Syed Nazmul Islam
ID: IT-22023
Session: 2021-2022(re-admission)
Course: ICT-1200(1st year 2nd semester project)
Project Report: Bus Timing Management System Using C++ and SQLite

---

1. Introduction

The **Bus Timing Management System** is a simple program designed to help users check bus schedules and view the next available bus. This system is developed using **C++** and **SQLite** to manage and store bus schedules and login information. It provides a user-friendly interface for both bus passengers and the admin panel to view, add, and remove bus timings.

---

2. Objective

The main objective of the project is to build a simple and efficient bus scheduling system that can:

- Show next bus schedule time.
- Allow the admin to add or remove bus schedules.
- Manage login for admin to do changes in schedules.

---

3. System Features

- **Bus Schedule Viewing**: Users can view the next available bus based on the current time.
- **Admin Login System**: Admin can add or remove bus schedules after login with their password.
- **Data Storage**: Bus timings and admin login information's are stored in an SQLite database.

---

4. Technologies Used

- **C++**: Used for coding the program.
- **SQLite**: A database used to store bus timings and login information.
- **CodeBlocks**: The code editor used for programming c++.
- **SQLite3 Library**: Integrated with C++ to manage database operations.

---

5. SQLite Database

Two tables are used to store information:

1. **bus Table**:
    - time (INT): Stores the bus departure time in HHMM format.
    - origin (VARCHAR (50)): Stores the departure location of the bus.
    - destination (VARCHAR(50): Stores the destination location of the bus.
    -

***SQL Query**: CREATE TABLE IF NOT EXISTS bus (time INT, origin VARCHAR(50), destination VARCHAR(50));*

2. **login info Table**:
    - username (VARCHAR (50)): Stores the username for login.
    - password (INT): Stores the password as an integer.

      ***SQL Query**: CREATE TABLE IF NOT EXISTS logininfo(username VARCHAR(50), password INT);*

---

6. Key Functionalities

1. **Next Bus Display**:
    - Displays the next available bus that is departing after the current time.
    - Uses current time to filter bus departing time.
    - Displays information such as the departure time, origin, and destination.
2. **Login System**:
    - Verifies user username and password from the database.
    - If the admin is not registered, the system do registration.
    - After login, the admin can access functionalities to add or remove buses.
3. **Admin Panel**:
    - **Add New Bus**: Admin can add new bus schedules by giving the time, origin, and destination.
    - **Remove Bus**: Admin can remove a bus from the schedule by specifying the bus's departure time in HHMM format.
4. **Logout and Termination**:
    - Admin can log out of the system and return to the main menu.
    - Users can terminate the entire program from the menu.

---

7. Code Structure

- **Main Function**: Initializes the SQLite database, creates necessary tables, and continuously displays the menu for user interaction.
- **Menu Function (menubar)**: Displays the main menu and navigates the user to different functions based on their selection.
- **Bus Management Functions**:
    - **nextbus()**: Displays the next available buses from the database.
    - **addnewbus()**: Adds a new bus schedule into the bus table.
    - **removebus()**: Removes a bus schedule from the bus table based on the time provided.
- **Login and Registration Functions**:
    - **login()**:Ask for username and password, checks if they exist in the logininfo table.
    - **registerUser()**: Registers a new admin user if none exist in the system.
    - **isUserRegistered()**: Checks if there is an existing user in the logininfo table.
- **Utility Functions**:
    - **curtime()**: Gets the current time in HHMM format to compare with bus timings.

---

8. Challenges and Improvements

Challenges:

1. Making connection between SQLite database and c++ was challenge for me .As I was using database for first time. I had to spend a lot of time to using SQL on c++ code . Setting MySQL with CodeBlocks seems very difficult to me . so I used SQLIte database.
2. I got some difficulty while creating login functionality. As for some buffer was hitting to my code . Every time I got wrong password error while debugging.

Improvements:

1. The process of storing password is insecure. In future, password hashing should be implemented for better security.

2. The interface could be expanded to support more user-friendly features, such as searching buses by origin and destination.

3.In future we can develop a GUI interface for better user experience.

---

9. Conclusion

The **Bus Timing Management System** does its job in small environments like ours one work for **MBSTU** Campus. Using **SQLite** database make it more useful. In farther it can be developed by GUI Interface. It successfully does login for changing schedule information.

---

10. Future Scope

There is a huge scope for developing this program. We can use hashing technique for user authentication. It will secure the system more efficiently. We can develop a mobile app using an android system. We may add some search functionality to develop user experience. We may give a GUI interface for better user experience.

---

11. Appendix

Tools:

1. C++ Compiler (G++/MinGW)
2. SQLite3 Library
3. CodeBlocks