
# Security Fundamentals and Development (H7SFD)

## BSHC3A

## CA1 – Group Project Report

**Group:** N

**Members:**

Emma Kelly, X21385283

Jane Keyes, X22121358

Aisha Ntuli, X21341006

Kevin Osifo, X21717211

Ayomide Idowu, X21742591

Lecturer: Dr. Vanessa Ayala-Rivera


## National College of Ireland

## Project Submission Sheet

| | | | |
|---|---|---|---|
| **Student Name:** | Emma Kelly | | |
| **Student ID:** | X21385283 | | |
| **Programme:** | Computing | **Year:** | 3rd |
| **Module:** | Security Fundamentals and Development | | |
| **Lecturer:** | Vanessa Ayala-Rivera | | |
| **Submission Due Date:** | 19/11/2023 | | |
| **Project Title:** | Password Manager | | |
| **Word Count:** | 1,409 Words | | |

**I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.**
**ALL internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.**

| | |
|---|---|
| **Signature:** | Emma Kelly |
| **Date:** | 15/11/2023 |

**PLEASE READ THE FOLLOWING INSTRUCTIONS:**

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

| Office Use Only | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# AI Acknowledgement Supplement

## I. Security Fundamentals and Development

## II. Password Manager

| Your Name/Student Number | Course | Date |
|---|---|---|
| **Emma Kelly** | Computing | 17/11/2023 |
| **Aisha Ntuli** | Computing | 17/11/2023 |
| **Jane Keyes** | Computing | 17/11/2023 |
| **Kevin Osifo** | Computing | 17/11/2023 |
| **Ayomide Idowu** | Computing | 17/11/2023 |

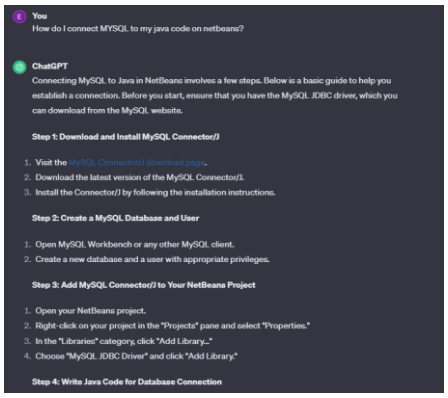
## AI Acknowledgment

| Tool Name | Brief Description | Link to tool |
|---|---|---|
| **ChatGPT** | Brief coding tutorials | https://chat.openai.com/ |

## **III.** Description of AI Usage

| ChatGPT | |
|---|---|
| **Brief coding tutorials** | |
| How do I connect my database to my code? | [tutorial on how to connect database to code] |

## **IV.** Evidence of AI Usage



You
How do I connect MYSQL to my java code on netbeans?

ChatGPT
Connecting MySQL to Java in NetBeans involves a few steps. Below is a basic guide to help you establish a connection. Before you start, ensure that you have the MySQL JDBC driver, which you can download from the MySQL website.

**Step 1: Download and Install MySQL Connector/J**

1. Visit the MySQL Connector/J download page.
2. Download the latest version of the MySQL Connector/J.
3. Install the Connector/J by following the installation instructions.

**Step 2: Create a MySQL Database and User**

1. Open MySQL Workbench or any other MySQL client.
2. Create a new database and a user with appropriate privileges.

**Step 3: Add MySQL Connector/J to Your NetBeans Project**

1. Open your NetBeans project.
2. Right-click on your project in the "Projects" pane and select "Properties."
3. In the "Libraries" category, click "Add Library..."
4. Choose "MySQL JDBC Driver" and click "Add Library."

**Step 4: Write Java Code for Database Connection**

## 1. Contributions of each member of the Group

| StudentID | StudentName | Tasks |
|---|---|---|
| X21385283 | Emma Kelly | Demonstration Video, Group Report, Investigation and application Testing, Code Commenting |
| X22121358 | Jane Keyes | Investigation, Login GUI bug fixes and testing |
| X21341006 | Aisha Ntuli | Investigation, PasswordManagerApp bug fixes |
| X21742591 | Ayomide Idowu | Business Functionality, Technical Implementation writing of the login Use Case in the brief |
| X21717211 | Kevin Osifo | GUI and Application Coding & Commenting, MYSQL database, Testing |

## 2. Link to the demo video

https://github.com/emmakelly3001/PasswordManager_final/tree/main/Video%20Demo

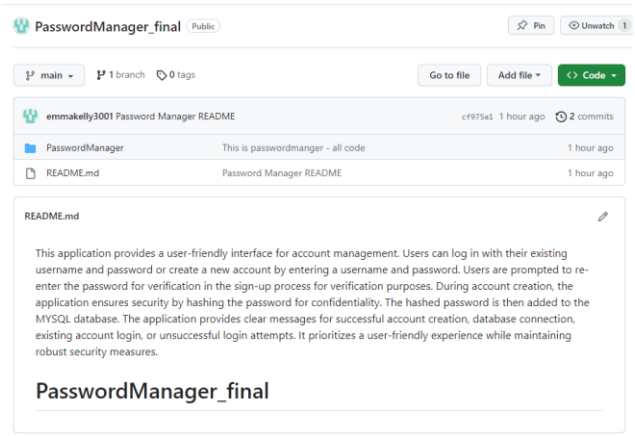
## 3. Link to the GitHub project

https://github.com/emmakelly3001/PasswordManager_final

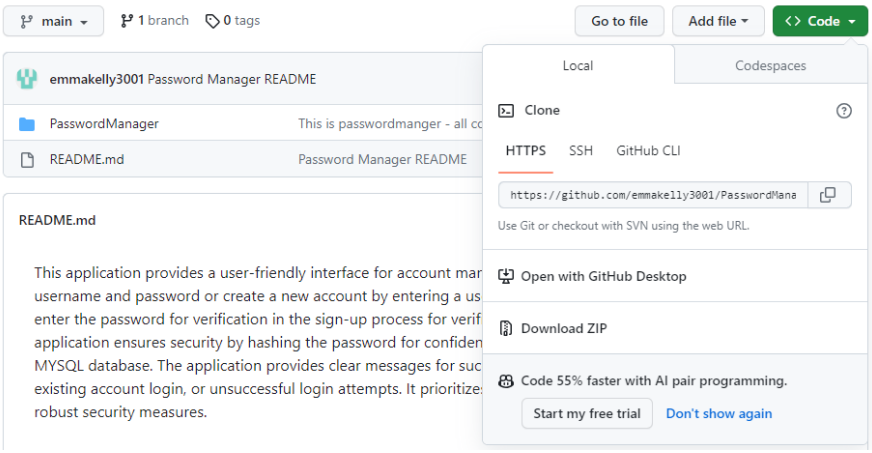
## 4. Manual

Describe the instructions to download, install and run your application.

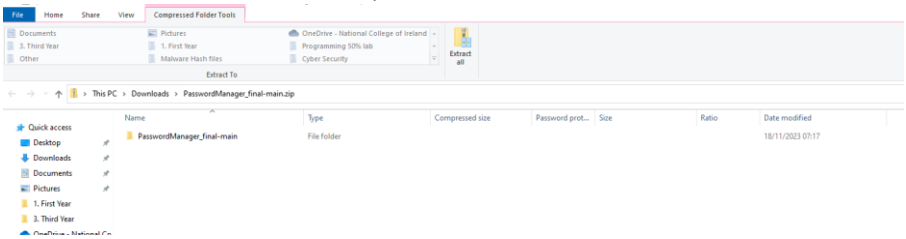
1. Click the GitHub link above. This will bring you to this page.



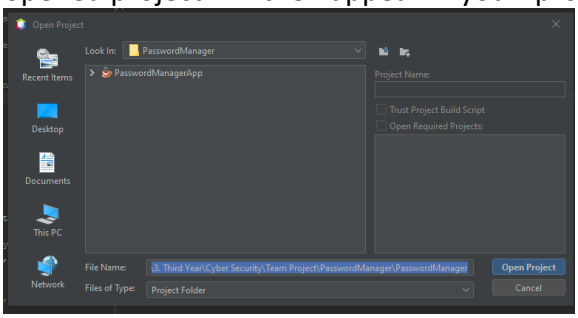

2. Click on ‘code’ > ‘download zip’

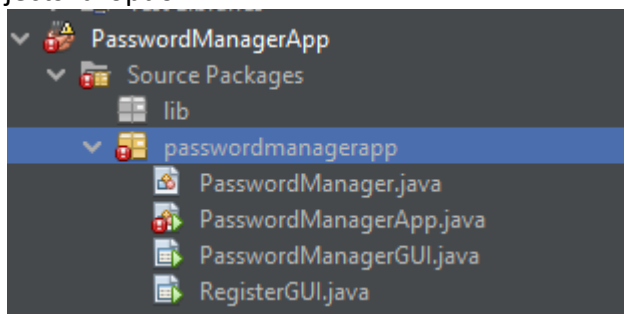



3. Click on the downloaded file, then ‘extract all’

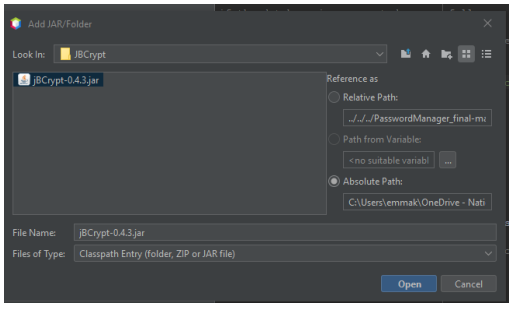


4. After the zipped folder has been extracted to the folder of your choice, go into NetBeans > file> open project. You will then have to find the extracted file in the given prompt. > Open Project. The opened project will then appear in your ‘projects’ dropdown.

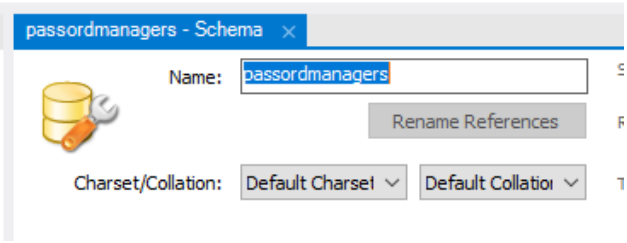


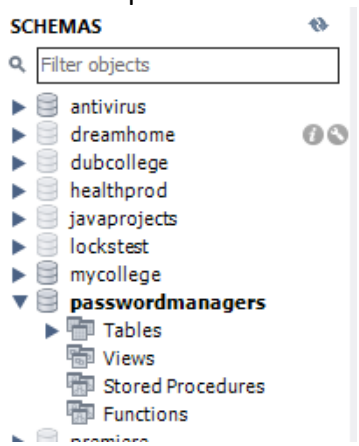

5. In the ‘projects’ dropdown, you will see your project. You will see different packages. One package is containing all of your files, there is another file called ‘libraries’. This includes the JDK file you are using. Right click on the ‘libraries’ file and click ‘add JAR/folder’. You will then navigate back to the unzipped folder that the code was in, and locate the JBCrypt file. Click on that file, and it will be added to your project.



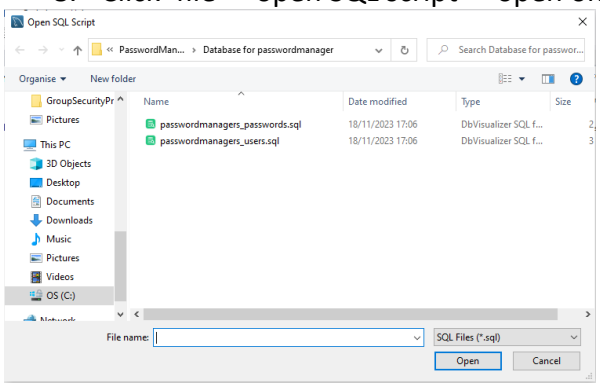
6. Open MYSQL > Click into any connector you have> right click the white space underneath all of your schemas > click ‘create schema’ > name the schema ‘passwordmanagers’ > click ‘apply’ and then ‘finish’



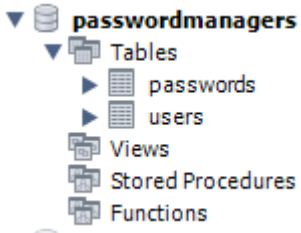

7. Beside the ‘schemas’ tab, there is a refresh button. Click ‘refresh’ and your schema should appear in the dropdown.



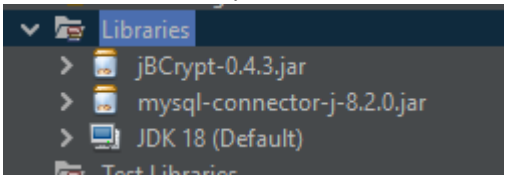

8. Click ‘file’> ‘open SQL script’ > open one your scripts in the given explorer> click ‘open’



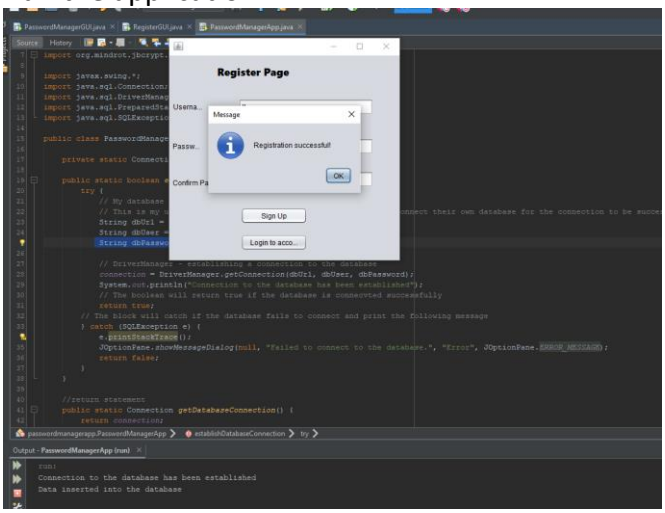

9. Click the yellow lightening button to load the table > reload the schema and the table should be added to your new schema. Do the same for both files.




10. Go NetBeans, and like you did to connect the JBCrypt, connect the MYSQL Connector. Right click libraries> add JAR/folder>Find the SQL Connector and add it. It will then be displayed as below.



11. In the ‘PasswordManagerApp’, change the line of code ‘String dbPassword =’ to your personal root password.
12. Run the application.




## 5. Summary of the Application

This application provides a user-friendly interface for account management.
Users can log in with their existing username and password or create a new account by entering a username and password.
Users are prompted to re-enter the password for verification in the sign-up process for verification purposes.
During account creation, the application ensures security by hashing the password for confidentiality.
The hashed password is then added to the MYSQL database.
The application provides clear messages for successful account creation, database connection, existing account login, or unsuccessful login attempts.
It prioritizes a user-friendly experience while maintaining robust security measures.

# **Chosen Use Case Scenario:** User (Brian) Login

### 5.1 Business Functionality

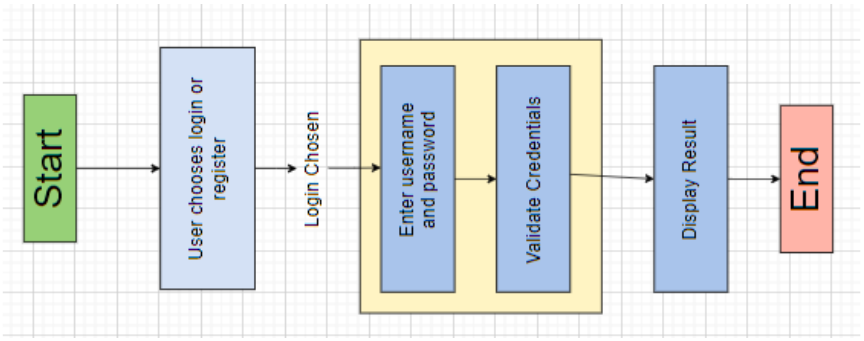
The purpose of this Application is to stores users' passwords safely that supports authentication by securely hashing and verifying user passwords to protect user accounts. There are two paths a user can take to enter their account, "Register page" or "Log into your account" The Register page page allows the user to enter their desired username and password, then to confirm their password with the text fields provided; "utF, "pF" and "cPF". With these inputted correctly the users information is then stored in the database and the password is hashed with the use of JBCrypt.

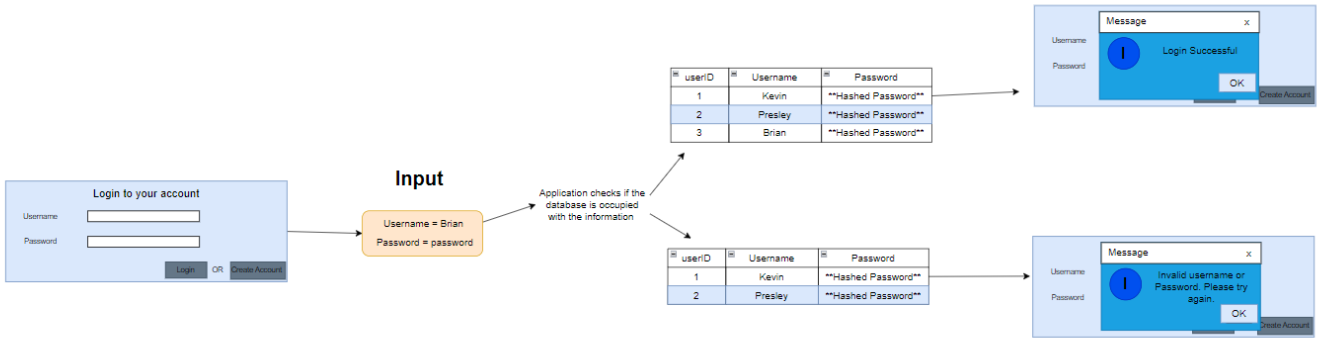
The Log into your account page allows the user to log into their account which has been made, if credentials are correct the user should successfully log into their account. The process is made secure with hashing as even in the database the passwords appear hashed.

### 5.2 Technical Implementation

The application used to code was NetBeans IDE 14, this enabled us to use a GUI with JFrame to make a nice and easy visual interface for user's to use.
The programming language we chose to go with was java as this is what the group was familiar with and it is most of our strong points.
We used two libraries which were BCrypt and JDK connector/j. BCrypt was used to encrypt the hashed passwords for authentication and the JDK connector/j is used to connect the Java file to the database.

### 5.3 Design of the Application



# **Chosen Use Case Scenario:** User (Brian) Register

**Business Functionality**

The purpose of the secure sign-up application is to provide users with a secure and user-friendly environment for creating accounts with confidential data.
The application ensures the confidentiality of user passwords through the use of password hashing. Additionally, the application provides feedback messages to users about the success or failure of the registration process.

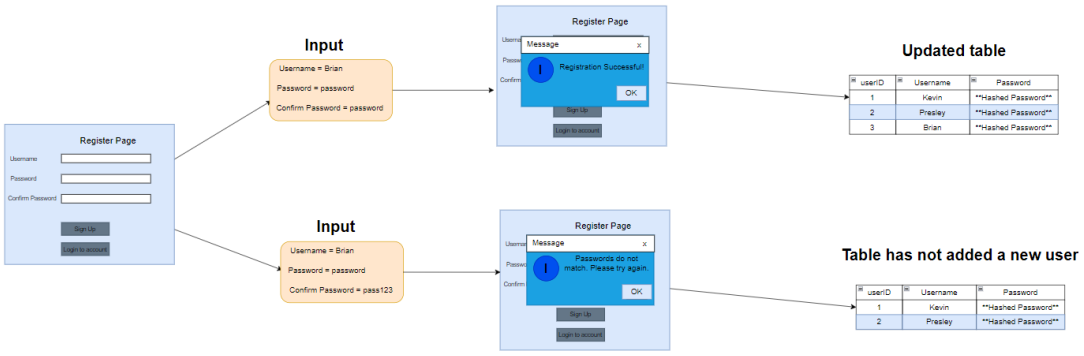
**Value Offered by the Application:**

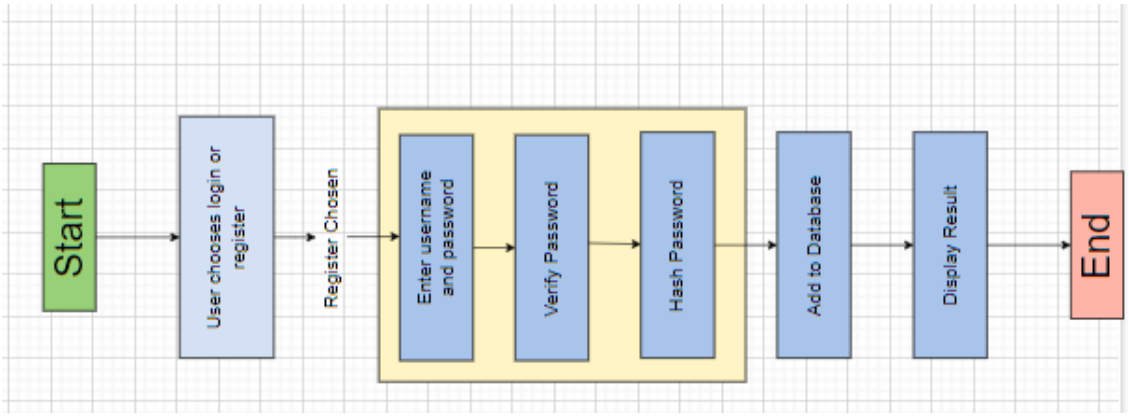
1. **Security:**
   - Passwords are hashed using a secure hashing algorithm (BCrypt) to protect user credentials from unauthorized access.
2. **User-Friendly Interface:**
   - The graphical user interface (GUI) facilitates a seamless user experience, allowing users to easily input their credentials and receive clear feedback.
3. **Confidentiality:**
   - User passwords are handled with utmost confidentiality, with only the hashed versions stored in the database.
4. **Verification:**
   - During the sign-up process, users are prompted to enter their password twice for verification, ensuring accuracy and minimizing input errors.
5. **Informative Feedback:**
   - The application provides users with informative messages, such as confirming successful account creation or alerting to existing account information.

**Technical Implementation**

As above, the application used to code was NetBeans IDE 14 and we used JFrame to make the GUI.
The programming language we chose to go with was also Java.
We used the same two libraries as mentioned in the login use case, which were BCrypt and JDK connector/j.

**Design of the Application**




Start
User chooses login or register
Register Chosen
Enter username and password
Verify Password
Hash Password
Add to Database
Display Result
End