Database Management System

Mini Project Report

On

PROJECT HANDLER

Submitted by

DATTHA PRASAD
4SN17CS019

HARINANDAN
4SN17CS028

COMPUTER SCIENCE & ENGINEERING
(Visvesvaraya Technological University)

Under the Guidance of

Mr. Sudhakara B
Assistant Professor

Department of Computer Science & Engineering

SRINIVAS INSTITUTE OF TECHNOLOGY

(NAAC ACCREDITED)

MANGALURU-574143, KARNATAKA

2019 – 2020

SRINIVAS INSTITUTE OF TECHNOLOGY

(NAAC ACCREDITED)

MANGALURU -574143 – KARNATAKA

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

*This is to certify that the project entitled **"PROJECT HANDLER",** is an authentic record of the work carried out by*

DATTHA PRASAD	**HARINANDAN**
4SN17CS019	**4SN17CS028**

as prescribed by Visvesvaraya Technological University, Belagavi, for V Semester B.E. in Computer Science & Engineering during the year 2019-2020.

Mr.Sudhakara B	**Dr. Shivakumar G.S**	**Dr. Shrinivasa Mayya D**
Project Guide	**Head of the Department**	**Principal**

Name of the Examiners **Signature with Date**

1.

2.

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any work would be incomplete without thanking the persons who made it perfect with their constant guidance and encouragement.

We take this opportunity to express our sincere thanks and indebtedness to our Project guide and mentor, **Mr.Sudhakara B , Assistant Professor, Department of CS&E**, for his support and guidance. His vision and suggestions throughout the project period has been fundamental in the completion of the project.

We would like to extend our deep sense of acknowledgement to our **Project Co-ordinator Mrs.Sowmya.K, Assistant Professor, Department of CS&E**, for her encouragement and timely advice provided to us during the project work.

We extend our warm gratitude to **Dr. Shivakumar G. S, Head of the Department, Department of CS&E**, for his constant support and advice that helped us to complete this project successfully.

We are extremely grateful to our beloved **Principal, Dr. Shrinivasa Mayya D.** for his encouragement to come up with new ideas and advice to express them in a systematic manner.

We also like to thank all Teaching & Non-teaching staff of Srinivas Institute of Technology, Mangalore for their kind co-operation during the course of our work.

Dattha Prasad

Harinandan

ABSTRACT

Project handler is a software application to manage all the activities concern to the project of student. This project will help the faculty to store all project related information. This application is useful store all the information related to the report, team information, etc. This project also provided with insertion, update of project detail created by student.

The main objective of the application is to automate the existing method of managing data manually to systematic storage of data, which can be searched any time as and when required. The other main objective of this application is to store all the project and project related information except source code and document. The other main objective of this project is to store the student details who are creating the project.

TABLE OF CONTENTS

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

This system is designed in favor of the faculties, which helps them to save the records of the students about their project and other things.It helps them from the manual work from which it is very difficult to find the record of the students and the team details of the students .All the students details about project record at the present are managed manually by the coordinator .It is difficult to maintain hardcopy of each student . Thus there are a lot of repetitions which can be easily avoided .Hence there is a lot of strain on the person who are maintaining these records and software's are not usually used in this context.This particular project deals with the problems on managing a student project records and avoids the problems which occur when carried manually identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the exiting system with system which is more user friendly.We can improve the efficiency of handling the project especially for the Engineering college, through this they cannot require so efficient person to handle and calculate the things .

1.1 Problem Statement

The software product "Project Handler" will be an application that will be used for maintaining the records in an organized manner and to replace old paper work system .This project aims at automating the project collection of student for smooth working of the faculty by automating almost all the activities. Updations and modifications will be easily achievable and all the submission work would be accurate.

1.2 Scope of the Project

The scope of Project Handler includes:

- Authenticate end user at their login.
- Provide the details of student ,faculty, team, department.
- All student reports.
- Providing interface to update the student details.

CHAPTER 2

REQUIREMENT SPECIFICATION

2.1 Functional Requirements

A functional specification (functional specifications document (FSD), functional requirements specification) in systems engineering and software development is a document that specifies the functions that a system or component must perform. A functional specification is the more technical response to a matching requirements document.

A functional specification does not define the inner workings of the proposed system, it does not include the specification of how the system function will be implemented. Instead, it focuses on what various outside agents (for example, people using the program, computer peripherals, or other computers) might observe when interacting with the system.

Functional requirement for Project Handler includes:

- User : for update the student, create the team, registration, etc.

2.2 Non-Functional Requirements

In systems engineering and requirements engineering, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.

This software should produce the informative error messages, if any errors are found in the input program. The system must be capable of updating/displaying the record count within an acceptably short interval of the number of records changing.

2.3 Hardware Requirements

The minimum/recommended hardware configuration required for developing the proposed software is given below:

- 512 MB RAM
- 1.2GHz Processor

2.4 Software Requirements

- Front End
    - Net Beans IDE 8.0.1
    - MySQL 8.0 Command Line Client
    - MySQL connector for Net Beans
- Back End
    - MySQL Server 8.0 / MySQL Workbench 8.0
    - Windows 7/8/10

2.5 Software Tools Used

Project Handler is designed using Net Beans as front end user interface design tool and MySQL Server 8.0 at backend for creating tables and storing related information.

2.5.1 Front End Tool

NetBeans is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages. NetBeans is also referred to as a platform of modular components used for developing Java desktop applications. An IDE is much more than a text editor. The NetBeans Editor indents lines, matches words and brackets, and highlights source code syntactically and semantically. It lets you easily refactor code, with a range of handy and powerful tools, while it also provides code templates, coding tips and code generators. The editor supports many languages from Java, C/C++, XML and HTML, to PHP, Groovy, Javadoc, JavaScript and JSP. Because the editor is extensible, you can plug in support for many other languages.

Keeping a clear overview of large applications, with thousands of folders and files, and millions of lines of code, is a daunting task. NetBeans IDE provides different views of your data, from multiple project windows to helpful tools for setting up your applications and managing them efficiently, letting you drill down into your data quickly and easily, while giving you versioning tools via Subversion, Mercurial, and Git integration out of the box.

2.5.2 Back End Database Used

MySQL, pronounced either "My S-Q-L" or "My Sequel," is an open source relational database management system. It is based on the structure query language (SQL), which is used for adding, removing, and modifying information in the database. Standard SQL commands, such as ADD, DROP, INSERT, and UPDATE can be used with MySQL.

MySQL can be used for a variety of applications, but is most commonly found on Web servers. A website that uses MySQL may include Web pages that access information from a database. These pages are often referred to as "dynamic," meaning the content of each page is generated from a database as the page loads. Websites that use dynamic Web pages are often referred to as database-driven websites.

Many database-driven websites that use MySQL also use a Web scripting language like PHP to access information from the database. MySQL commands can be incorporated into the PHP code, allowing part or all of a Web page to be generated from database information. Because both MySQL and PHP are both open source (meaning they are free to download and use), the PHP/MySQL combination has become a popular choice for database-driven websites.

CHAPTER 3

SYSTEM DESIGN

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

3.1 ER Diagram

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases. ER diagrams are used to sketch out the design of a database. Figure 3.1 depicts the ER diagram of Project Handler. It shows various entities, their attributes, relationships with other entities, cardinality ratios between entities and participation constraints used in designing PH database.

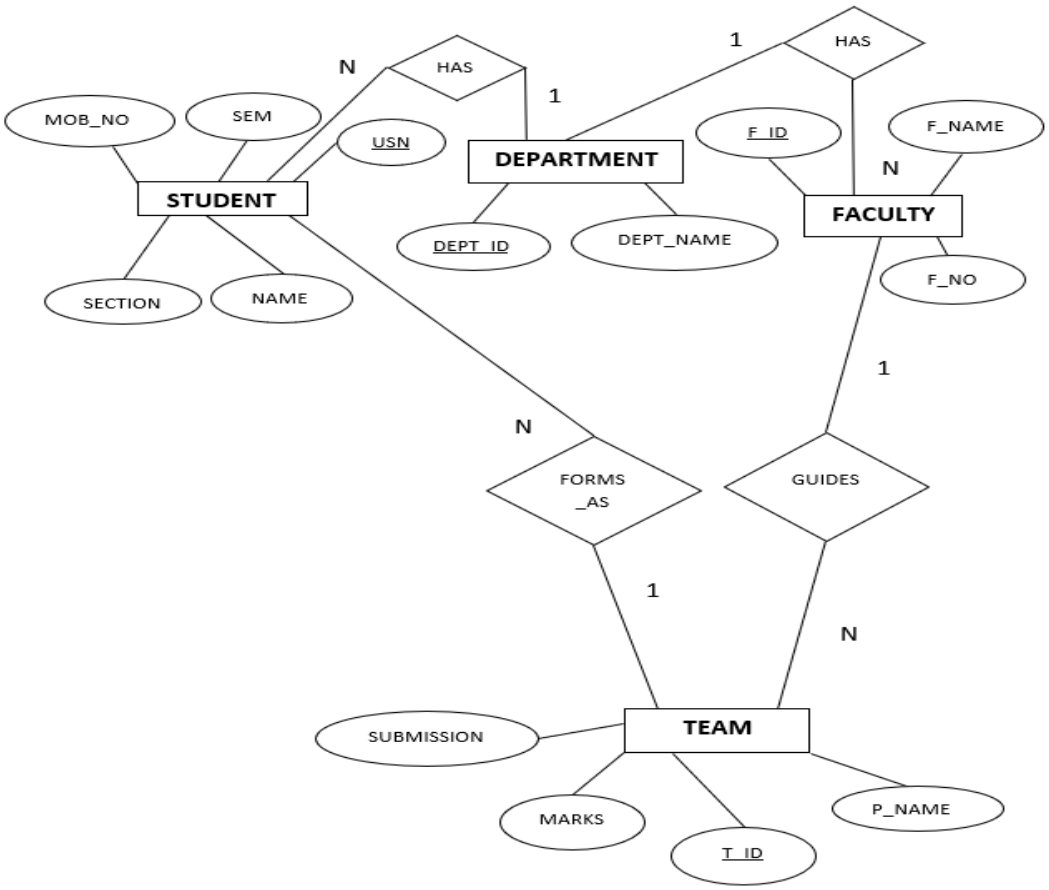


Figure 3.1: *ER Diagram*

3.2 Schema Diagram

A schema is the structure behind data organization. It is a visual representation of how different table relationships enable the schema's underlying mission business rules for which the database is created. In a schema diagram, all database tables are designated with unique columns and special features, e.g., primary/foreign keys or not null, etc. Formats and symbols for expression are universally understood, eliminating the possibility of confusion. The table relationships also are expressed via a parent table's primary key lines when joined with the child table's corresponding foreign keys.

Schema diagrams have an important function because they force database developers to transpose ideas to paper. This provides an overview of the entire database, while facilitating future database administrator work. Figure 3.2 shows the schema diagram of Project Handler.

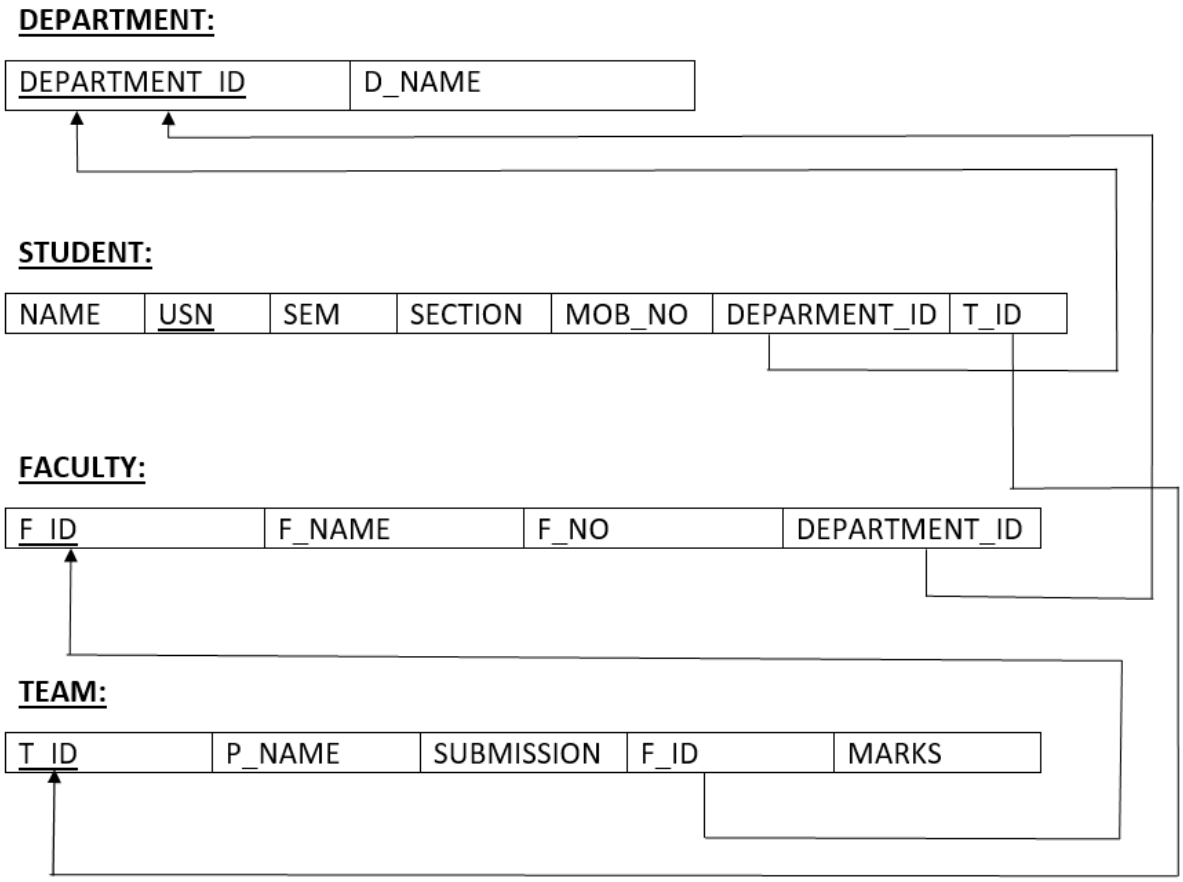


Figure 3.2: *Schema Diagram*

3.3 Database Table Design

The basic database unit is the table. A table is a unit consisting of rows of related information. Each row consists of fields of information where data is stored. Field attributes include information and rules that govern the data stored in the field. The field attributes and rules may limit the type of data stored in the field.

A field may be defined as a key or may be limited by rules requiring specific masks, such as a field may limited to dates, formatted numbers like telephone numbers, or be limited to a specific number of characters. The database schema contains these rules. Database tables used in PH are shown below.

Table 3.3.1:User table

Name	Data Type	Constraints
U_name	Varchar(40)	
U_Pass	Varchar(40)	

The table 3.3.1 is used to store user information like User name, Password when new User is added to database by the end user for security purpose .

Table 3.3.2:Department table

Name	Data Type	Constraint
Id	int(20)	Primary Key
Department	varchar(60)	

The table 3.3.2 is used to store the information related to the department.

Table 3.3.3:Student table

Name	Data Type	Constraint
Name	varchar(40)	
Usn	varchar(40)	Primary Key
Sem	int(20)	
Section	varchar(20)	
Student_mobile	varchar(40)	
Id	int(20)	Foreign Key
Team_Id	varchar(25)	Foreign Key

The table 3.3.3 is used to store the information about the student in each sem or branch in each department.

Table 3.3.4:Faculty table

Name	Data Type	Data Type
Faculty_Id	varchar(25)	Primary Key
Faculty	varchar(40)	
Mobile	varchar(40)	
Id	int(20)	Foreign Key

The table 3.3.4 is used to store the information about the faculty in each department.

Table 3.3.5:Team table

Name	Data Type	Constraint
Team_Id	varchar(25)	Primary Key
Project	varchar(40)	
Faculty_Id	varchar(25)	Foreign Key
Marks	int(20)	
Submission	bit	

The table 3.3.5 is used to store the information about the team and respective projects in each department.

3.4 Use-case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. Users interacting with application are shown outside with stickman symbol.

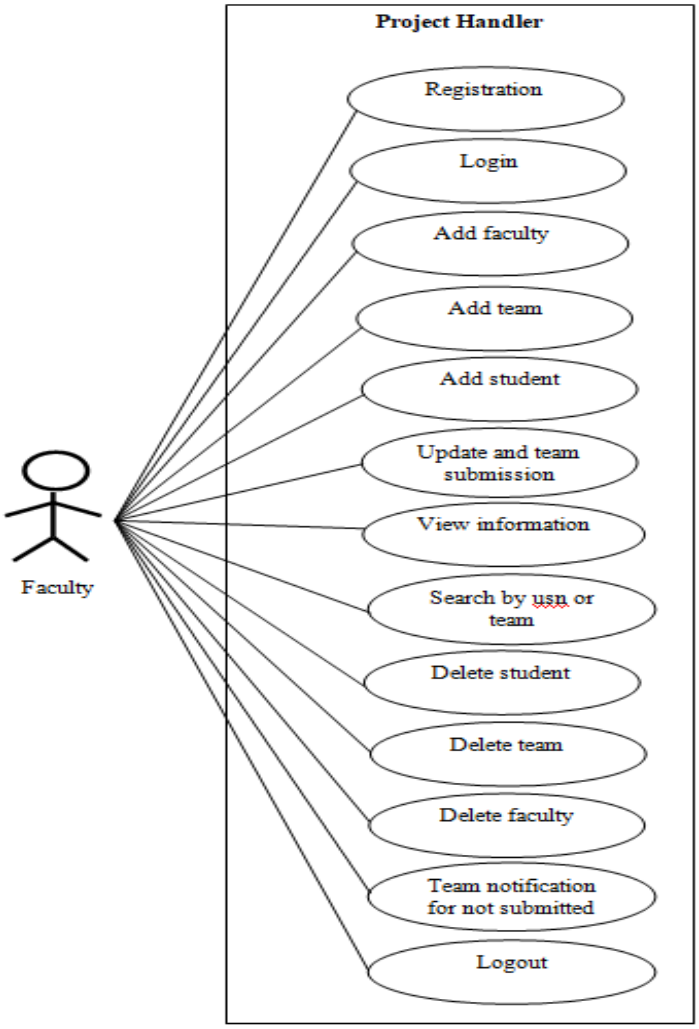


Figure 3.4:*Use-case Diagram*

CHAPTER 4

IMPLEMENTATION

System implementation is the important stage of project when the theoretical design is tuned into practical system.

4.1 Creating Database Using MySQL

Queries used for creating database and different tables used in Project Handler are given below.

To create new database

mysql> create database Project;

To use newly created database

mysql> use Project;

Database changed

To create table USER

mysql> create table User(U_name varchar(11), U_pass varchar(11));

To create table DEPARTMENT

mysql>create table department(ID int(20),DEPARTMENT varchar(40),constraint pk_did primary key(ID));

To create table FACULTY

mysql>create table faculty(FACULTY_ID varchar(25),FACULTY varchar(40),MOBILE int(20),ID int(20),constraint pk_facutly_id primary key(FACULTY_ID),constraint fk_did foreign key(ID) references department(ID) on update cascade on delete cascade);

To create table TEAM

mysql>create table team(TEAM_ID varchar(25),PROJECT varchar(40),FACULTY_ID varchar(25),MARKS int(20),SUBMISSION bit,constraint pk_tid primary key(TEAM_ID),constraint fk_team_fid foreign key(FACULTY_ID) references faculty(FACULTY_ID) on update cascade on delete cascade);

To create table STUDENT

mysql>create table student(NAME varchar(40),USN varchar(30),SEM int(20),SECTION varchar(20),STUDENT_MOBILE int(20),ID int(20),TEAM_ID varchar(25),constraint pk_usn primary key(USN),constraint fk_did_student foreign key(ID) references department(ID) on update cascade on delete cascade,constraint fk_tid_student foreign key(TEAM_ID) references team(TEAM_ID) on update cascade on delete cascade);

To insert values to LOGIN table

mysql>insert into User values('ADMIN','123');

To view values in table USER

```
mysql> SELECT * FROM USER;
+----------+----------+
| U_name   | U_pass   |
+----------+----------+
| ADMIN    | 123      |
| USER     | USER123  |
+----------+----------+
```

Figure 4.1.1: *User table*

To insert values to DEPARTMENT table

mysql> insert into department values(1,'COMPUTER SCIENCE AND ENGINEERING');

To view values in table DEPARTMENT

```
mysql> SELECT * FROM DEPARTMENT;
+----+-------------------------------------+
| ID | DEPARTMENT                          |
+----+-------------------------------------+
|  1 | COMPUTER SCIENCE AND ENGINEERING    |
|  2 | MECHANICAL ENGINEERING              |
|  3 | ELECTRONICS ENGINEERING             |
|  4 | AERONAUTICAL ENGINEERING            |
|  5 | CIVIL ENGINEERING                   |
+----+-------------------------------------+
```

Figure 4.1.2: *Department table*

To insert values to FACULTY table

mysql> insert into faculty values('F1','SUDHAKARA B',944787897,1);

To view values in table FACULTY

```
mysql> SELECT * FROM FACULTY;
+------------+-----------------+-----------+------+
| FACULTY_ID | FACULTY         | MOBILE    | ID   |
+------------+-----------------+-----------+------+
| F1         | SUDHAKARA B     | 944787897 |    1 |
| F2         | MOHAN K         | 944787898 |    1 |
| F3         | NAGARAJ HEBBER  | 944787798 |    1 |
| F4         | SOWMYA K        | 844997798 |    1 |
| F5         | GANGOTHRI       | 844997790 |    1 |
+------------+-----------------+-----------+------+
```

Figure 4.1.3: *Faculty table*

To insert values to STUDENT table

mysql> insert into student values('DATTHA PRASAD',4SN17CS019,5,'A',978767896,1,T1);

To view values in table STUDENT

```
mysql> SELECT * FROM STUDENT;
+-----------------+------------+------+---------+----------------+------+---------+
| NAME            | USN        | SEM  | SECTION | STUDENT_MOBILE | ID   | TEAM_ID |
+-----------------+------------+------+---------+----------------+------+---------+
| ANUDEEP         | 4SN17CS005 |    5 | A       |      978009090 |    1 | T2      |
| PRJAWAL         | 4SN17CS011 |    5 | A       |      978769990 |    1 | T4      |
| DHANUSH         | 4SN17CS017 |    5 | A       |      908769990 |    1 | T4      |
| DATTHA PRASAD   | 4SN17CS019 |    5 | A       |      978767896 |    1 | T1      |
| DHEERAJ J RAO   | 4SN17CS021 |    5 | A       |      932788890 |    1 | T5      |
| HARINANDAN      | 4SN17CS028 |    5 | A       |      978769096 |    1 | T1      |
| HARIPRASAD      | 4SN17CS029 |    5 | A       |      908788890 |    1 | T5      |
| HRITHIK         | 4SN17CS030 |    5 | A       |      978769090 |    1 | T2      |
| KOUSHIK BANGERA | 4SN17CS041 |    5 | A       |      978005650 |    1 | T3      |
| NIKIL FLOYD     | 4SN17CS048 |    5 | A       |      978767650 |    1 | T3      |
+-----------------+------------+------+---------+----------------+------+---------+
```

Figure 4.1.4: *Student table*

To insert values to TEAM table

Mysql> insert into team values('T1','PROJECT HANDLER','F1',40,1);

To view values in table TEAM

```
mysql> SELECT * FROM TEAM;
+---------+---------------------------+------------+-------+------------+
| TEAM_ID | PROJECT                   | FACULTY_ID | MARKS | SUBMISSION |
+---------+---------------------------+------------+-------+------------+
| T1      | PROJECT HANDLER           | F1         |    40 | ▯          |
| T2      | HOSPITAL MANAGEMENT SYSTEM | F2         |    35 |            |
| T3      | ONLINE APPTITUDE          | F3         |    40 | ▯          |
| T4      | TAXI MANAGEMENT SYSTEM    | F4         |    40 | ▯          |
| T5      | CRICKET MANAGEMENT SYSTEM | F5         |    40 | ▯          |
+---------+---------------------------+------------+-------+------------+
```

Figure 4.1.5: *Team table*

4.2 Stored Procedure

A stored procedure is a prepared SQL code that can be reused over and over again. So if an SQL query needs to be written over and over again, save it as a stored procedure, and then just call it to execute it. It is also possible to pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.

Stored procedure used in Project Handler is as follows

```
DELIMITER //
CREATE PROCEDURE get_department()
 BEGIN
SELECT * FROM department;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE ‘dps1’(IN fid1 varchar(25))
BEGIN
UPDATE TEAM SET FACULTY_ID=NULL WHERE FACULTY_ID=fid1;
END
DELIMITER ;

DELIMITER //
CREATE PROCEDURE ‘dps2’(IN TID1 varchar(25))
BEGIN
UPDATE STUDENT SET TEAM_ID=NULL WHERE TEAM_ID=TID1;
END
DELIMITER ;
```

4.3 Triggers

A trigger is a special type of stored procedure that automatically executes when an event occurs in the database server. DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view. These triggers fire when any valid event is fired, regardless of whether or not any table rows are affected.

Trigger used in Project Handler is as follows

Here Trigger is used whenever the faculty is deleted from the faculty table it should be appeared as null in the team.

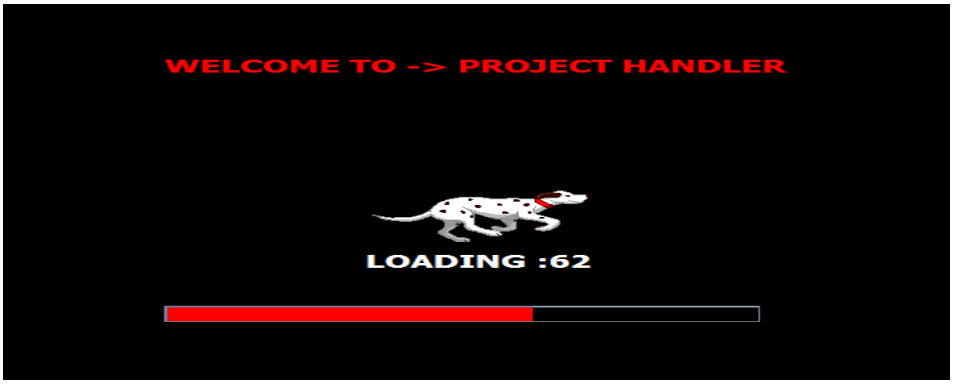
```
DELIMITER $$
CREATE TRIGGER DPS22 BEFORE DELETE ON FACULTY
FOR EACH ROW
BEGIN
        If(OLD.FACULTY_ID IS NOT NULL) THEN
        Call ‘dps1’(OLD.FACULTY_ID);
        END If
END $$
DELIMITER ;
```

Here Trigger is used whenever the team is deleted from the team table it should be appeared as null in the student.

```
DELIMITER $$
CREATE TRIGGER DPS11 BEFORE DELETE ON TEAM
FOR EACH ROW
BEGIN
        If(OLD.TEAM_ID IS NOT NULL) THEN
        Call ‘dps2’(OLD.TEAM_ID);
        END If
END $$
DELIMITER ;
```

CHAPTER 5

SCREEN SHOTS



Welcome page

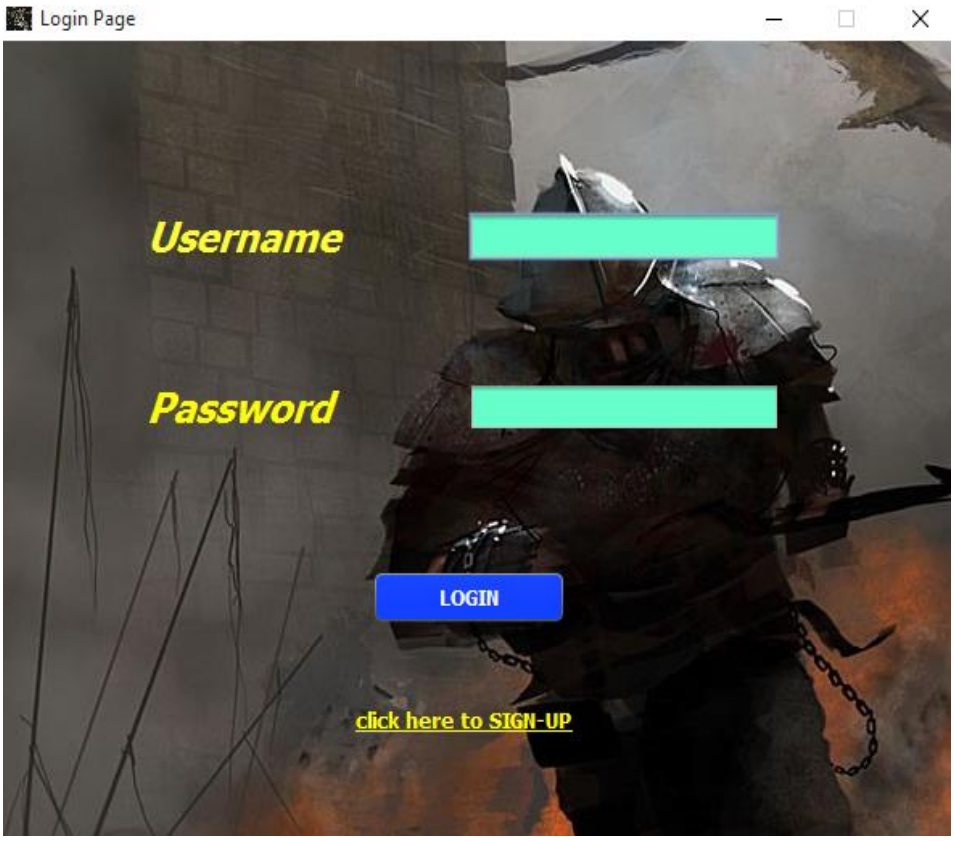


Figure 5.1: *Login page*

The figure 5.1 shows the design of Login page. The Login page contains the username, password and link for sign-up.

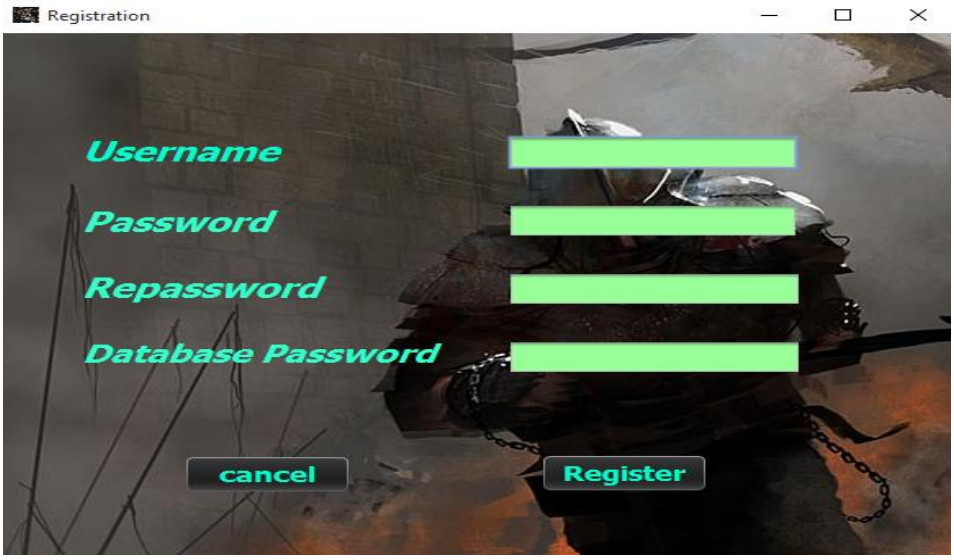


Figure 5.2: *Registration page*

The figure 5.2 shows the design of Registration page. This page contains the username, password, confirm password and database password for creating user account.

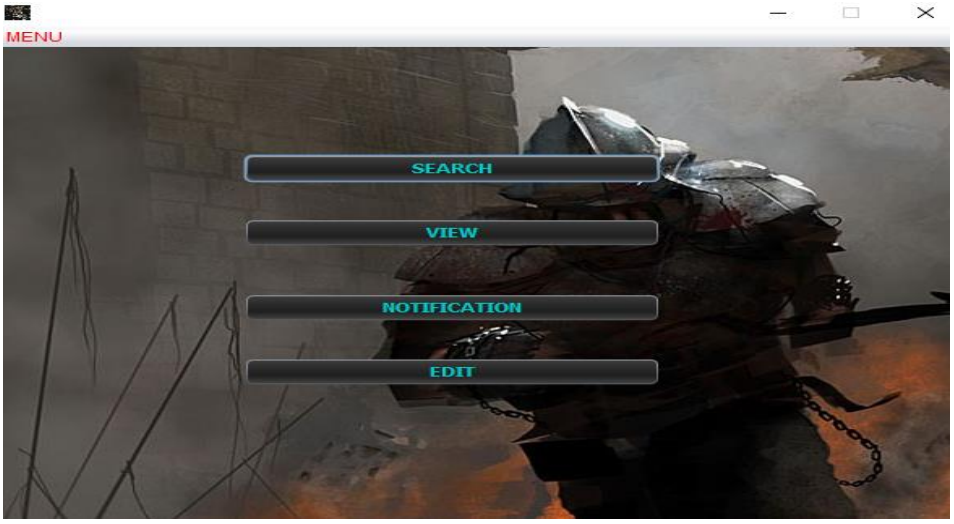


Figure 5.3: *Home page*

The figure 5.3 shows the design of Home page. This page contains the search, view, notification, edit option for user, Also a menu option which contains logout and information of software .

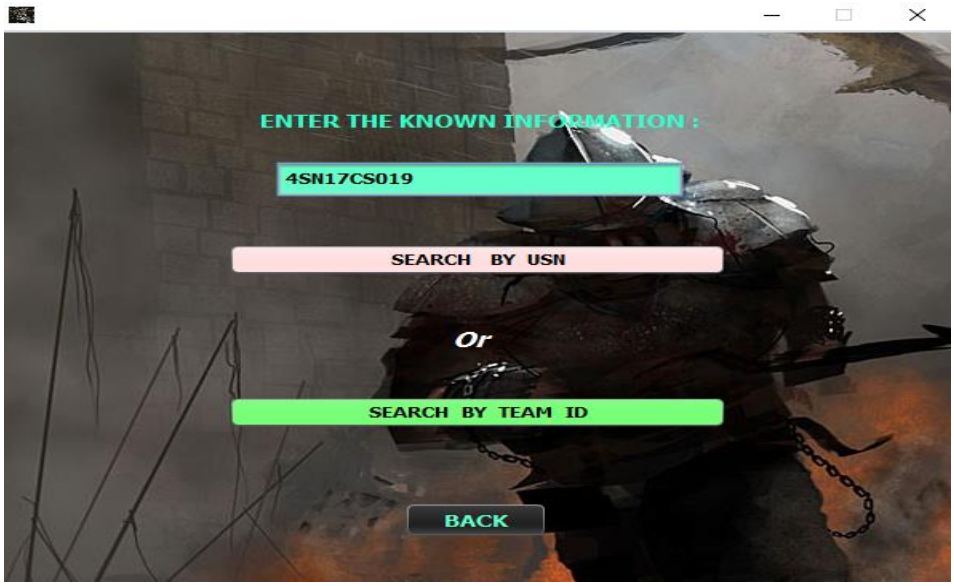


Figure 5.4: *Search page*

The figure 5.4 shows the design of Search page. This page contains the search option for student (usn) and team (team_id).

Figure 5.5: *Student detail*

The figure 5.5 shows the design of Student detail. This page contains the details of the student whose usn is given in the search page.

Figure 5.6 *Team detail*

The figure 5.6 shows the design of team detail. This page contains the details of the team whose team_id is given in the search page

DEPARTMENT (DEFAULT)

ID	DEPARTMENT
1	COMPUTER SCIENCE AND ENGINEERING
2	INFORMATION SCIENCE AND ENGINEERING
3	MECHANICAL ENGINEERING
4	ELECTRICAL ENGINEERING
5	MARINE ENGINEERING
6	ELECTRONICS AND COMMUNICATION ENGINEERING
7	ELECTRICAL AND ELECTRONICS ENGINEERING
8	CIVI ENGINEERING
9	NANO TECHNOLOGY AND ENGINEERING
10	AERONAUTICAL ENGINEERING
11	OTHERS

Figure 5.7:*Department information*

NOT-SUBMITED-LIST

TEAM_ID	STUDENT_MOBILE	USN
T1	7349585302	4SN17CS019
t2	7349585302	4SN17CS021
T1	7349887565	4SN17CS028
t2	734988757777	4SN17CS029

Figure 5.8:*Notification*

The figure 5.8 shows the design of notification table. This page contains information about the student who are not submitted.

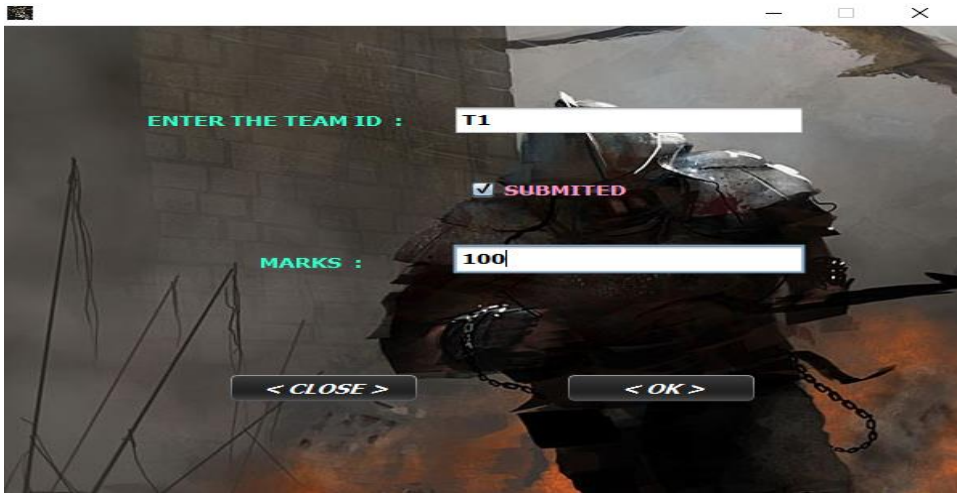


Figure 5.9:*Team update form*

The figure 5.9 shows the design of team update form. This page contains the team_id and update field for marks and submission.

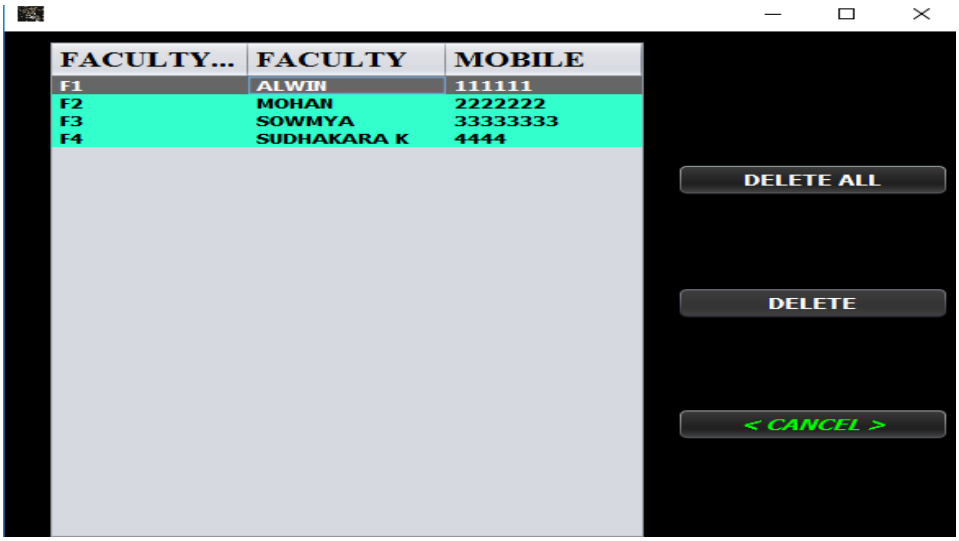


Figure 6.0:*Delete faculty form*

CONCLUSION AND SCOPE FOR FUTURE WORK

Project Handler provides a computerized version of project handling system which will benefit the staff. It makes entire process easy where staff can generate record and manage the records. It also has a facility for user login where faculty can login and can see the team information, status of students project as well as notification about the submission.

There is a future scope of this facility includes many features such as online submission of report, synopsis and other things also it can store the source code, project folder ,databases thus making it more interactive more user friendly and project which fulfills each users need in the best way possible.

BIBLIOGRAPHY

- Database Systems Models, Languages, Design and Applications Programming, RamezElmasri and ShamKantB.Navathe, 6th Edition, Pearson
- Netbeans: The Definitive Guide, 1st Edition, Tim Boudreau, Jesse Glick, Simeon Lewis.
- Abraham Silberschatz, Henry F. Korth and S. Sudarshan, Database System Concepts, McGraw-Hill Education (Asia), Fifth Edition, 2006.
- Shio Kumar Singh, Database Systems Concepts, Designs and Application, Pearson Education, Second Edition, 2011.
- www.wikipedia.org
- https://github.com/datthaprasad
- https://Stackoverflow.com