

Registration Management System

Registration and Student Card
Generation Management System



BSE-3B

SEMESTER

PROJECT

....

TEAM MEMBERS



Hamza
Altaf

SP20-BSE-047






Ossama
Mehmood

SP20-BSE-069



Sara
Sultan

SP20-BSE-063

 Introduction Documentation Requirement System Features Modeling Phase User Interference Database Connectivity Implementation Maintenance

Registration and Student Card Generation Management System

PREPARED FOR










Ma'am,
Ayesha Hussain
&
Walia Fatima

PREPARED BY

Hamza Altaf – Sara Sultan – Ossama Mehmood

June 11, 2021

TABLE OF

-  Introduction
-  Documentation
-  Requirement
-  System Features
-  Modeling Phase
-  User Interference
-  Database Connectivity
-  Implementation
-  Maintenance

Introduction

Documentation

Requirement Engineering

--- Functional Requirements

--- Non-Functional Requirements

System Features

Modeling & Diagram Phase

User Interference Design

Database Connectivity


Implementation


Maintenance & Performance


CONTEXT

Registration

 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation


 Maintenance


Introduction


- This **Project reviews** the relevant research conducted on online student enrollment used by various universities. Includes essentials for online registration such as online registration program, online registration course, and others.
- To **achieve this goal** it is **difficult** to use the system as information is scattered, it can be cluttered, and collecting relevant information can be time-consuming. It can be used by **educational institutions** or colleges to maintain student records easily. It also provides a time-consuming process to view, add, edit and delete student marks.
- The **online registration system** will allow online submission of student applications, Student Registrations, Online quiz or test. Saved test results, change student profile.

Registration

 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation

 Maintenance

Documentation

➤ In Documentation phase we use to build a strong analysis and requirement gathering of our project.

➤ We use to made several document included

❑ Vision Document

❑ Software Requirements Specification (SRS)


Developed Use Cases


Full Dress Use Cases included in


Documentation with there brief descriptions.

Registration

 Introduction


 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation

 Maintenance

Requirement

➤ Requirement are divided into : -

➤ **Functional** requirements.


➤ **Non-functional** requirements.


Functional Requirements : -


- Storage of student information, including marks and visits
- Technical management
- Report production

Registration


 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity


 Implementation


 Maintenance


- Tool-assisted editing
- Student and Admin access
- Registration or subscription management
 - Add/Update/Delete Student Information
 - Add/Update/Delete Subject Information & field choice Information.
 - Create/Delete Student and Admin Accounts

Registration

 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation


 Maintenance


Non-Functional requirements.


- In this system, the authentication of the user is a crucial issue.
- During this system, user authentication are going to be done by login by user name and Arcanum and classified by user sort.
- Users can get access to the system as permissions square measure classified for that style of user.

Registration

 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation


 Maintenance


System Features (Full Dressed Use Cases) : -


- In this system, the authentication of the user is a crucial issue.
- Title: Enter the goal of the utilization case – ideally as a brief, active phrase.
- Description: Describe the goal and context of this use case. This is often sometimes a distended version of what you entered within the “Title” field.
- Primary Actor: A person or a software/hardware system that interacts along with your system to attain the goal of this use case.
- Pre-conditions: Describe the state the system is in before the primary event during this use case.

Registration

 Introduction


 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation

 Maintenance

System Features (Full Dressed Use Cases) : -

- Post-condition: Describe the state the system is in in any case the events during this use case have taken place.
- Main Success Scenario: As you'll be able to see, this field contains the instance from our previous post – i.e. the flow of events from pre-conditions to post-conditions, once nothing goes wrong.

Registration

- Introduction
- Documentation
- Requirement
- System Features
- Modeling Phase
- User Interference
- Database Connectivity
- Implementation
- Maintenance

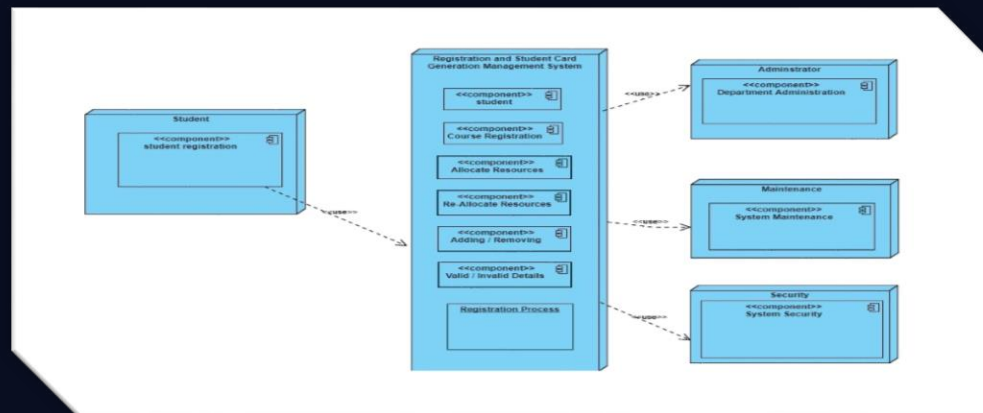
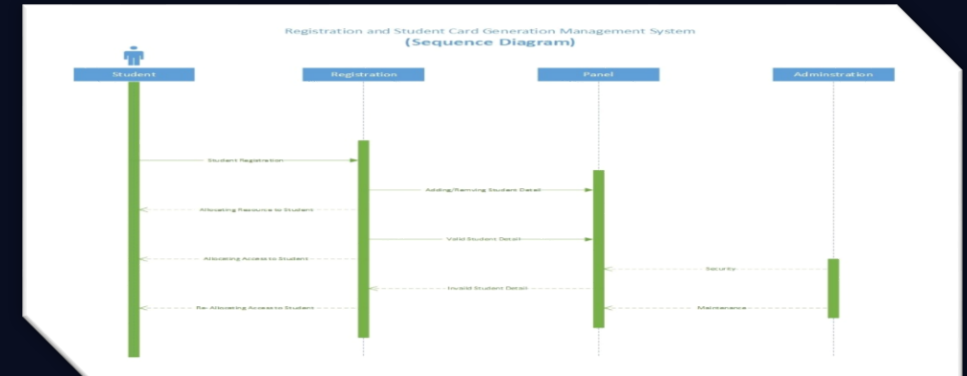
Modeling Phase

Sequence & Deployment Diagram :-

In the **sequence diagram** we build a strong connection between student, registration, panel and administrators.

There's a step **followed** in a sequence that can bring it back as well as move forward depending on the condition satisfied or not.

It's to **build a sequence** or a fellow moving backward or forward using nodes.





Deployment Diagram is more towards implementation the **component** are used to build the relation.


The **Student Register** and moving forward too whole **registration** propose with the help of administrator and factors based on **maintainability** or **security**.

Registration

 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation

 Maintenance










User Interference

- In User Interface Design, We'll design a login page to access the home interface to perform different operation on the Interface.
- Login Page will use to access Home Page.

Java Frame (Jframe)

- We're using Jframe to design user interface.
- Implemented as an instance of the JFrame class, is a window that has decorations such as a border, a title, and supports button components that close or iconify the window.
- Applications with a GUI usually include at least one frame

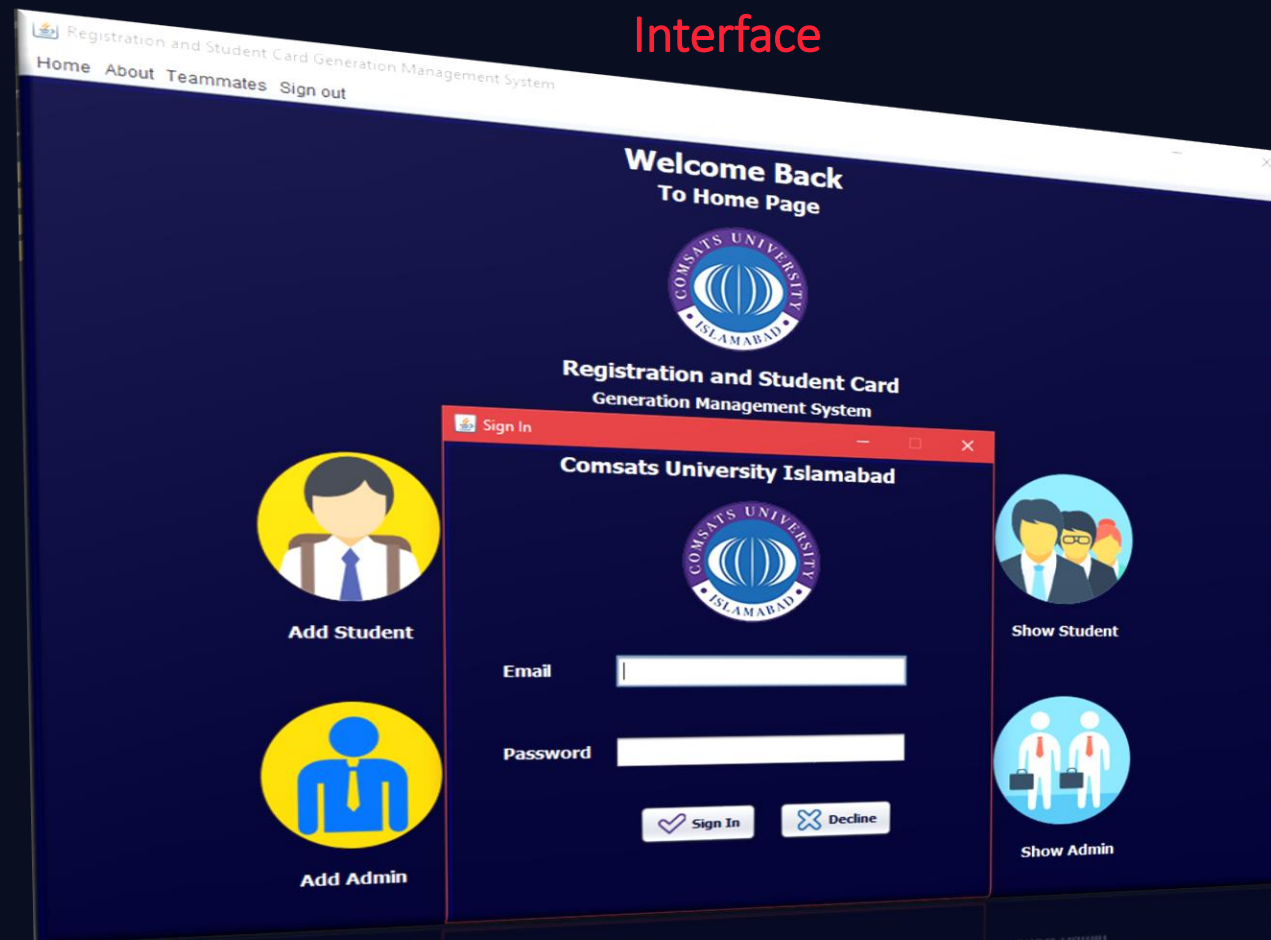
Registration

-  Introduction
-  Documentation
-  Requirement
-  System Features
-  Modeling Phase
-  User Interference
-  Database Connectivity
-  Implementation
-  Maintenance










Semester Project

User Interference

Project Interface



Registration

-  Introduction
-  Documentation
-  Requirement
-  System Features
-  Modeling Phase
-  User Interference
-  Database Connectivity
-  Implementation
-  Maintenance

User Interference

Login
Page

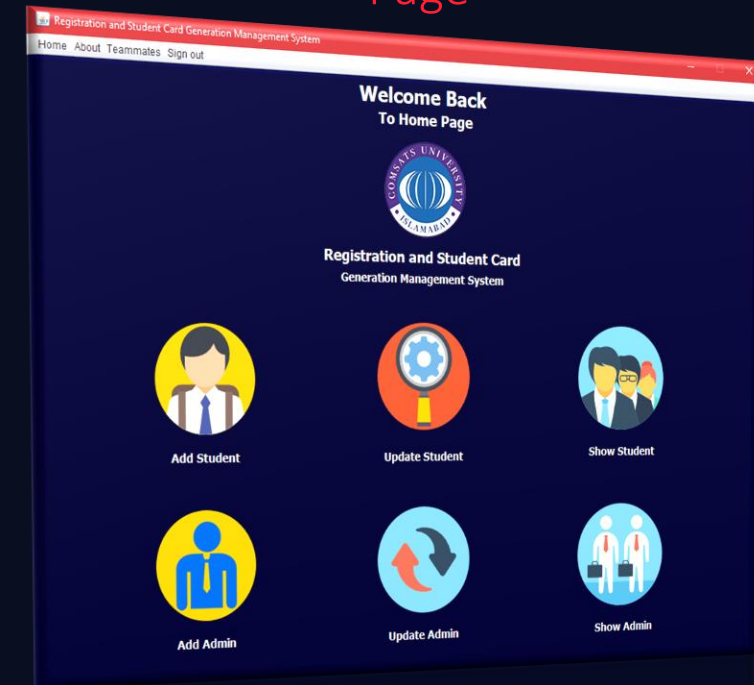


The login page is titled "Sign In" and "Comsats University Islamabad". It features the university's logo at the top center. Below the logo, there are two input fields: "Email" and "Password". At the bottom, there are two buttons: "Sign In" (with a heart icon) and "Decline" (with an X icon).

Login Page Interface

Login Page will use to access Home Page.










Home
Page



Home Page Interface

Home Page will use to access all functionalities.

Registration

-  Introduction
-  Documentation
-  Requirement
-  System Features
-  Modeling Phase
-  User Interference
-  Database Connectivity
-  Implementation
-  Maintenance

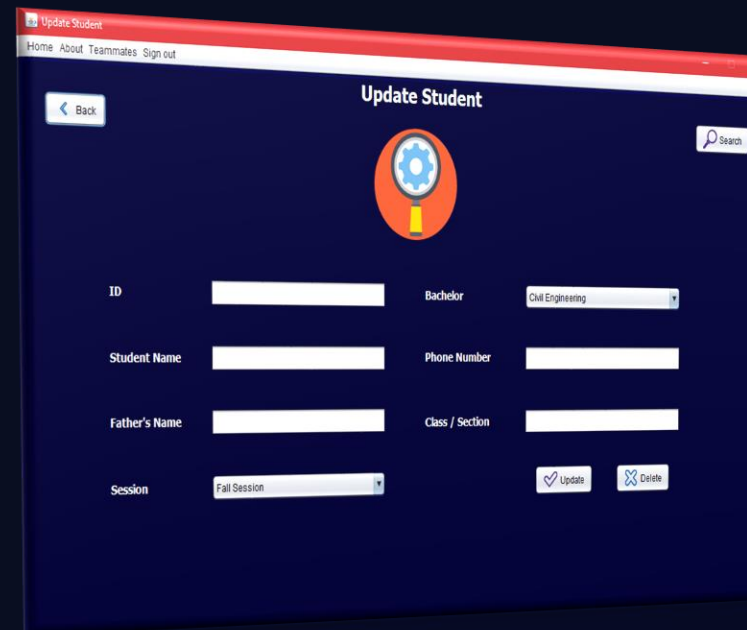
User Interference

Add Student



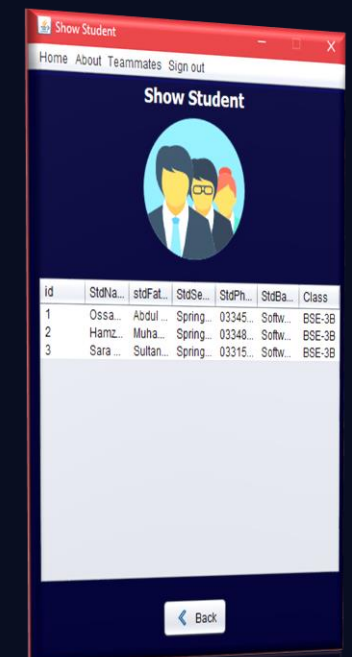
The 'Add Student' form is a web application window with a dark blue header and a light blue sidebar. The main content area is white. It features a large yellow circular icon with a person silhouette at the top. Below it, there are input fields for 'Student Name', 'Father's Name', 'Session' (a dropdown menu), 'Bachelor' (a dropdown menu), 'Phone Number', and 'Class/Section'. At the bottom, there are 'Submit' and 'Back' buttons.

Update Student



The 'Update Student' form is a web application window with a dark blue header and a light blue sidebar. The main content area is white. It features a large orange circular icon with a gear and a person silhouette at the top. Below it, there are input fields for 'ID', 'Bachelor' (a dropdown menu), 'Student Name', 'Phone Number', 'Father's Name', 'Class / Section', and 'Session' (a dropdown menu). At the bottom, there are 'Update' and 'Delete' buttons.

Show Student



The 'Show Student' form is a web application window with a dark blue header and a light blue sidebar. The main content area is white. It features a large blue circular icon with two people silhouettes at the top. Below it, there is a table with student data. At the bottom, there is a 'Back' button.

Id	StdNa...	stdFat...	StdSe...	StdPh...	StdBa...	Class
1	Ossa...	Abdul...	Spring...	03345...	Softw...	BSE-3B
2	Hamz...	Muha...	Spring...	03348...	Softw...	BSE-3B
3	Sara ...	Sultan...	Spring...	03315...	Softw...	BSE-3B

Student Interface

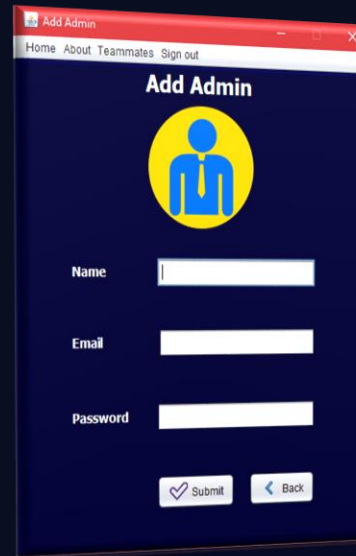
Add, Update/Delete/Search and Show Student

Registration

- Introduction
- Documentation
- Requirement
- System Features
- Modeling Phase
- User Interference
- Database Connectivity
- Implementation
- Maintenance

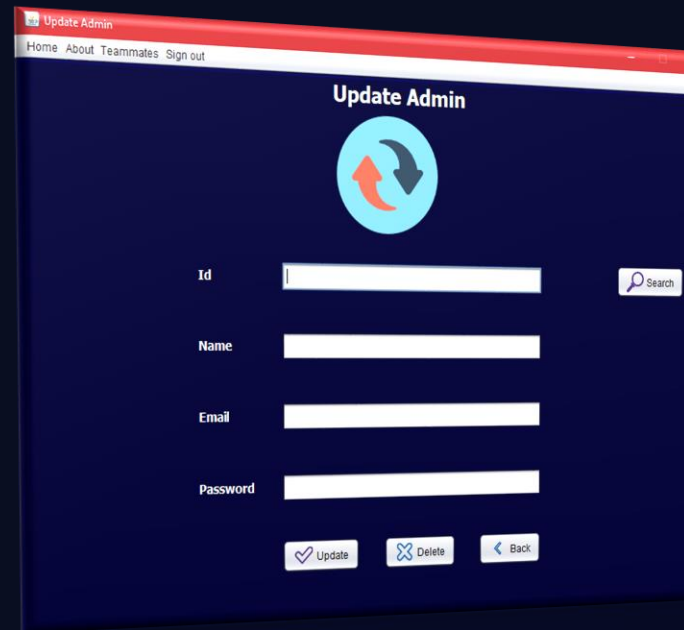
User Interference

Add
Admin



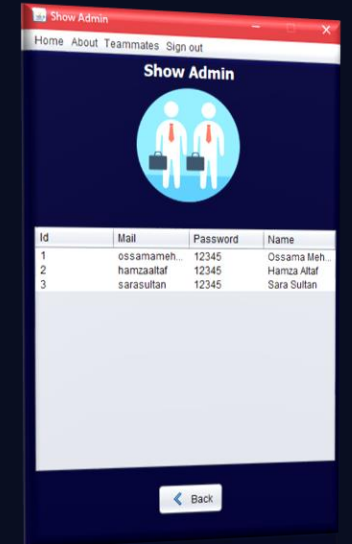
A screenshot of the 'Add Admin' web form. It features a yellow circular icon with a blue person silhouette. Below the icon are three input fields labeled 'Name', 'Email', and 'Password'. At the bottom, there are two buttons: 'Submit' with a checkmark icon and 'Back' with a left arrow icon.

Update
Admin



A screenshot of the 'Update Admin' web form. It features a blue circular icon with two curved arrows. Below the icon are four input fields labeled 'Id', 'Name', 'Email', and 'Password'. To the right of the 'Id' field is a search icon. At the bottom, there are three buttons: 'Update' with a checkmark icon, 'Delete' with an 'X' icon, and 'Back' with a left arrow icon.

Show
Admin












A screenshot of the 'Show Admin' web page. It features a blue circular icon with two people silhouettes. Below the icon is a table with four columns: 'Id', 'Mail', 'Password', and 'Name'. The table contains three rows of data. At the bottom, there is a 'Back' button with a left arrow icon.

Id	Mail	Password	Name
1	ossamameh...	12345	Ossama Meh...
2	hamzaaltaf	12345	Hamza Altaf
3	sarasultan	12345	Sara Sultan

Admin Interface

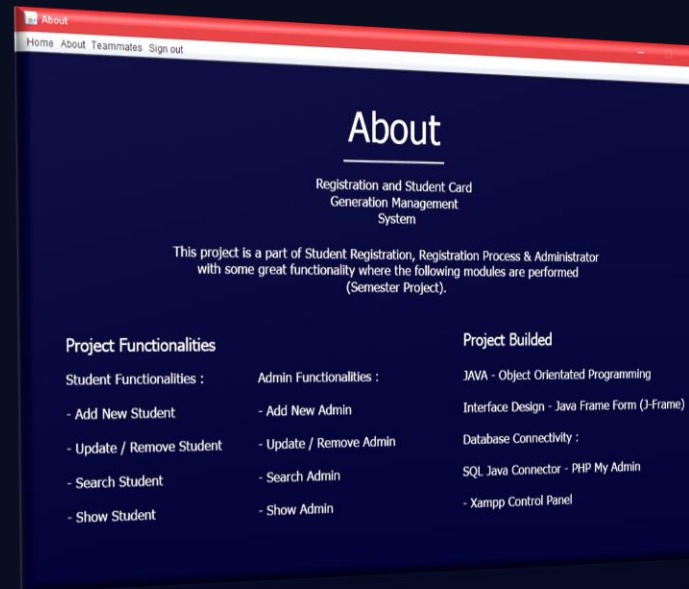
Add, Update/Delete/Search and Show Admin

Registration

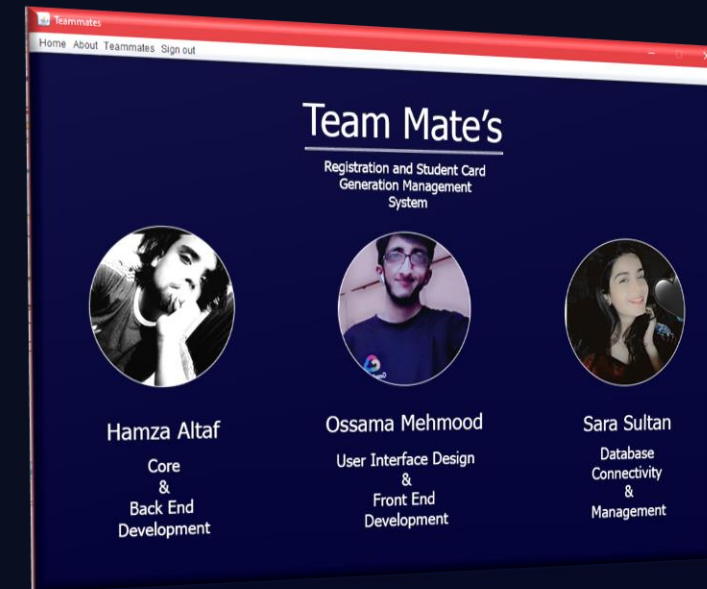
-  Introduction
-  Documentation
-  Requirement
-  System Features
-  Modeling Phase
-  User Interference
-  Database Connectivity
-  Implementation
-  Maintenance

User Interference

About Page



Teammates Page





About and Teammates


About and Teammates Page Interface

Registration

 Introduction

 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation

 Maintenance

Database Connectivity

PHP My Admin :

We're using **Localhost – Php My Admin** – Online Database to store details.

We used **SQL Connector** to build connection between database and NetBeans.

We use **XAMPP** Control Panel to access Database **phpmyadmin**.

We're storing **Student, Admin** and **there registration** details there.

We're able to **Add/Update/Delete - Student/Admin** details directly from Database as well as from our Registration Management System.

Registration

Introduction

Documentation

Requirement

System Features

Modeling Phase

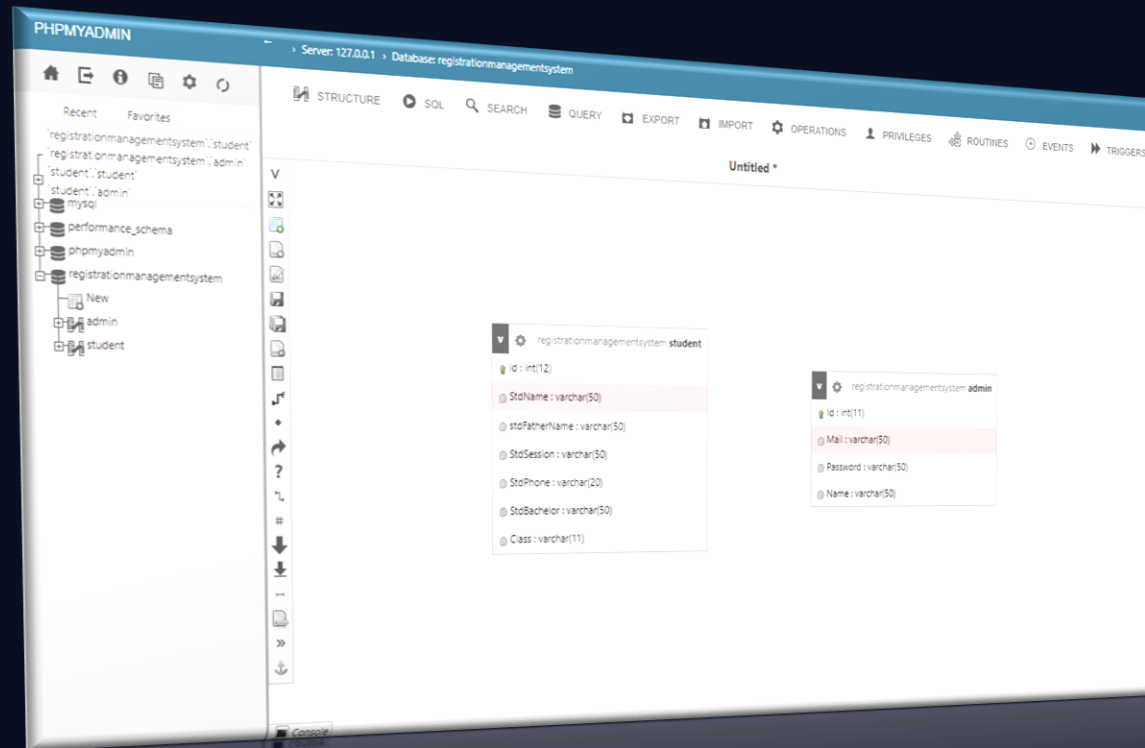
User Interference

Database Connectivity

Implementation










Maintenance

Database Connectivity



Access Our Database Here
<http://localhost/phpmyadmin>

Registration

-  Introduction
-  Documentation
-  Requirement
-  System Features
-  Modeling Phase
-  User Interference
-  Database Connectivity
-  Implementation
-  Maintenance


Implementation


In **Implementation** phase we use to **code** or **developed** the **Registration Management System**


- Implementation will be performing using **NetBeans**
- We're using **Jframe** with implementation to design a user friendly interface as well with implementation.
- We're using **Phpmyadmin** (Localhost) for storing Student, Admin and Registration details.

Registration

 Introduction


 Documentation

 Requirement


 System Features

 Modeling Phase

 User Interference

 Database Connectivity

 Implementation

 Maintenance

Maintenance

In **Maintenance**, We use to deal with **different factors** that help use to toward **improvement in performance**.

1. Delete **irrelevant content** and **store old content** elsewhere
2. Purge **inactive** users
3. **Schedule version** or **system updates** and down time
4. Reorganize **modules** in a **logical order**
5. Cut out the **“fluff”**



THANK YOU SO MUCH

Ma'am Ayesha & Walia

From: (SP20) - BSE 3B - (Group Mate)
Hamza Altaf (048), Sara Sultan (063), Ossama
Mehmood (069)