

DBMS : MICRO-PROJECT

Student Management System Report

Introduction

This report presents a comprehensive overview of a Student Management System (SMS) developed as a DBMS microproject. The system leverages Python's Tkinter library for the graphical user interface (GUI) and MySQL for database management. The main focus of this report is on the database design and functionalities implemented to manage student data efficiently.

Sytem Overview and Detailed Functionalities

System Overview

The Student Management System consists of two main components:

1. Login System
2. Student Management Interface

Login System

The login system ensures secure access to the student management functionalities. The predefined credentials are:

Username: ADMIN

Password: 1234

Upon successful login, the user is granted access to the main student management interface. Incorrect credentials or empty fields trigger error messages.

Student Management Interface

The student management interface allows users to perform various operations on student data, including:

- Adding a Student
- Updating Student Information
- Deleting a Student
- Searching for a Student
- Exporting Student Data
- Viewing All Students

Database Schema

The database, designed using MySQL, includes a single database named studentmanagementsystem with one table, student. The schema for the student table is as follows:

id: int not null primary key	gender: varchar(20)
name: varchar(50)	dob: varchar(20)
mobile: varchar(15)	added_date: varchar(20)
email: varchar(50)	added_time: varchar(20)
address: varchar(100)	

Database Operations

1. Connecting to the Database

The system connects to the MySQL database using the pymysql library in Python. It checks for the existence of the studentmanagementsystem database and the student table, creating them if necessary.

2. Adding a Student

The system collects data from the user and executes an INSERT query to add a new record to the student table, ensuring all required fields are populated and the student ID is unique.

3. Updating Student Information

To update an existing student record, the system retrieves the student ID and the new data from the user, executing an UPDATE query to modify the existing record.

4. Deleting a Student

To delete a student record, the system retrieves the student ID from the user and executes a DELETE query to remove the record from the student table.

5. Searching for a Student

The system allows users to search for student records based on various criteria, executing a SELECT query to retrieve matching records.

6. Exporting Student Data

The system provides an option to export student data to a CSV file. It retrieves all records from the student table and writes them to a CSV file using the pandas library in Python.

User Interface Design

The user interface, designed using Tkinter, provides a modern and intuitive experience. The main components include:

- **Login Screen:** A simple login form with fields for username and password.
- **Main Window:** The main application window with buttons for various operations and a table to display student records.
- **Forms for Data Entry:** Separate forms for adding, updating, and searching student data.

Security Features

To ensure data security and integrity, the system includes:

1. **Login Authentication:** Prevents unauthorized access.
2. **Confirmation Prompts:** Ensures critical actions like deletion are confirmed by the user.
3. **Data Validation:** Ensures all required fields are filled before adding or updating records.

Conclusion

The Student Management System is an effective tool for managing student data, leveraging a robust database design and a user-friendly interface. The integration of security features ensures data integrity and prevents unauthorized access.

References and Credits

1. MySQL Documentation: MySQL 8.0 Reference Manual.
Retrieved from <https://dev.mysql.com/doc/>
2. Pandas Documentation: pandas 1.2.3 documentation.
Retrieved from https://pandas.pydata.org/pandas-docs/version/1.2.3/user_guide/index.html
3. Tkinter Documentation: Tkinter 8.5 reference: a GUI for Python. Retrieved from
<https://docs.python.org/3/library/tkinter.html>
4. PyMySQL Documentation: PyMySQL 1.0.2 documentation.
Retrieved from <https://pymysql.readthedocs.io/en/latest/>
5. Online Tutorials and Resources: A special thanks to online tutorials and forums that provided valuable insights and code snippets.

[YouTube Playlist](#)

Images and Icons

Background Image: Downloaded from Google.

Icons: Downloaded from [Flaticon](#).

Music

Demonstration Video Music: Downloaded from [YouTube](#).