Project: Banking Portal System

Python & MySQL

Study Course

B207 Cyber Security

By

Chehab Hany Mohamed Elsayed Elsayed Elmenoufi

Student Number: GH1034223

Under the Guidance of

Prof. Sami Alsalamin

GitHub URL: https://github.com/Chippo90/Cyber-Security/tree/main

Video Recording URL: https://youtu.be/yXnfBkVD7wg

Gisma University of Applied Sciences

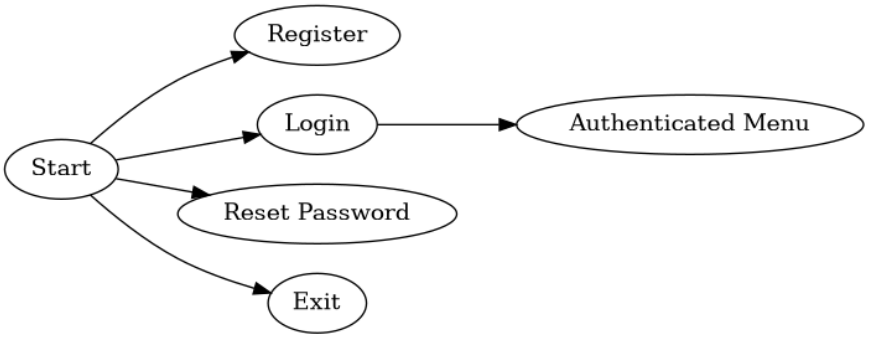
Berlin, Germany

June 2025

Table of Contents

1. Introduction

This project is a basic banking application in Python with MySQL integration. It allows users to register, log in and manage their account. It provides security with hashing and validation, and it integrates with a MySQL database for data storage.



2. Imports

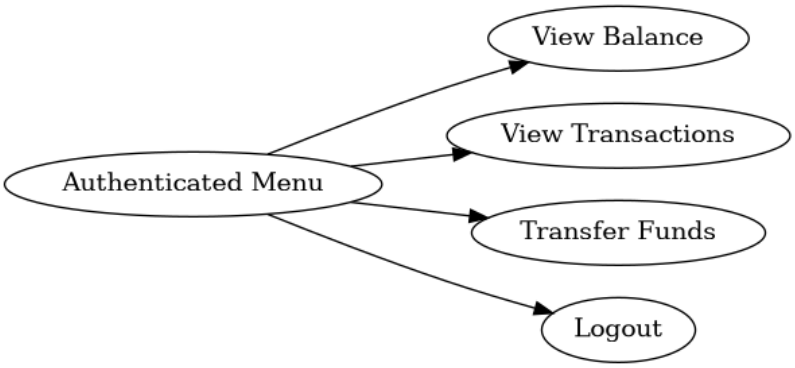
- **mysql.connector:** Database connectivity.(*MySQL :: MySQL Connector/Python Developer Guide :: 5.1 Connecting to MySQL Using Connector/Python*, no date)
- **bcrypt:** Secure password hashing.(Murugan, 2024)
- **configparser:** Read database configuration file.(*configparser — Configuration file parser*, no date)
- **datetime:** Timestamping transactions.(*datetime — Basic date and time types*, no date)
- **re:** Password strength validation.(*re — Regular expression operations*, no date)
- **sys:** program exit control.(*sys — System-specific parameters and functions*, no date)

3. File Description

File Name	Description
banking_app.py	Main application script.
db_config.ini	Database configuration file for MySQL credentials.
database.sql	MySQL database creation and main tables('B103 - Databases and Big Data - SQL Queries.pdf', no date)

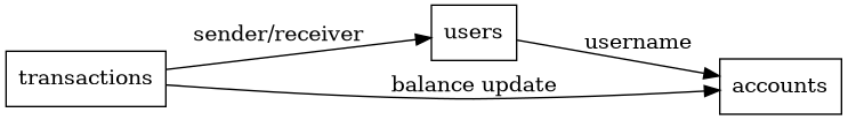
4. Key Features

- **Registration:** Require password strength and stores hashed passwords.
- **Login:** Authenticate customers.
- **Account Management:**
    - View Current Balance.
    - View Transaction History.
    - Transfer Funds.
- **Password Reset:** Simulated token password reset.



5. Implementation

- **Password Security:** Passwords are hashed before being stored in the database.
- **Database Operations:** All reactions with the database are done using queries to enhance security.



- **Transactions:** Every fund transfer is recorded with a time for tracing.

6. Conclusion and Future Work

The application successfully shows a secure banking system with many benefits and opens the door for further development.

Future Work:

- Implement multi factor authentication.
- Add email password reset.
- Create a user interface for better visuality.

7. References

‘B103 - Databases and Big Data - SQL Queries.pdf’ (no date).

configparser — Configuration file parser (no date) *Python documentation*. Available at: https://docs.python.org/3/library/configparser.html (Accessed: 11 June 2025).

datetime — Basic date and time types (no date) *Python documentation*. Available at: https://docs.python.org/3/library/datetime.html (Accessed: 11 June 2025).

Murugan, S. (2024) ‘Hashing Passwords Using the Top 5 Python Libraries’, *Top Python Libraries*, 13 December. Available at: https://medium.com/top-python-libraries/hashing-passwords-using-the-top-5-python-libraries-5ec530973b17 (Accessed: 11 June 2025).

MySQL :: MySQL Connector/Python Developer Guide :: 5.1 Connecting to MySQL Using Connector/Python (no date). Available at: https://dev.mysql.com/doc/connector-python/en/connector-python-example-connecting.html (Accessed: 11 June 2025).

re — Regular expression operations (no date) *Python documentation*. Available at: https://docs.python.org/3/library/re.html (Accessed: 11 June 2025).

sys — System-specific parameters and functions (no date) *Python documentation*. Available at: https://docs.python.org/3/library/sys.html (Accessed: 11 June 2025).