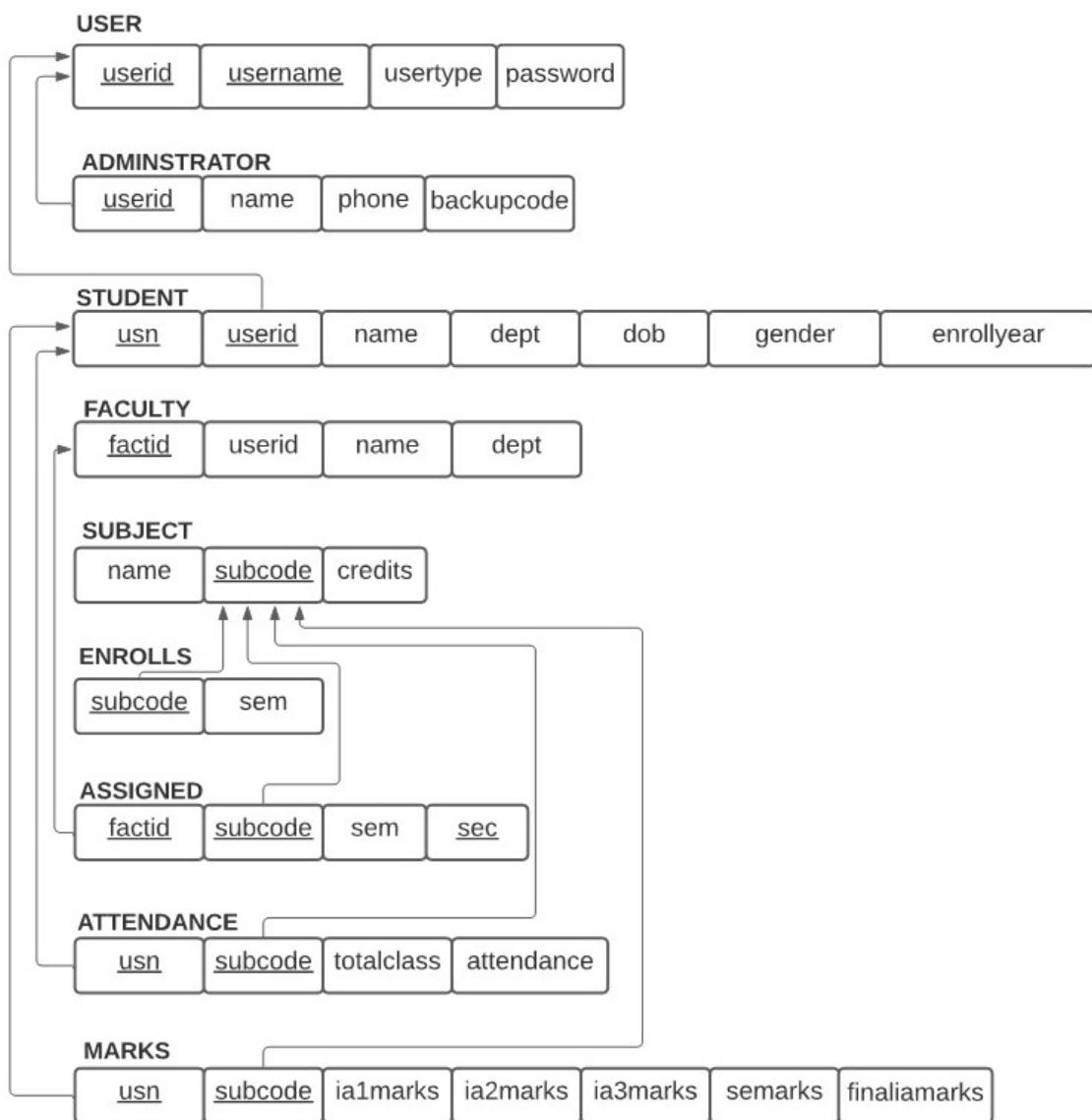
#### **3.3 Schema Diagram**

The schema diagram of a database system is its structure described in a formal language supported by the database management system (DBMS). The formal definition of a database schema is a set of formulas called integrity constraints imposed on a database.

The term "schema" refers to the organization of data as a blueprint of how the database is constructed. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language. The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modelled in the database.

**ER DIAGRAM**

## SCHEMA DIAGRAM



## 4. SYSTEM IMPLEMENTATION

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MySQL database has been chosen for developing the relevant databases.

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
PASSWORD	varchar(30)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
USERID	int		NO			select,insert,update,references
USERNAME	varchar(30)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references
USERTYPE	varchar(10)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references

**Fig 4.1 User Table:** This table stores the user details.

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
BACKUPCODE	varchar(10)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
NAME	varchar(30)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
PHONE	varchar(10)		YES	utf8mb4	utf8mb4_0900_...	select,insert,update,references
USERID	int		NO			select,insert,update,references

**Fig 4.2 Administrator table:** This table stores the administrator details.

studentmanagement.faculty						
Info	Columns	Indexes	Triggers	Foreign keys	Partitions	Grants
	Column	Type	Default Value	Nullable	Character Set	Collation
	DEPARTMENT	varchar(50)		YES	utf8mb4	utf8mb4_0900_...
	FACTID	varchar(15)		NO	utf8mb4	utf8mb4_0900_...
	NAME	varchar(30)		YES	utf8mb4	utf8mb4_0900_...
	USERID	int		NO		select,insert,update,references

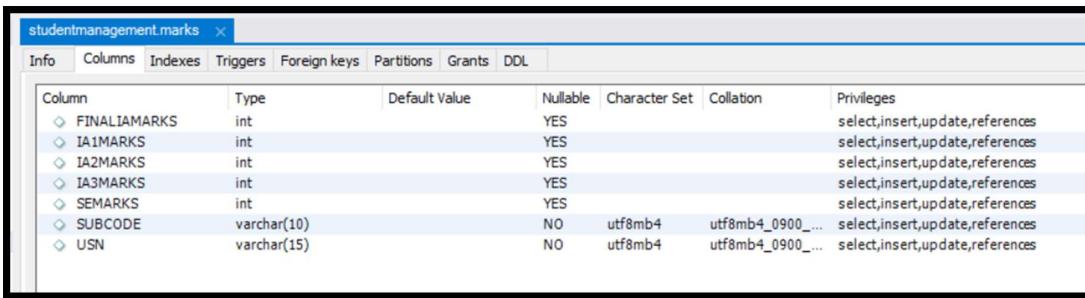
**Fig 4.3 Faculty table:** This table stores the faculty details.

studentmanagement.student						
Info	Columns	Indexes	Triggers	Foreign keys	Partitions	Grants
	Column	Type	Default Value	Nullable	Character Set	Collation
	DATEOFBIRTH	date		YES		select,insert,update,references
	DEPARTMENT	varchar(50)		YES	utf8mb4	utf8mb4_0900_...
	ENROLLYEAR	int		YES		select,insert,update,references
	GENDER	varchar(8)		YES	utf8mb4	utf8mb4_0900_...
	NAME	varchar(30)		YES	utf8mb4	utf8mb4_0900_...
	SEC	varchar(1)		YES	utf8mb4	utf8mb4_0900_...
	SEM	int		YES		select,insert,update,references
	USERID	int		NO		select,insert,update,references
	USN	varchar(15)		NO	utf8mb4	utf8mb4_0900_...

**Fig 4.4 Student table:** This table stores the student details.

studentmanagement.subject						
Info	Columns	Indexes	Triggers	Foreign keys	Partitions	Grants
	Column	Type	Default Value	Nullable	Character Set	Collation
	CREDITS	int		YES		select,insert,update,references
	NAME	varchar(60)		YES	utf8mb4	utf8mb4_0900_...
	SUBCODE	varchar(10)		NO	utf8mb4	utf8mb4_0900_...

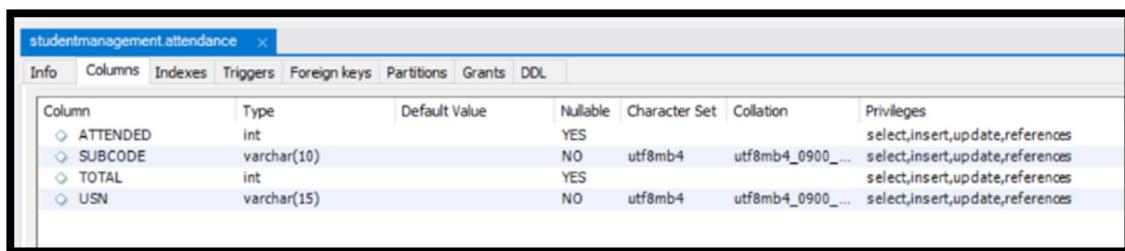
**Fig 4.5 Subject table:** This table stores information about the subjects.



The screenshot shows the 'studentmanagement.marks' table structure. The table has seven columns: FINALIAMARKS, IA1MARKS, IA2MARKS, IA3MARKS, SEMARKS, SUBCODE, and USN. The 'Info' tab is selected. The table structure is as follows:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
FINALIAMARKS	int		YES			select,insert,update,references
IA1MARKS	int		YES			select,insert,update,references
IA2MARKS	int		YES			select,insert,update,references
IA3MARKS	int		YES			select,insert,update,references
SEMARKS	int		YES			select,insert,update,references
SUBCODE	varchar(10)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references
USN	varchar(15)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references

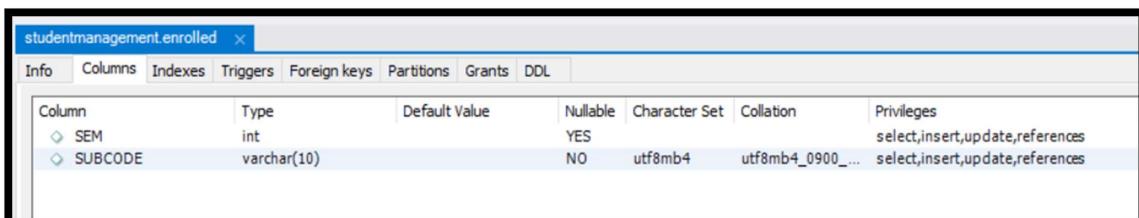
**Fig 4.6 Marks table:** This table stores the marks obtained by the students in a particular subject in all examination taken by them during the semester.



The screenshot shows the 'studentmanagement.attendance' table structure. The table has four columns: ATTENDED, SUBCODE, TOTAL, and USN. The 'Info' tab is selected. The table structure is as follows:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
ATTENDED	int		YES			select,insert,update,references
SUBCODE	varchar(10)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references
TOTAL	int		YES			select,insert,update,references
USN	varchar(15)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references

**Fig 4.7 Attendance table:** This table stores the attendance of the students for the subjects they are enrolled in.



The screenshot shows the 'studentmanagement.enrolled' table structure. The table has three columns: SEM, SUBCODE, and USN. The 'Info' tab is selected. The table structure is as follows:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
SEM	int		YES			select,insert,update,references
SUBCODE	varchar(10)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references

**Fig 4.8 Enrolled table:** This table stores the semester and section of students

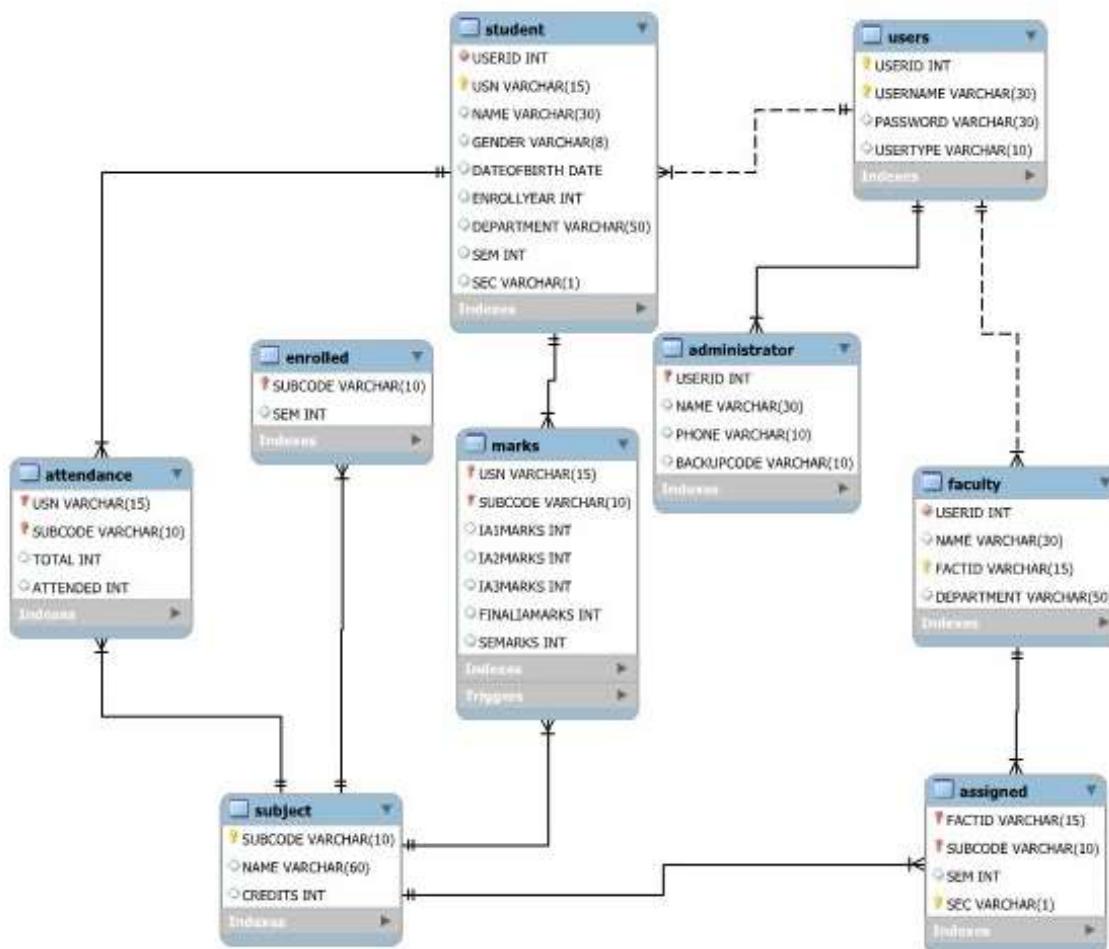
Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
FACTID	varchar(15)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references
SEC	varchar(1)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references
SEM	int		YES			select,insert,update,references
SUBCODE	varchar(10)		NO	utf8mb4	utf8mb4_0900_...	select,insert,update,references

**Fig 4.9 Assigned table:** This table stores the data of the subjects assigned to faculties for specific semesters and sections.

```
-- TRIGGER: To Update Final IA Marks when all three IA Marks are available
DROP TRIGGER IF EXISTS MARKSTRIGGER;
Delimiter //
CREATE TRIGGER MARKSTRIGGER
BEFORE UPDATE ON MARKS
FOR EACH ROW
BEGIN
IF (NEW.IA1MARKS IS NOT NULL AND NEW.IA2MARKS IS NOT NULL AND NEW.IA3MARKS IS NOT NULL) THEN
SET NEW.FINALIAMARKS = (NEW.IA1MARKS + NEW.IA2MARKS + NEW.IA3MARKS)/3;
END IF;
END;
//
Delimiter ;
```

**Fig 4.10 Trigger:** MarksTrigger is used for auto assigning average marks when all three IA Marks are Available.

## RELATION BETWEEN THE TABLES



### Description:

A Database named '**STUDENTMANAGEMENT**' has been used to store data of Student Management System. It has nine tables to fulfill the requirement of efficient storage of data within the database.

**User Table:** It stores User ID, Username, Password and User Type of a user and has been used for the purpose of authentication from the login page. The Default username and password set for the administrator is 'admin' and 'admin' respectively. UserId is a primary key for this table.

**Administrator Table:** It stores User ID, Name, Phone and Backup Code of the administrator and it has been used to store details required to reset the password for administrator. UserId is primary key as well as a foreign key referenced from the 'User' Table.

**Faculty Table:** It stores User ID, Name, Faculty ID and Department of the faculty. FactId is a primary key and UserId is a foreign key referenced from 'User' Table.

**Student Table:** It stores User ID, USN, Name, Gender, DateOfBirth, EnrollYear, Department, Semester and Sec of a Student. USN is the primary Key and UserId is a foreign key referenced from 'User' Table.

**Subject Table:** It stores Subject Code, Name and Credit of particular subject. SubCode is primary key for this table.

**Marks Tables:** It stores USN, Subject Code, IA1Marks, IA2Marks, IA3Marks, FinalIAMarks and SEMarks of the student. USN and SubCode are primary key together and where USN is a foreign key referenced from Student Table and SubCode is a foreign key referenced from Subject Table.

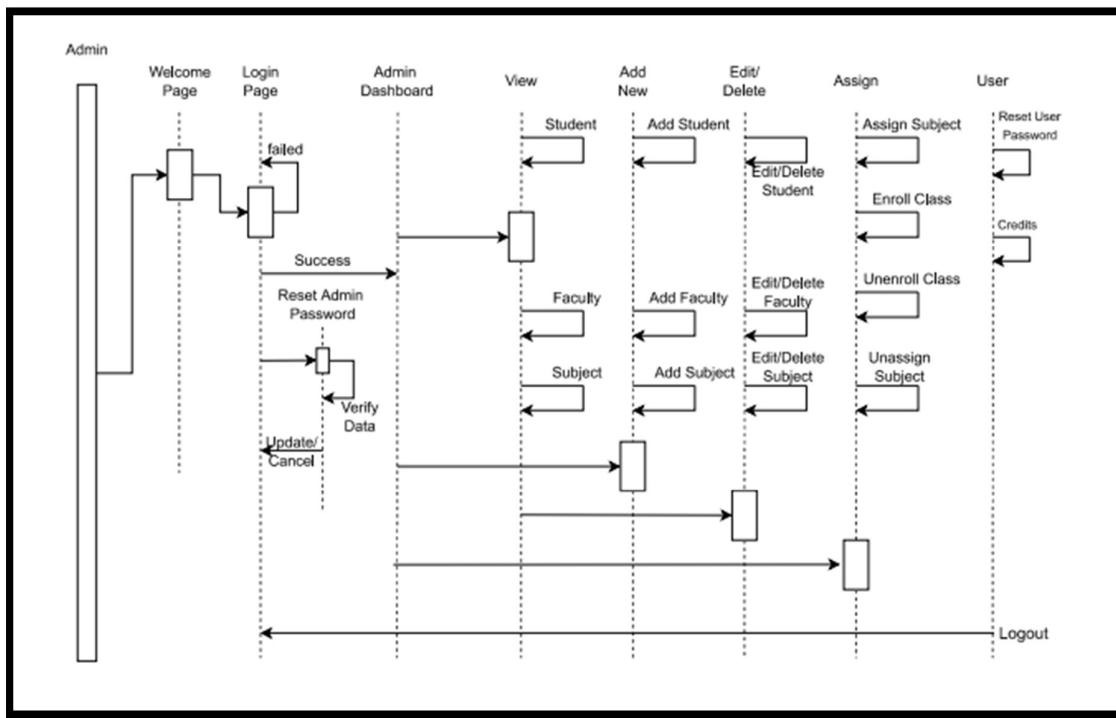
**Attendance Table:** It stores USN, Subject code, Total and Attended fields to store number of classes attended out of total classes conducted in a particular subject for a student. USN and SubCode are primary key together and where USN is a foreign key referenced from Student Table and SubCode is a foreign key referenced from Subject Table.

**Enrolled Table:** It stores subject code and semester to keep a track of which subject is being taught to which semesters. SubCode is Primary key as well as Foreign Key referenced from Subject Table.

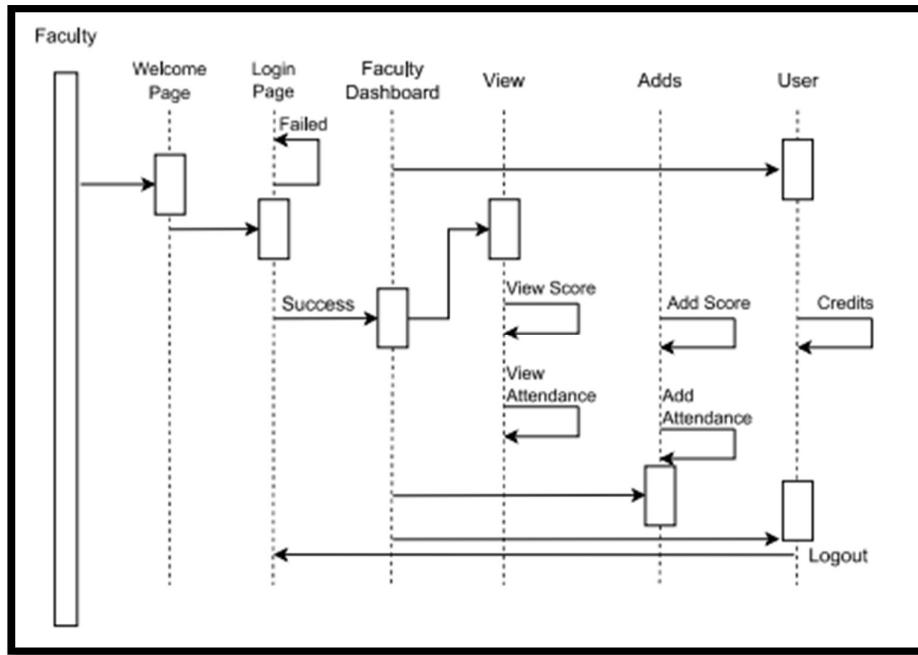
**Assigned Table:** It stores Faculty ID, Subject Code, Semester and Section to keep a track of which subject is assigned to which faculty for a particular class and section. Primary Key for this table consists of FactId, SubCode and Sec where SubCode is a Foreign Key referenced from Subject Table and FactID is a Foreign Key referenced from Faculty Table.

**Mark Trigger:** It is a Trigger applied on Marks Table. It checks before every update and for each row it sets the Final IA Marks to the average of IA 1 Marks, IA 2 Marks and IA 3 Marks if all of them are available in the database.

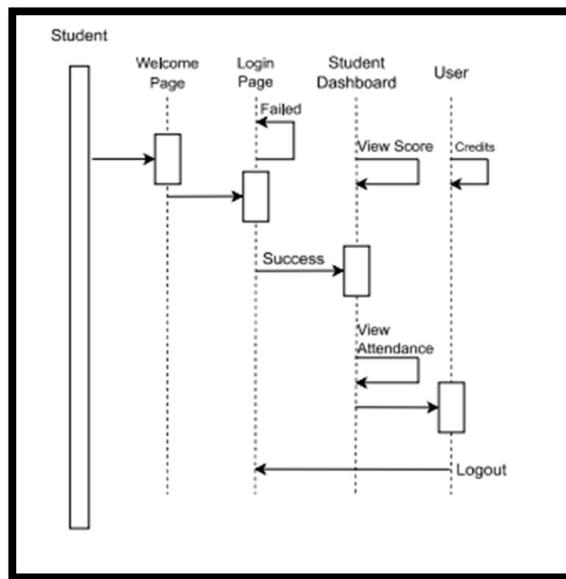
## SEQUENCE DIAGRAM



**Administrator:** The above sequence diagram shows the interaction of admin with the system in the time sequence as available in the software.



**Faculty:** The above sequence diagram shows the interaction of faculty with the system in the time sequence as available in the software.



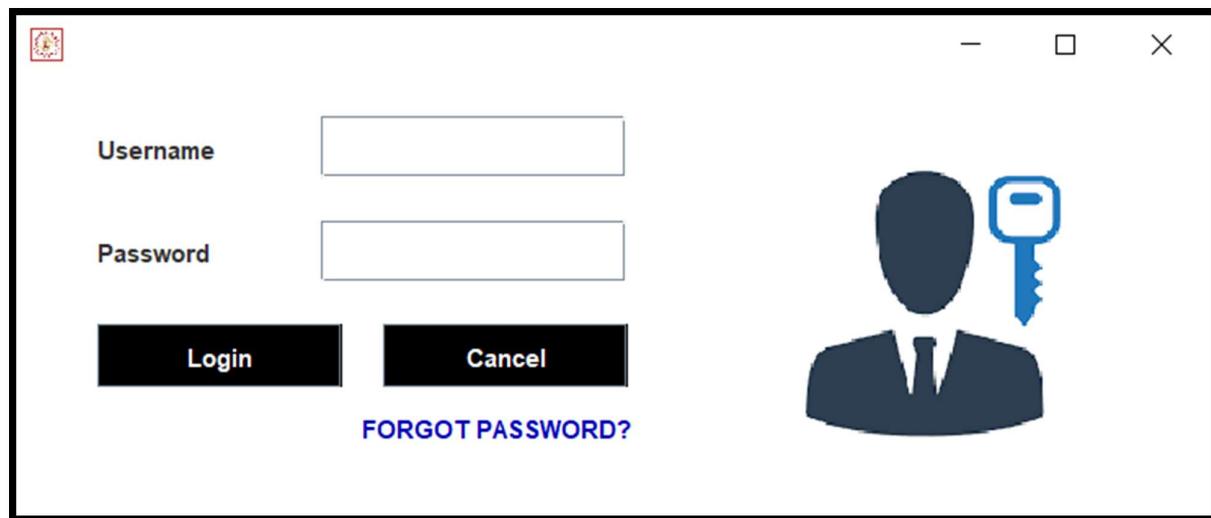
**Student:** The above sequence diagram shows the interaction of student with the system in the time sequence as available in the software.

## **5. RESULTS**

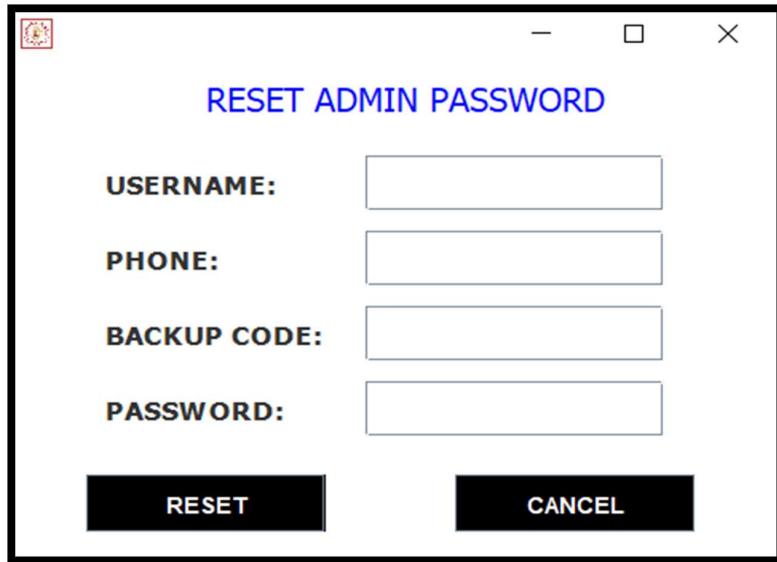
### **WELCOME AND LOGIN SECTION:**



**Fig 5.1 The Main Page:** This is welcome page of the Application



**Fig 5.2 Login Page:** This page provides the login facility to admin, faculty and students for logging in to their accounts.



A screenshot of a Windows-style dialog box titled "RESET ADMIN PASSWORD". The dialog contains four input fields labeled "USERNAME:", "PHONE:", "BACKUP CODE:", and "PASSWORD:". Below the fields are two buttons: "RESET" on the left and "CANCEL" on the right.

**Fig 5.3 Reset Admin Password:** This page helps to reset the password only for the admin with the help of the backup code available with them.

### ADMIN SECTION



**Fig 5.4 the Admin Dashboard Page:** This page provides all accessible sections to the admin.

USN	NAME	GENDER	DATE OF BIRTH	DEPARTMENT
1JS19IS006	Adwiti	Female	2000-06-17	Information Science and Engineering
1JS19IS088	Saugat	Male	2001-04-07	Information Science and Engineering
1JS19IS092	Shreya	Female	1995-01-01	Information Science and Engineering

Use Filter

**Fig 5.5 View Students:** This page is available for the admin to view the details of all students in the institution with filters of department, semester and section.

NAME	FACULTY ID	DEPARTMENT
Ramesh R S	CSE/001	Computer Science and Engineering
Rajesh K P	ISE/001	Information Science and Engineering

**Fig 5.6 View Faculty:** This page is available for the admin to view the details of all faculties in the institution with the filter of department.

SUBJECT CODE	TITLE	CREDITS
18CS42	Data Structures	4
18CS51	Management and Entrepreneurship	3
18CS52	Computer Networks and Security	4
18CS53	Database Management System	4

SAVE CSV: 

**Fig 5.7 View Subjects:** This page is available for the admin to view the subjects' details.

**ENTER STUDENT DETAILS**



NAME

BIRTHDATE  Jan 1995

USN

ENROLL YEAR

GENDER  Male  Female

USERNAME

PASSWORD

DEPARTMENT

SEMESTER

SECTION

**ADD STUDENT** **CANCEL**

**Fig 5.8 Add Students Page:** This page allows to add new students.

**ENTER FACULTY DETAILS**



NAME

FACULTY ID

DEPARTMENT

USERNAME

PASSWORD

**ADD FACULTY**    **CANCEL**

**Fig 5.9 Add Faculty Page:** This page allows to add new faculty

**ENTER SUBJECT DETAILS**



SUBJECT NAME

SUBJECT CODE

CREDITS

**ADD SUBJECT**    **CANCEL**

**Fig 5.10 Add Subjects Page:** This page allows to add new subject.

This screenshot shows a modal window titled 'Edit Student's Page'. At the top right are standard window control buttons: a minus sign for minimize, a square for maximize/minimize, and an 'X' for close. Below the title bar, there is a text input field with a placeholder 'ENTER USN:' and a 'Fetch Info' button to its right. The main body of the window contains several text input fields with labels: 'NAME:', 'DEPARTMENT:', 'GENDER:', 'SEM-SEC:', 'DATE OF BIRTH:', and 'ENROLL YEAR:'. At the bottom of the window are three buttons: 'EDIT' on the left, 'DELETE' in the middle, and 'CANCEL' on the right.

**Fig 5.11 Edit Student's Page:** This page helps to update/delete the student details after the enrolment of the students.

This screenshot shows a modal window titled 'Edit Faculty's Page'. At the top right are standard window control buttons: a minus sign for minimize, a square for maximize/minimize, and an 'X' for close. Below the title bar, there is a text input field with a placeholder 'ENTER FACULTY ID:' and a 'Fetch Info' button to its right. The main body of the window contains several text input fields with labels: 'NAME:', 'DEPARTMENT:', and 'DELETE' (which is positioned below the 'NAME:' label). At the bottom of the window are two buttons: 'DELETE' on the left and 'CANCEL' on the right.

**Fig 5.12 Edit Faculty's Page:** This page helps to delete the faculty details.

This screenshot shows a modal window titled "Edit Subject's Page". It contains fields for "ENTER SUBJECT CODE:" (with an input box), "TITLE:" (input box), and "CREDITS:" (input box). At the bottom are two buttons: "DELETE" on the left and "CANCEL" on the right.

**Fig 5.13 Edit Subject's Page:** This page helps to delete the Subjects details.

This screenshot shows a modal window titled "ASSIGN SUBJECT TO FACULTY". It contains fields for "Subject Code:" (input box), "Faculty ID:" (input box), and "Fetch Info" (button). Below these are "Subject Title:" and "Faculty Name:" labels. The "Semester-Section:" field contains two dropdown menus. At the bottom are "ASSIGN" and "CANCEL" buttons.

**Fig 5.14 Subject assignment Page:** From this page the admin can assign the subjects to selected faculty for specific semester and section.

The screenshot shows a window titled "ENROLL CLASS FOR SUBJECT". It contains three input fields: "Subject Code" with an empty text box, "Subject Title" with an empty text box, and "Semester" with a dropdown menu showing "-". Below these fields are two buttons: "ASSIGN" on the left and "CANCEL" on the right.

**Fig 5.15 Class Enrollment Page:** From this page the admin can assign the subject to a semester.

The screenshot shows a window titled "UN-ASSIGN SUBJECT TO FACULTY". It contains five input fields: "Subject Code" with an empty text box, "Faculty ID" with an empty text box, "Subject Title" with an empty text box, "Faculty Name" with an empty text box, and "Semester & Sections" with a dropdown menu showing "All". Below these fields are two buttons: "UN-ASSIGN" on the left and "CANCEL" on the right.

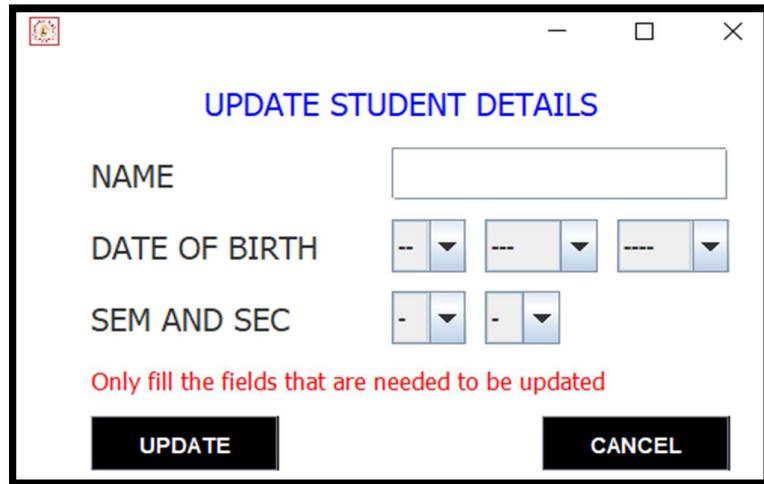
**Fig 5.16 Un-Assignment Page:** This page is used by the admin to un-assign a faculty from a particular subject for a specific section.

The window title is "UN-ENROLL CLASS FOR SUBJECT". It contains fields for "Subject Code" (with a "Fetch Info" button) and "Semester". At the bottom are "REMOVE" and "CANCEL" buttons.

**Fig 5.17 Un-Enrollment Page:** This page is used by the admin to un-enroll a semester from a particular subject.

The window title is "RESET USER PASSWORD". It has dropdowns for "USERTYPE" (set to "FACULTY") and "USERNAME", and a password field for "PASSWORD". At the bottom are "RESET" and "CANCEL" buttons.

**Fig 5.18 Reset Password:** This page is used by the admin to reset the password for students and faculty in situations when the user has forgot the password.



**Fig 5.19 Edit Students Page:** This page is used by the admin to update the student records.

### FACULTY SECTION



**Fig 5.20 The Faculty Dashboard:** This page provides all accessible sections to the faculty.

USN	NAME	IA1 MARKS	IA2 MARKS	IA3 MARKS	FINAL IA MARKS	SE MARKS
1JS19IS092	Shreya	42	45			

18CS51 SEM: 5 A  SAVE CSV:

**Fig 5.21 View Score Page:** This page is accessed by the faculty to view the given scores.

USN	NAME	ATTENDED	TOTAL
1JS19IS092	Shreya	29	40

18CS51 SEM: 5 A  SAVE CSV:

**Fig 5.22 View Attendance Page:** This page is accessed by the faculty to view the attendance of the students.

The screenshot shows a window titled "ENTER MARKS FOR STUDENTS". It contains the following fields:

- SUBJECT:** A dropdown menu showing "18CS51".
- SEM:** 5    **SEC:** A dropdown menu showing "A".
- USN:** A dropdown menu.
- NAME:** A text input field.
- IA1 MARKS:** An input field.
- IA2 MARKS:** An input field.
- IA3 MARKS:** An input field.
- SE MARKS:** An input field.

At the bottom are two buttons: "UPDATE" and "CLOSE".

**Fig 5.23 Add Marks:** This page is used by the faculty to assign marks to the students

The screenshot shows a window titled "ENTER ATTENDANCE FOR STUDENTS". It contains the following fields:

- SUBJECT:** A dropdown menu showing "18CS51".
- SEM:** 5    **SEC:** A dropdown menu showing "A".
- USN:** A dropdown menu.
- NAME:** A text input field.
- ATTENDED:** An input field.
- TOTAL:** An input field.

At the bottom are two buttons: "UPDATE" and "CLOSE".

**Fig 5.24 Attendance Page:** This page is used by the faculty to mark the attendance of the students.

## STUDENT SECTION



**Fig 5.25 The Student Dashboard:** This page provides all accessible sections to the student.

SUBCODE	SUBJECT TITLE	IA1 MARKS	IA2 MARKS	IA3 MARKS	FINAL IA MARKS	SE MARKS
18CS51	Management and Entrepreneurship	48	45	49	47	80
18CS53	Database Management System	45	43	49	46	

SAVE CSV:

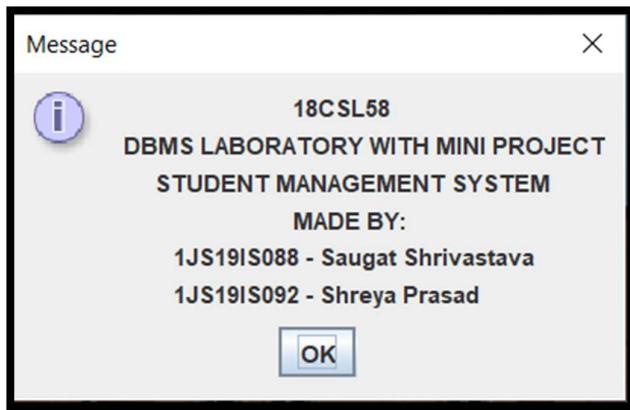
**Fig 5.26 View Scores:** This page is used by the student to view their marks obtained with respect to the subject.

SUBCODE	SUBJECT TITLE	ATTENDED	TOTAL
18CS51	Management and Entrepreneuership	39	45
18CS53	Database Management System	18	20

SAVE CSV: 

**Fig 5.27 View Attendance:** This page is used by the student to view their attendance status with respect to the subject.

#### INFORMATION DIALOG:



**Credits:** About The Developers.

## **6. CONCLUSION**

This project has been a rewarding experience in more than one way. The entire project work has enlightened us in the following areas.

- a) We have gained an insight into the working of an Education Institution. This represents a typical real-world situation.
- b) Our understanding of database design has been strengthened this is because in order to generate the final reports of database designing has to be properly followed.
- c) Scheduling a project and adhering to that schedule creates a strong sense of time management.
- d) Sense of teamwork has developed, and confidence of handling real life project has increased to a great extent.
- e) Initially, there were problem with the validation but with discussions, we were to implement validations.

➤ Limitations of the system:

- Online payment is not available at this version.
- Data delete & edit system is not available for few sections.
- User account not verified by Mobile SMS not available in this system.
- Loss of data due to mismanagement.

## **7. REFERENCES**

### **BOOKS:**

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

### **WEBSITES:**

[www.stackoverflow.com](http://www.stackoverflow.com)

[www.javatpoint.com](http://www.javatpoint.com)

[www.youtube.com](http://www.youtube.com)

[www.geeksforgeeks.com](http://www.geeksforgeeks.com)