Date: March 21, 2022

# **Introduction**

This project is part of PROG24178 Object Oriented Programming Java-2 class activities at Sheridan College. It aims to enhance our knowledge and experience of how to develop an application’s interface/screen using JavaFX and OOP.

# **Project members**

The group members of this project consist of the following students:

* Charanpreet Singh,
* Gurjot Singh,
* Khalid Joraid,
* Robin Bedminister,

# **Challenge**

Nowadays, we have many passwords and need a safe place to keep them. Some services provide a place to store passwords, such as a google account. However, anyone can access this data and view your passwords and where they are used. Also, suppose you keep the passwords in a file with password protection; you are at risk of forgetting the file password, failing to open the file, or if the file does not have a password, someone may open the unprotected file, and your passwords could be accessed.

# **Solution**

To solve the above issue, we could think of an application that keeps our passwords in one place and protect them from being read by encrypting them. Even if this file containing the encrypted passwords gets unwanted access, they remain protected since they are encrypted, and no one can understand what they are.

# **Project High Level Idea**

This project aims to build a Password Encryption application that allows users to enter a list of passwords with related information. Then, the system could encrypt these passwords and save them in a file. Users could retrieve this file at any time, decrypt the passwords for review, change, or add to them and then save it again. After saving the file, the password in the real format will not show up on either the screen or the file unless this application is used to open the file and decrypt the passwords again.

# **Project UML Design**

Diagram

Description automatically generated

# **Project GitHub Link**

You can link your local GitHub to the source code directly via this link

<https://github.com/kjoraid/passwordEncryption.git>

# **Project Interface**

Figure 1.1 A suggested form design.

Graphical user interface, website

Description automatically generated

1. JFXForm a form that shows the text fields and buttons as illustrated in the figure 1.1.
   1. Click (+) Insert line data to the grid
   2. Click (del) Delete line data from the grid
   3. Click (Save) Call Save method in the OpenSave class and passing Date, Filename, and Data Array
   4. Click (Open) Call Open method in the OpenSave class and receive Date, Filename, and Data Array
   5. Click (Encrypt/Decrypt): Call Encrypt method in the EncryptDecrypt class and passing the Data Array. The method will return an Array with encrypted data.

# **Project High Level Design**

Classes and Methods

1. PasswordEncryption.java

This class contains Main and Start methods to be used

    to design JFX screen.

    Work on start method and use the appropriate FX objects to design

    a screen that accepts data from user according to the design and

    follow these actions:

    A) Save the data into array:

        1- Each time user insert/delete new line,

            1.1 Call a method to add/delete this line into/from

                the array.

            1.2 Call another method to display the array

                on screen in a table view.

    B) Each button should be processed as following:

        1- When User Click on Open File button:

            1.1 Call a method to open the file

            1.2 This method will return an array with data from the file

            1.3 Check the data and then call the method to display the Array content

        2- When user click on Clear Screen:

            2.1 Confirm from user

            2.2 Clear the array

            2.3 Call the method to clear the screen

        3- When User Click on Encrypt/Decrypt button

            3.1 Call the Encrypt method and pass the Array of data to it

            3.2 The method will return an array with Encrypt/Decrypt data

            4.2 Call the method to display this table on screen

        4- When User Click on Save button

            4.1 Call the Save method and pass the array to it

            4.1 Confirm to the user that the system saves the file

            4.3 Show the file name on the text field

1. OpenSave.java

This class is responsible of:

    A) Open a file and retrieve the data to the caller by following

    these steps:

        1- Call the open class to Open a file in any folder

        2- Check if the file is correct

        3- Fill the data of the file into Array

        4- Return this array to the caller using return command

    B) Users will send an array to a method in this class to

        save the data on file by following these steps:

        1- User will pass an array with the data required to be saved

        2- Check the array if it is correct and contains data

        3- Call the save class and pass the array to it

1. EncryptDecrypt

This class is responsible of:

    A) Encrypt data passed by Array by following these steps:

        1- Check if the array is having data

        2- Call the Encrypt class to encrypt the data

        3- Return the array with encrypted details

    B) Decrypt data passed by an Array by following these steps:

        1- Check if the array is having data

        2- Call the Decrypt class to encrypt the data

        3- Return the array with Decrypted details