



# GraphicEra

HILL UNIVERSITY

Established by an Act of the State Legislature of Uttarakhand (Adhiniyam Sankhya 12 of 2011)

**MINI PROJECT**  
**on**  
**A COMPLETE ERP**  
**SYSTEM**  
**(CSE IV Semester Mini Project  
report)**

**Submitted to:**

Mr. Akash Chauhan

(CC-CSE-D1-IV-Sem)

**Submitted by:**

Name: Daksh Purohit

Roll. No:2218650

CSE-D1-IVSem

# **TABLE OF CONTENTS**

<b>S NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
1.	Introduction	3-4
2.	Hardware/System Requirements	5
3.	Utilities	6-7
4.	Flowchart	8
5.	Snapshots	9-12
6.	Conclusion	13

# **CHAPTER-1**

## **INTRODUCTION**

### **Overview:-**

The ERP Portal project is a comprehensive web-based application designed to facilitate user authentication and profile management within an enterprise resource planning (ERP) system. Leveraging Node.js and Express.js for the backend, it efficiently handles user sessions and database interactions with MySQL, while the frontend, built with HTML, CSS, and JavaScript, provides a seamless and interactive user experience. The application features secure login and signup processes, user profile updates, and personalized greetings. This project serves as a practical educational tool for understanding full-stack development, database management, and web security, while also laying the groundwork for scalable and extensible ERP systems. Its modular design and emphasis on user experience and security make it a robust solution for real-world business applications, addressing common web application needs and preparing developers for market demands.

### **Motivation Behind This Project:-**

The ERP Portal project is a web-based application designed to manage user authentication and profiles, forming a foundational component of an enterprise resource planning (ERP) system. Built with Node.js, Express.js, HTML, CSS, JavaScript, and MySQL, it provides login, signup, and profile management functionalities. The project aims to enhance learning in full-stack development, database management, and web security, while offering a user-friendly interface for seamless navigation and personalization. It serves as a practical and scalable base for more complex ERP systems, addressing real-world application needs and showcasing best practices in security and user experience.

## **Features:-**

### **User Authentication:**

- 1.Login: Users can log in using their username and password.
- 2.Signup: New users can register by providing a username, password, roll no,10<sup>th</sup> percentage, 12<sup>th</sup> percentage and their profile photo.
- 3.Session Management: User sessions are managed to keep track of authenticated users.

### **User Profile Management:**

- 1.Profile Update: Authenticated users can update their profile information, such as username and password.
- 2.Profile Display: The application greets the logged-in user by displaying their profile photo, username, roll no,10<sup>th</sup> percentage, 12<sup>th</sup> percentage .

The homepage provides options for users to navigate to the login or signup pages.

# **CHAPTER-2**

## **Hardware/System Requirements**

### **Hardware Requirements**

**Processor:** Minimum of 1 GHz CPU (2 GHz recommended)

**RAM:** Minimum of 1 GB (2 GB or more recommended for better performance)

**Storage:** At least 500 MB of free disk space for the project files and database

### **Software Requirements**

#### **1.Operating System:**

- Windows 7 or later
- macOS 10.12 or later
- Linux (any modern distribution)

#### **2.Node.js:**

- Version 14.x or later
- Node Package Manager (npm) included with Node.js

#### **3.Database:**

- MySQL 5.7 or later

#### **4.Web Browser:**

- Latest version of Google Chrome, Mozilla Firefox, Microsoft Edge, or Safari

#### **5.Development Tools:**

- Code Editor (e.g., Visual Studio Code, Sublime Text, Atom)

# **CHAPTER-3**

## **Utility Files:-**

### **Frontend**

The frontend consists of static HTML files, enhanced with CSS for styling and JavaScript for dynamic functionality:

- **HTML Files:**
  - **index.html:** The homepage offering navigation to login and signup.
  - **login.html:** The login page with a form for user credentials.
  - **signup.html:** The signup page for new user registration.
  - **profile.html:** The profile page allowing users to view their details.
  
- **JavaScript:**
  - Uses Fetch API to send asynchronous requests to the backend.
  - Handles form submissions, displays success or error messages, and manages redirections upon successful operations.
  
- **CSS:**
  - Provides basic styling to enhance the user interface, making the forms more user-friendly and visually appealing.

## **Database Schema**

- **Database:** erp\_portal
- **Table:** users
  - **Columns:**
    - id (INT, AUTO\_INCREMENT, PRIMARY KEY): Unique identifier for each user.
    - username (VARCHAR): The username of the user.
    - password (VARCHAR): The password of the user.
    - roll\_number VARCHAR(20): The roll number of the user.
    - percentage\_10th DECIMAL(5,2): The percentage of 10<sup>th</sup> class.
    - percentage\_12th DECIMAL(5,2): The percentage of 12<sup>th</sup> class.
    - profile\_image LONGBLOB: Contains image of the user in LONGBLOB

## **Security Considerations**

### **Session Security:**

- The session cookie should be configured with secure and HTTP.
- Only flags in a production environment to prevent security vulnerabilities.

### **Image Storage:**

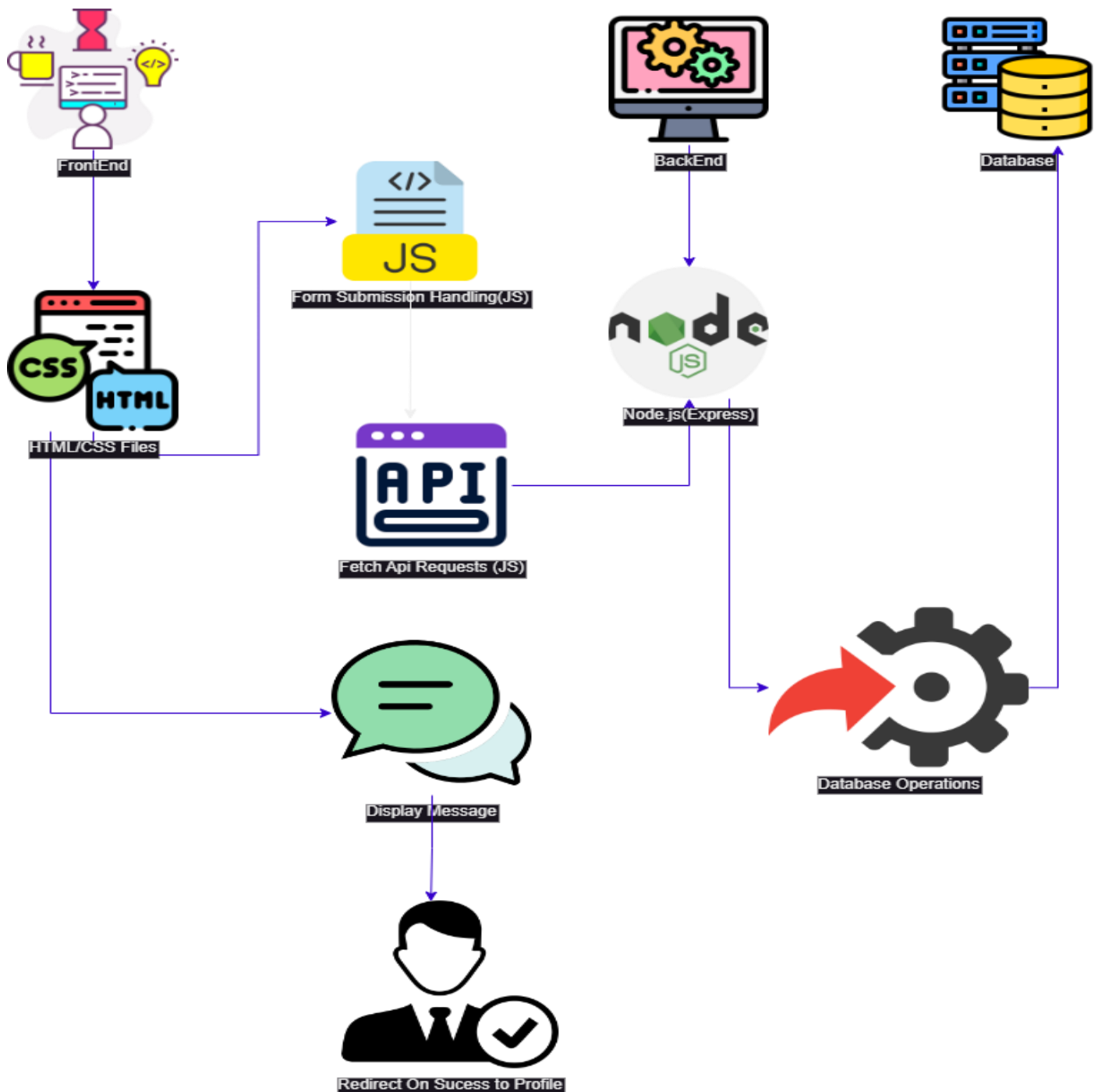
- Stored Image in LONGBLOB format .

### **Dependencies (npm packages)**

- **express:** A fast, minimalist web framework for Node.js used to create web applications and APIs.
- **mysql:** A module for interacting with MySQL databases from a Node.js application, enabling SQL queries and CRUD operations.
- **path:** Provides utilities for handling and transforming file and directory paths across different operating systems.
- **express-session:** Middleware for Express to manage user sessions, storing data between HTTP requests.
- **multer:** Middleware for handling multipart/form-data, mainly used for file uploads in an Express application.
- **crypto:** Provides cryptographic functionality, including hashing, HMAC, encryption, and decryption using OpenSSL wrappers.

## CHAPTER-4

### Flowchart:-

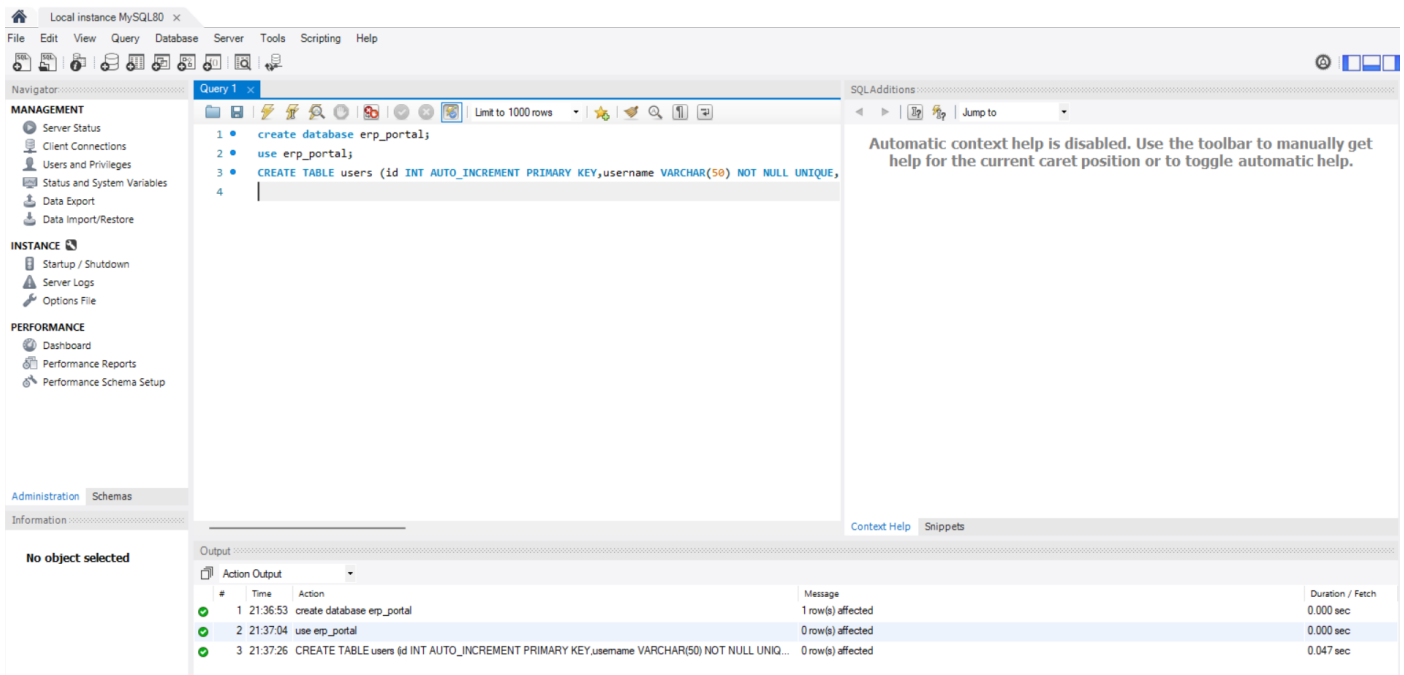




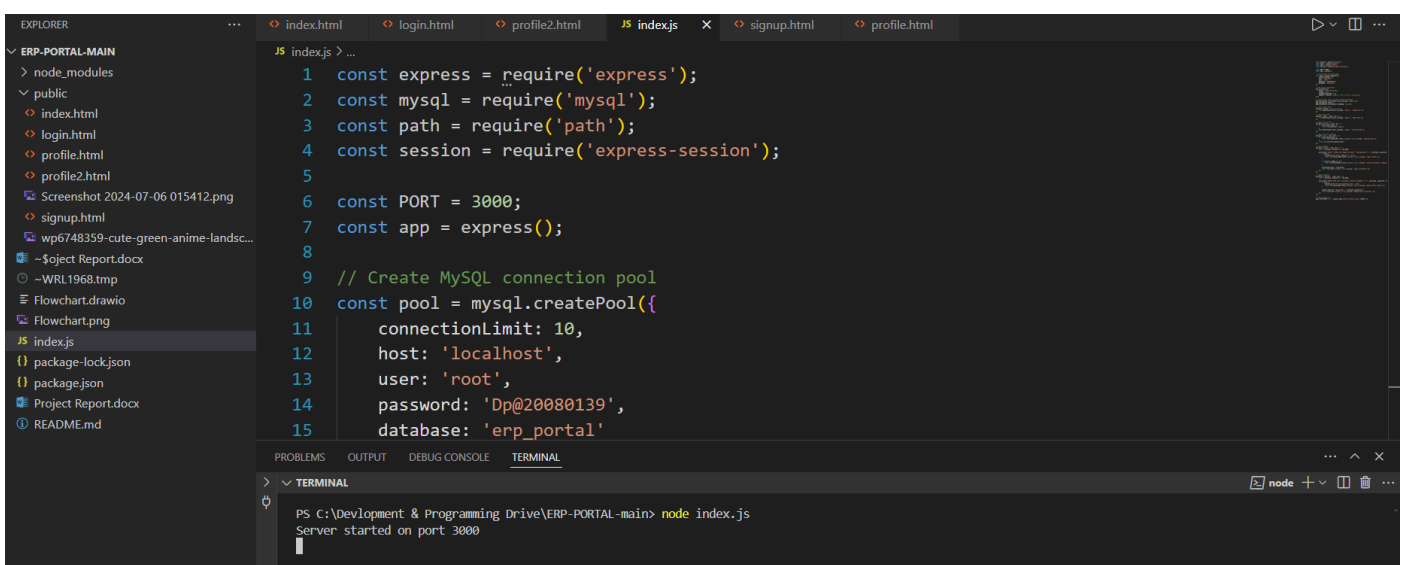
# CHAPTER-5

## SNAPSHOTS

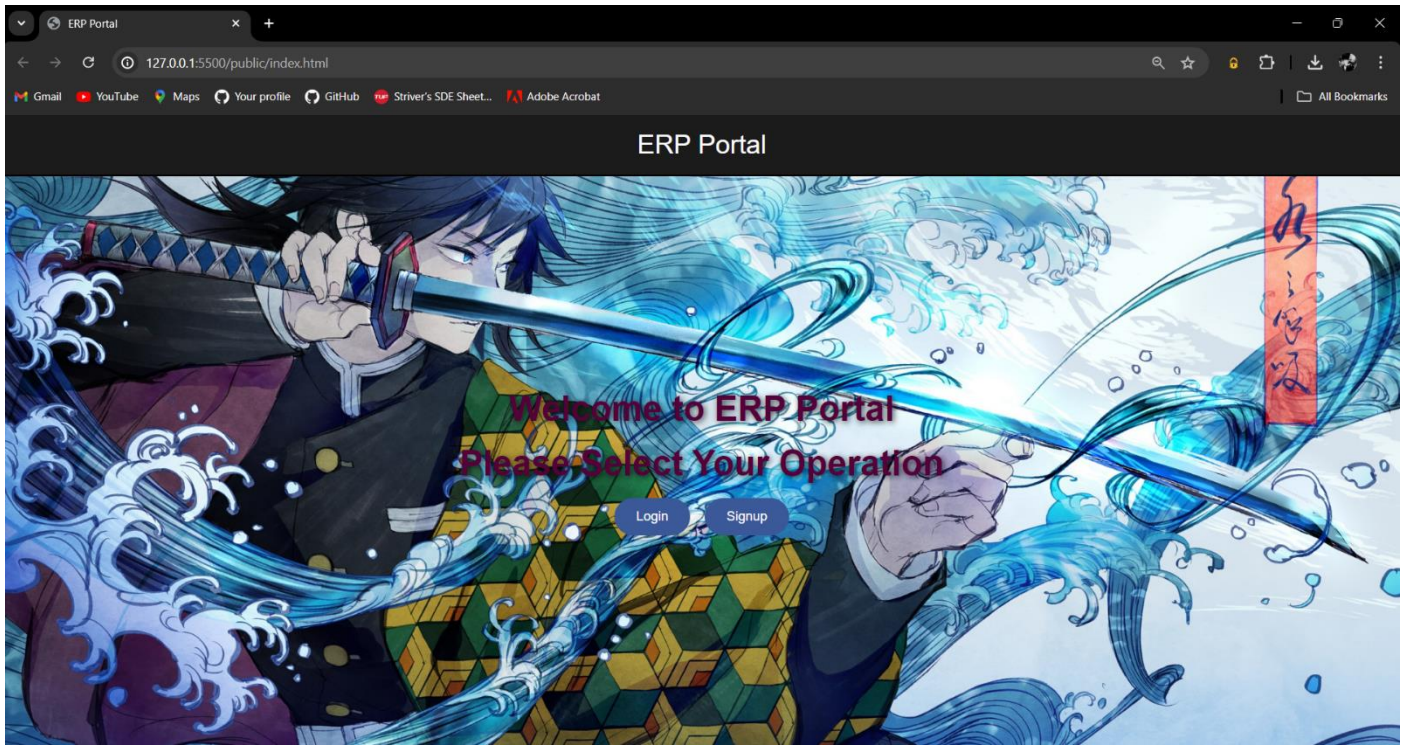
1.Create a database named `erp_portal` and use it . Create a table `users` with following parameters



2.Open the folder and run the `index.js` file and start the sever on port 3000



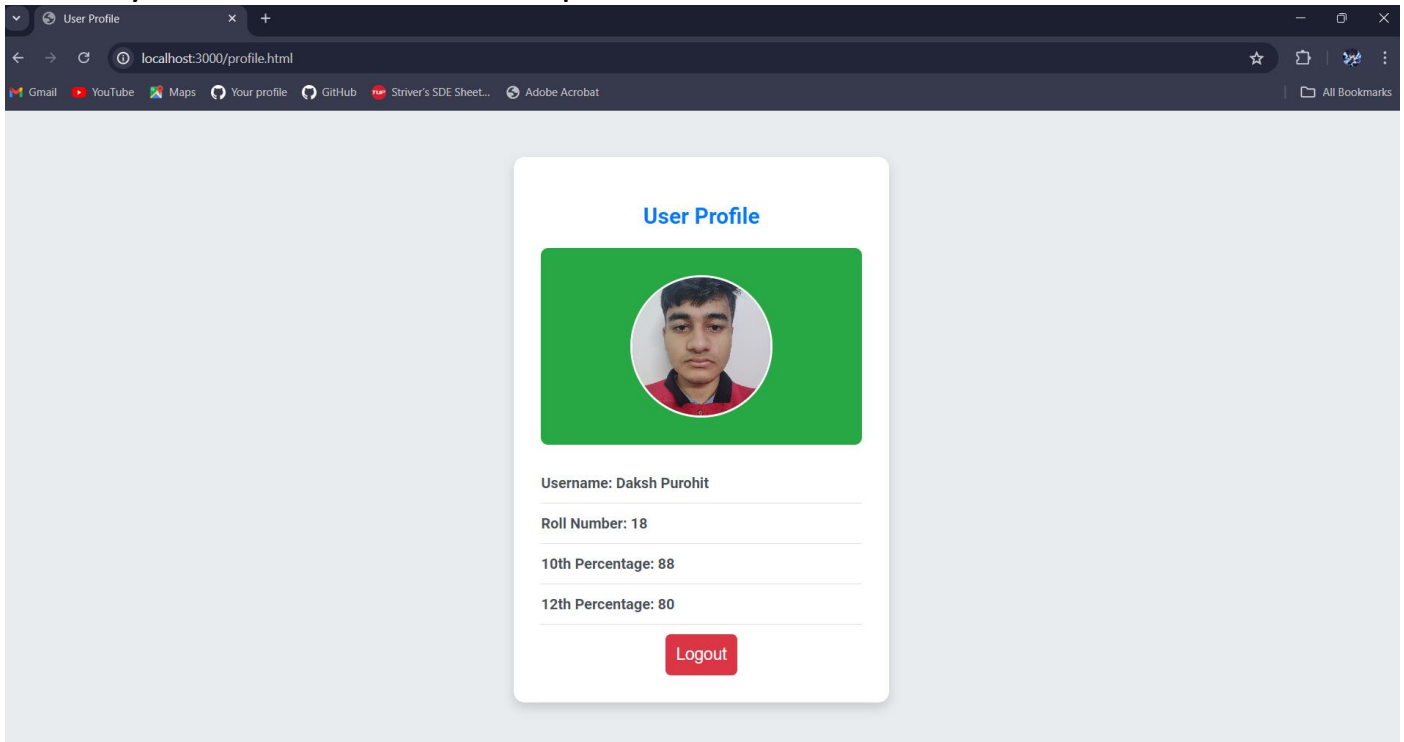
3. Open any browser and use type in search url  
(<http://localhost:3000/index.html>)



4. Select Signup option and you will be redirected to signup.html type the credentials and they will be updated on table in database

A screenshot of a web browser displaying the 'Signup' form. The browser's address bar shows 'localhost:3000/signup.html'. The form is titled 'Signup' and contains the following fields: 'Username' (with the value 'Daksh Purohit'), 'Password' (with masked characters '.....'), 'Roll Number' (with the value '18'), '10th Percentage' (with the value '88'), and '12th Percentage' (with the value '80'). There is also a 'Profile Image' section with a 'Choose File' button and a file name 'WhatsApp Imag...\_b77db634.jpg'. At the bottom of the form is a green 'Signup' button.

## 5. Now you will be redirected to profile.html



## 6. To view the table in Database use query ***select \* from users;***

The screenshot shows the MySQL Workbench interface. The 'Query Editor' window contains the following SQL queries:

```
1 • create database erp_portal;
2 • use erp_portal;
3 • CREATE TABLE users (id INT AUTO_INCREMENT PRIMARY KEY, username VARCHAR(50) NOT NULL UNIQUE, password VARCHAR(255) NOT NULL, roll_number VARCHAR(20)
4 • select * from users;
```

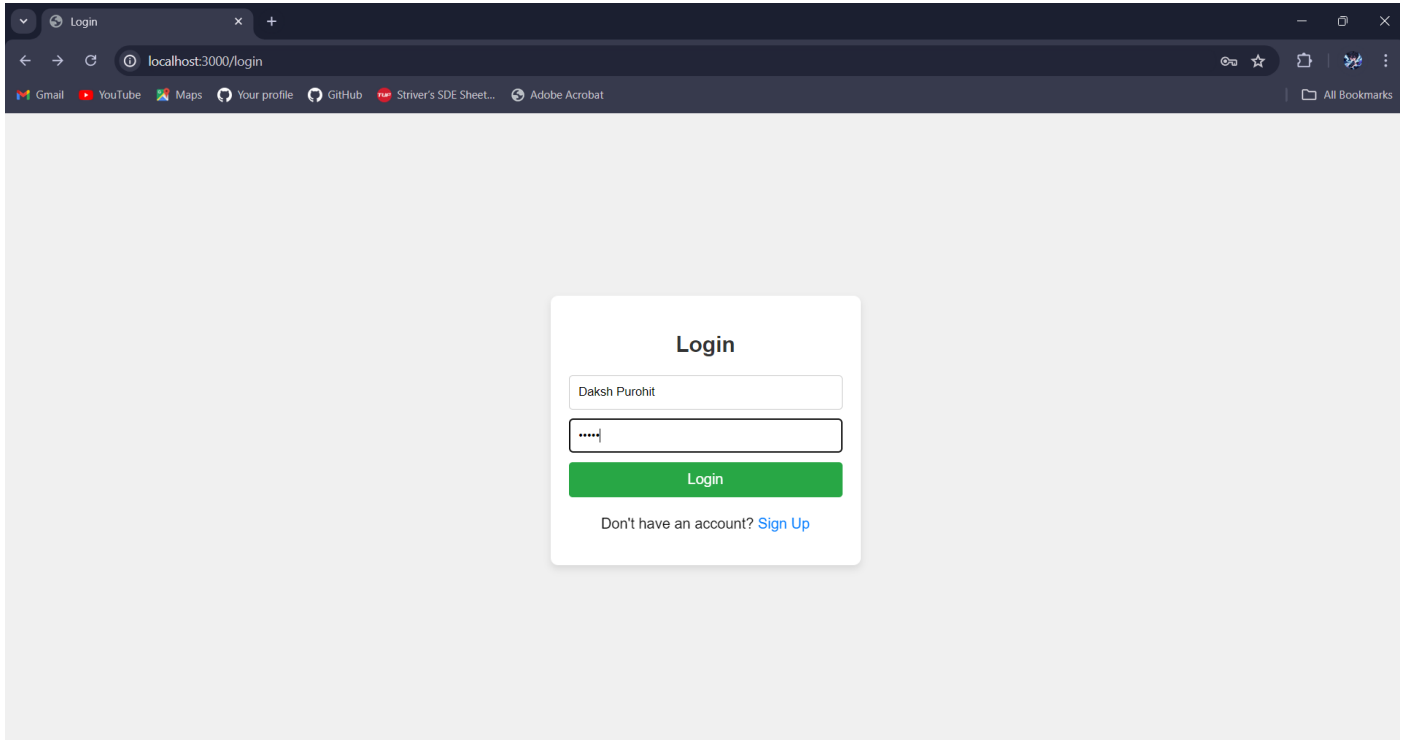
The 'Result Grid' shows the output of the query:

#	id	username	password	roll_number	percentage_10th	percentage_12th	profile_image
1	1	Daksh Purohit	12345	18	88.00	80.00	image

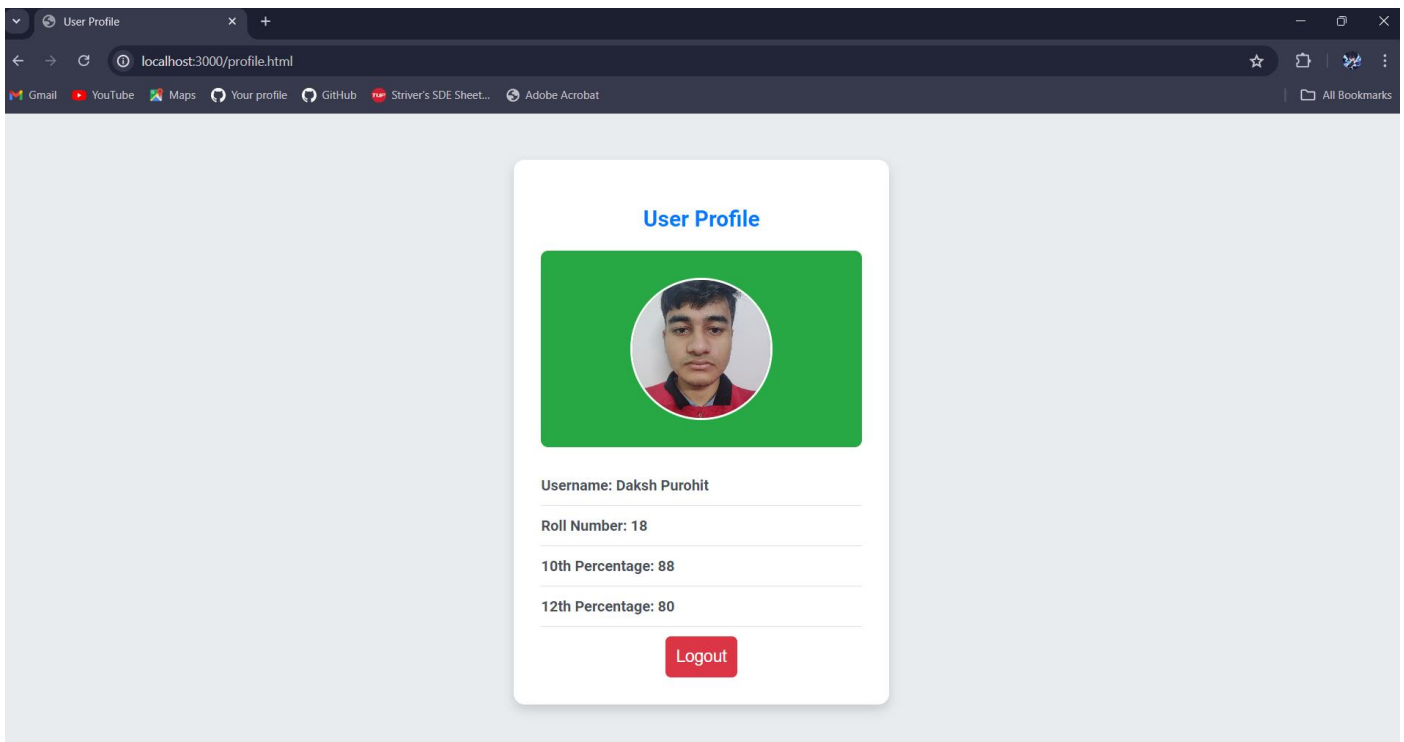
The 'Output' window shows the execution log:

#	Time	Action	Message	Duration / Fetch
1	21:36:53	create database erp_portal	1 row(s) affected	0.000 sec
2	21:37:04	use erp_portal	0 row(s) affected	0.000 sec
3	21:37:26	CREATE TABLE users (id INT AUTO_INCREMENT PRIMARY KEY, username VARCHAR(50) NOT NULL UNIQUE, password VARCHAR(255) NOT NULL, roll_number VARCHAR(20)	0 row(s) affected	0.047 sec
4	21:41:40	select * from users LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

6.Now click logout and you will be redirected to index.html and use login option



7.You will be redirected to profile.html which proves data stored in database is used for credentials which was sent while using signup



## **CHAPTER-6**

### **CONCLUSION**

The ERP Portal project successfully demonstrates the development of a fundamental web-based application for user authentication and profile management, integral to any enterprise resource planning (ERP) system. Through this project, key skills in full-stack development were honed, utilizing Node.js and Express.js for the backend, MySQL for database management, and HTML, CSS, and JavaScript for the frontend. The application features secure login and signup functionalities, user profile updates, and personalized greetings, providing a robust and user-friendly experience. This project not only serves as a practical educational tool but also lays a strong foundation for building more complex and scalable ERP systems. It emphasizes the importance of web security, efficient database interaction, and seamless user navigation, making it a valuable resource for both learning and practical application in real-world business environments.

