

Final Report

(University Management System)

Course Code: CS254

Course Title: DBMS Lab

Semester: B. Tech 4th Sem

Section: S1

Academic Year: 2020-21

Course Instructor: Dr. Annappa B and

Mr. Sharath Yaji

Team Members:

1. Apurba Ranjan Mahapatra, CS109, 7752073346, apurbaranjanmahapatra.191cs109@nitk.edu.in
2. Prakhar Singh, CS144, 8127075416, prakhar.191cs144@nitk.edu.in

1 Abstract

This program is designed to digitise the functionalities of University Management System. University management is one of the best examples to explain dbms in action. This program contains most of the aspects of an university works and lets you interact with all processes in an user friendly manner.

Main Objective of this project to facilitate proper management of new entries be it a faculty recruitment or student admission through use of electronic medium. It automates the record keeping of students , Staff information , courses offered, students results, attendance, attendance report. This program, being simple in design and working, does not require much of training to users, and can be used as a powerful tool for digitise a University Management System.

In developing this database project we had to use (GUI) front end, VS Code, a powerful front-end tool is used for getting Graphical User Interface (GUI) based integrated platform and coding simplicity. For designing front end HTML, CSS, Java Script has been used, and PHP for backend MySQL is used for database as per our requirement

Key Features:

1. Login/Logout facility
2. Teacher recruitment and management detail
3. New student enrollment details
4. Student Attendance management
5. Course management
6. Student batch allotment

Software Specifications:

- Frontend: HTML, CSS, JAVA SCRIPT
- Backend: PHP
- Database: MYSQL



Figure 1: Technologies Used in this project

2 Introduction

Introduction:

Our Project titled “University Management System” gives idea about the management in University. In this package we have included all the necessary functions such as Admin login, Viewing all students, teachers, classes, subjects and attendance database. In our project admin of university can login and view all the database. Admin can take record of admission of students and he can manage the records of students. Similarly he can take record of teachers in UNIVERSITY. It can produce a detailed report of students attendance, class, sections and subjects.

We have included SQL (Structured Query Language) as a back end database as well in our project as it is a purely relational database language. Our project contains different db tables, they are students, teachers, classes, subjects, sections, attendance. Usually in an university, there are problems to keep record of all data. By looking this we have tried to solve this problem and many other potential problems efficiently. It can be made more interactive by adding many more module and gui into our project. Our project provides various services to university admin like taking student admission, cancellation of admission, enquiring about any student or teacher. Our project has a big scope in Small institutions since maintaining routing activities are very easy and efficient.

Functioning of our Project:

To take care of all problems on conventional university management system i.e manually our project proposed an efficient and reliable system. University Management System is a project to maintain all activities in an university, that are performed frequently on an university. The user of our project is

1. Admin

Log In Page: This element of our project gives an interface to login admin by entering assigned username password.

Dashboard: This element gives an interface to see details available in all tables, they are Students, Teachers, Classes, Subjects, Sections, Attendance. Admin can go into any table by clicking view button below each table.

Student Section: This section of our project keeps records of all of the students

in the university.it keeps Id,Registration number,Roll Number,Name,Photo,Class,section of a student.admin can update and delete student record.

Teacher section: It contains information about all teachers in University.Id ,name,Assigned Subjects,Class,Sections are the various parameters stored in the teacher section

Classes: This contains information about all class present in the university , it has following parameter,they are ID,Nmae,Sections,Class Teacher.Admin can update , delete and add new classes.

Subjects: In this Section of our project contains all the subjects that are offered by university.it contains parameters ,they are ID,subject,Codeand Subject Type.here Admin can add new data and can update and delete existing data.

Sections: This keeps record of all present section of a class. Parameters are ID,Name..Here Admin can add new data and can update and delete existing data.

Attendance: In this section our project asks 1st to give input of class section.

After this it will show all atdata to Admin.

Functionality of the current project can be easily extended according to the needs.However,in its current state it can serve small functionality of a university.

3 ER Diagram

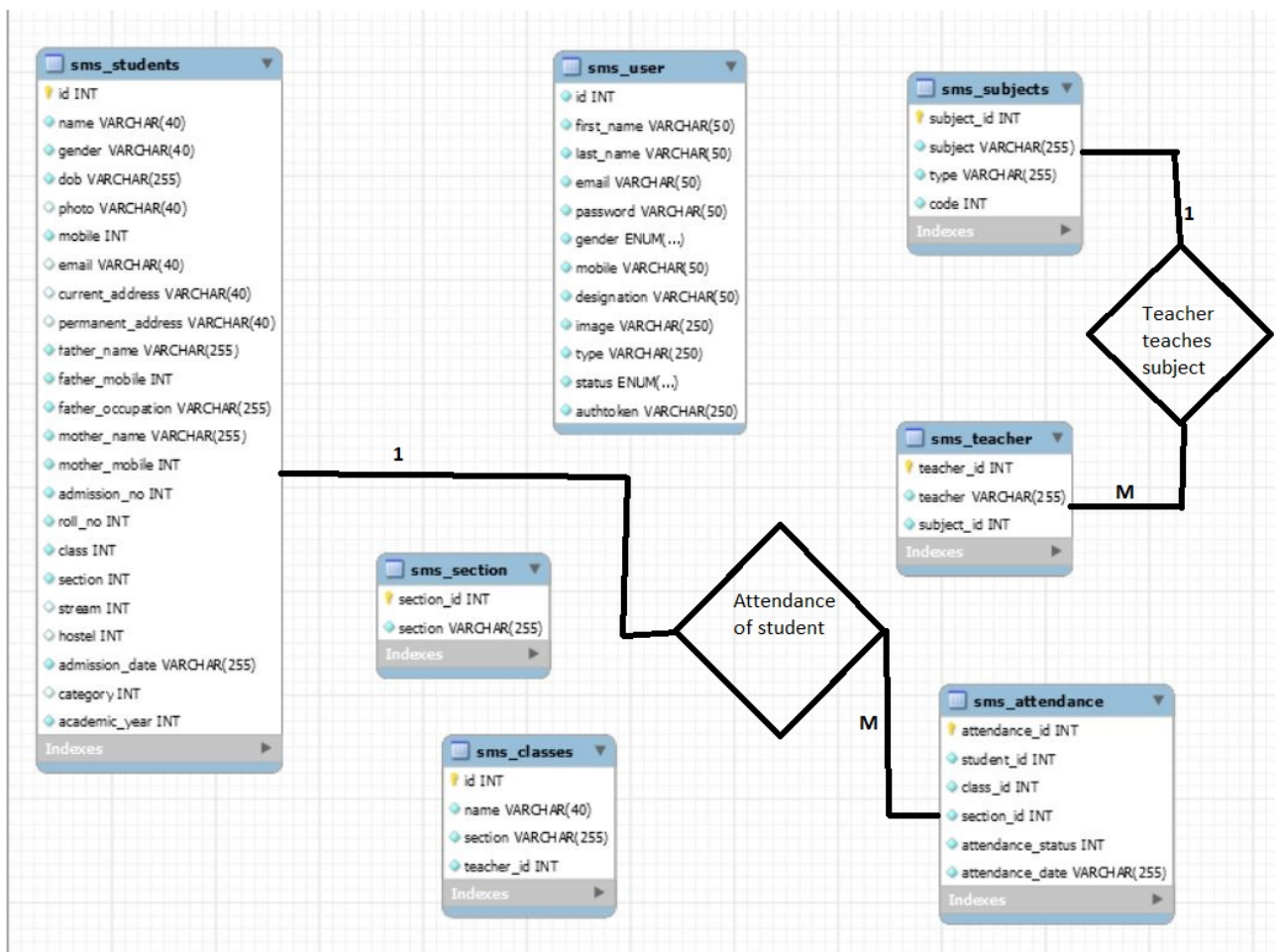


Figure 2: ER Diagram

4 Source Code

Front-end: In this project we have used HTML CSS JAVA SCRIPT for front end designing.

```
;!DOCTYPE html;
;html;
;head;
;meta http-equiv="Content-Type" content="text/html; charset=utf-8" /;
;link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css";
;link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap-
theme.min.css";
;script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js";/script;
;script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js";/script;
;!-- jQuery --;

; /head;
;body class="" ;
;div role="navigation" class="navbar navbar-default navbar-static-top" style="background:
black;" ;
;div class="container" ;
;div class="navbar-header" ;
;button data-target=".navbar-collapse" data-toggle="collapse" class="navbar-toggle" type="button" ;
;span class="sr-only";Toggle navigation;/span;
;span class="icon-bar";/span;
;span class="icon-bar";/span;
;span class="icon-bar";/span;
;/button;
;h3 class="navbar-brand";NIT-K;/h3;
;h3 style="color:white;text-align:center; margin:20px 200px";College Management System;/h3;
;/div;
;div class="navbar-collapse collapse" ;

; /div;!--/.nav-collapse --;

;/div;
```

;/div;

`<div class="container" style="min-height:500px;">`

Back End Database: For database we have use MYSQL

```
CREATE TABLE university_students(  
  id int(10) UNSIGNED NOT NULL,  
  name varchar(40) NOT NULL,  
  gender varchar(40) NOT NULL,  
  dob varchar(255) NOT NULL,  
  photo varchar(40) DEFAULT NULL,  
  mobile int(10) UNSIGNED NOT NULL,  
  email varchar(40) DEFAULT NULL,  
  current_address varchar(40) DEFAULT NULL,  
  permanent_address varchar(40) DEFAULT NULL,  
  father_name varchar(255) NOT NULL,  
  father_mobile int(10) UNSIGNED NOT NULL,  
  father_occupation varchar(255) NOT NULL,  
  mother_name varchar(255) NOT NULL,  
  mother_mobile int(10) UNSIGNED NOT NULL,  
  admission_no int(11) NOT NULL,  
  roll_no int(11) NOT NULL,  
  class int(10) UNSIGNED NOT NULL,  
  section int(11) NOT NULL,  
  stream int(10) UNSIGNED DEFAULT NULL,  
  hostel int(10) UNSIGNED DEFAULT NULL,  
  admission_date varchar(255) NOT NULL,  
  category int(10) UNSIGNED DEFAULT NULL,  
  academic_year int(10) UNSIGNED NOT NULL  
);
```

```
ALTER TABLE university_students  
ADD PRIMARY KEY(id);
```

```

CREATE TABLE university_attendance(
attendance_idint(11)NOTNULL,
student_idint(11)NOTNULL,
class_idint(11)NOTNULL,
section_idint(11)NOTNULL,
attendance_statusint(11)NOTNULL,
attendance_datevarchar(255)NOTNULL);

```

```

ALTER TABLE university_attendance
ADDPRIMARYKEY(attendance_id);

```

```

CREATE TABLE university_classes(
idint(10)UNSIGNEDNOTNULL,
namevarchar(40)NOTNULL,
sectionvarchar(255)NOTNULL,
teacher_idint(11)NOTNULL);

```

```

ALTER TABLE university_classes
ADDPRIMARYKEY(id);

```

```

CREATE TABLE university_section(
section_idint(11)NOTNULL,
sectionvarchar(255)NOTNULL);

```

```

ALTER TABLE university_section
ADDPRIMARYKEY(section_id);

```

```

CREATE TABLE university_subjects(
subject_idint(11)NOTNULL,
subjectvarchar(255)NOTNULL,
typevarchar(255)NOTNULL,
codeint(11)NOTNULL
);

```

```

ALTERTABLEuniversity_subjects
ADDPRIMARYKEY(subject_id);

```



```
CREATE TABLE university_teacher(
teacher_idint(11)NOTNULL,
teachervarchar(255)NOTNULL,
subject_idint(11)NOTNULL);
```

```
ALTER TABLE university_teacher
ADDPRIARYKEY(teacher_id);
```

```
CREATE TABLE university_user(
idint(11)NOTNULL,
first_namevarchar(50)NOTNULL,
last_namevarchar(50)NOTNULL,
emailvarchar(50)NOTNULL,
passwordvarchar(50)NOTNULL,
genderenum('male','female')CHARACTERSETutf8NOTNULL,
mobilevarchar(50)NOTNULL,
designationvarchar(50)NOTNULL,
imagevarchar(250)NOTNULL,
typevarchar(250)NOTNULLDEFAULT'general',
statusenum('active','pending','deleted','")NOTNULLDEFAULT'pending',
authtokenvarchar(250)NOTNULL
);
```

GITHUB LINK : [Click this link for complete source code](#)

5 Results

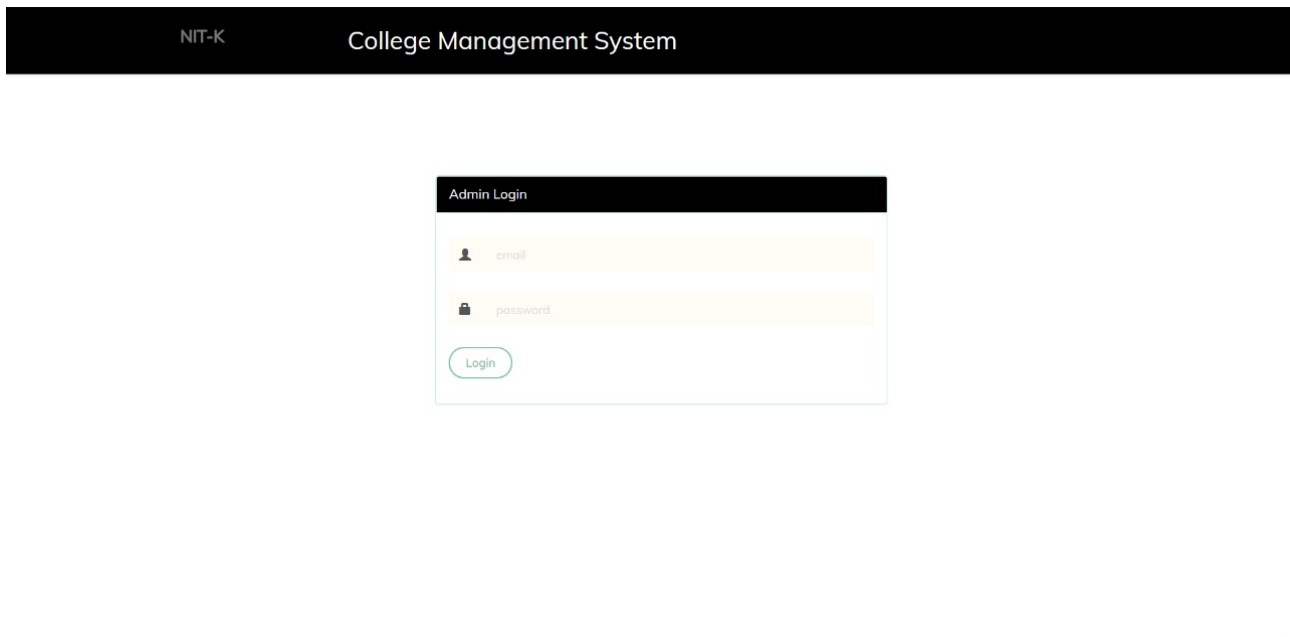


Figure 3: Admin login page

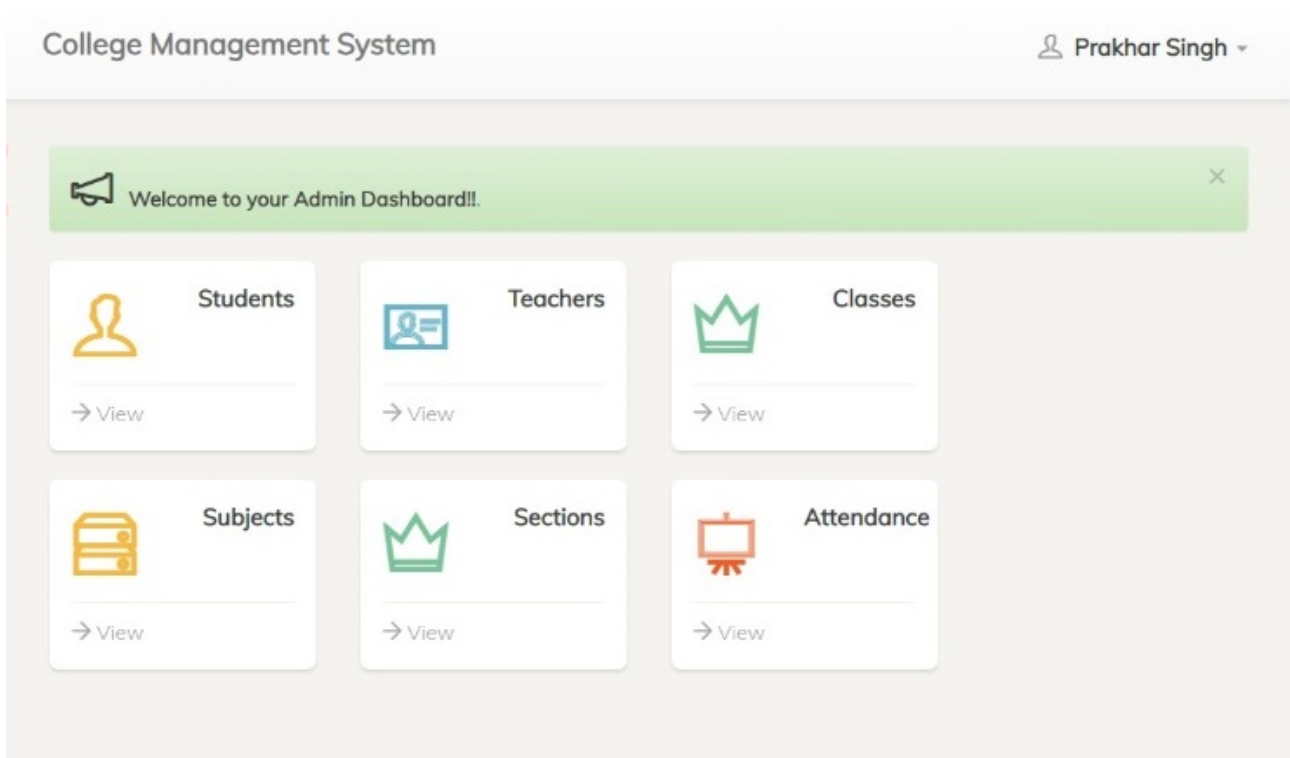


Figure 4: Dashboard

[Add New Class](#)

ID	Name ↕	Sections ↕	Class Teacher ↕		
4	class3	B	George	Update	Delete
3	class2	C	Mohan	Update	Delete
2	class1	A	Daniel	Update	Delete

Showing 1 to 3 of 3 entries

[Previous](#)
[1](#)
[Next](#)

Figure 5: Database record page

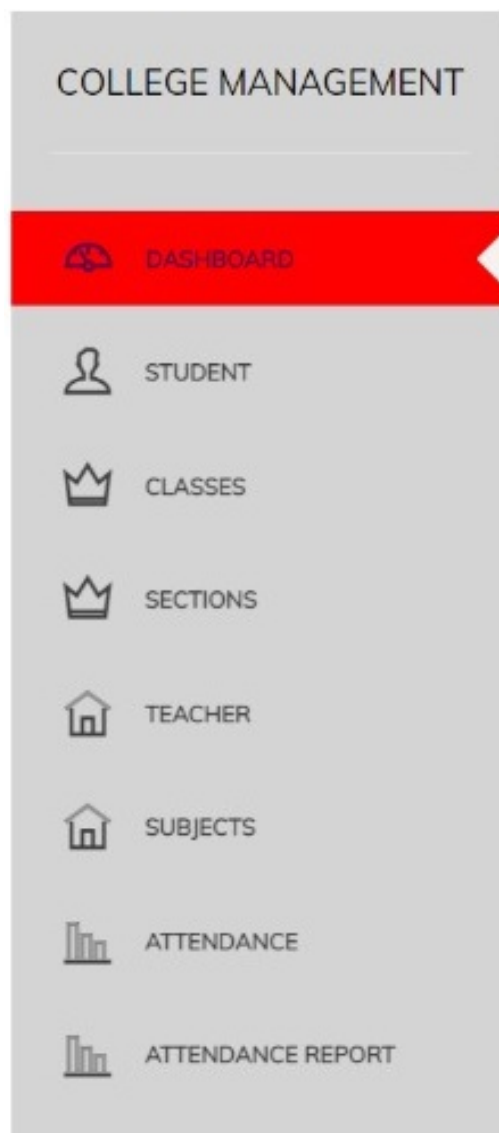


Figure 6: Navigation pane

6 References:

1. <https://stackoverflow.com/>
2. <https://dev.mysql.com/>
3. <https://athemes.com/collections/free-bootstrap-admin-templates/>
4. <https://www.flaticon.com/>

*** END ***