Brandon Mohan and Jared Dunn
CMSC 413 Password Manager
12/4/2021

Design

For the overall website, we went with a black/gray/white theme. Red is used to differentiate buttons.

We tried to design the site to be simple to use and easy to navigate. Any somewhat tech savvy user should be able to use our site with ease.

Our site is also lightweight. It is perfect for people who want to store their credentials on a slower machine or in a VM.

User Manual

To run our project, you must first import the required libraries. We used Flask, cryptocode, and flask-login. These can be installed via pip.

Once you have the required libraries, you can run our project via "python app.py". From there it will take you to the login screen.

Login/Sign in to your password manager

| Username | Password | Login | Register |

- On the login screen, you can login with your credentials, or create a new account.
- Note that all passwords will be checked to make sure they meet the following criteria:
    - Must be at least 8 characters long
    - Must contain at least 1 number and 1 letter
    - Must not have more than 3 consecutive characters ex. aaaaaa or 1111111
    - Must not have more than 3 sequential letters ex. abcde or edcba
    - Must not have more than 3 sequential numbers ex. 12345 or 54321

Once you create an account or login, you will be taken to the password management screen.

This is what the password management screen looks like for new users.
- On the password management screen, you can add and delete passwords.
- Note that you must provide a website/username/password for an entry to be added.

Adding login credentials will populate the screen, like so:

Website	Username	Password	
Walmart	WalmartUser	WalmartPass1	DeleteEntry#0
Target	TargentAccount	supersecretpas2232	DeleteEntry#1

- From this screen, you can delete passwords you have added. Simply delete the corresponding delete button on the entry you want to remove.

If you would like to delete the master account, simply delete passwords.txt from the directory. The project will run from a fresh start.

It should be noted that we use SHA-256 to generate hashes. We encrypt passwords via AES-GCM encryption. All hashes/passwords are stored in passwords.txt.
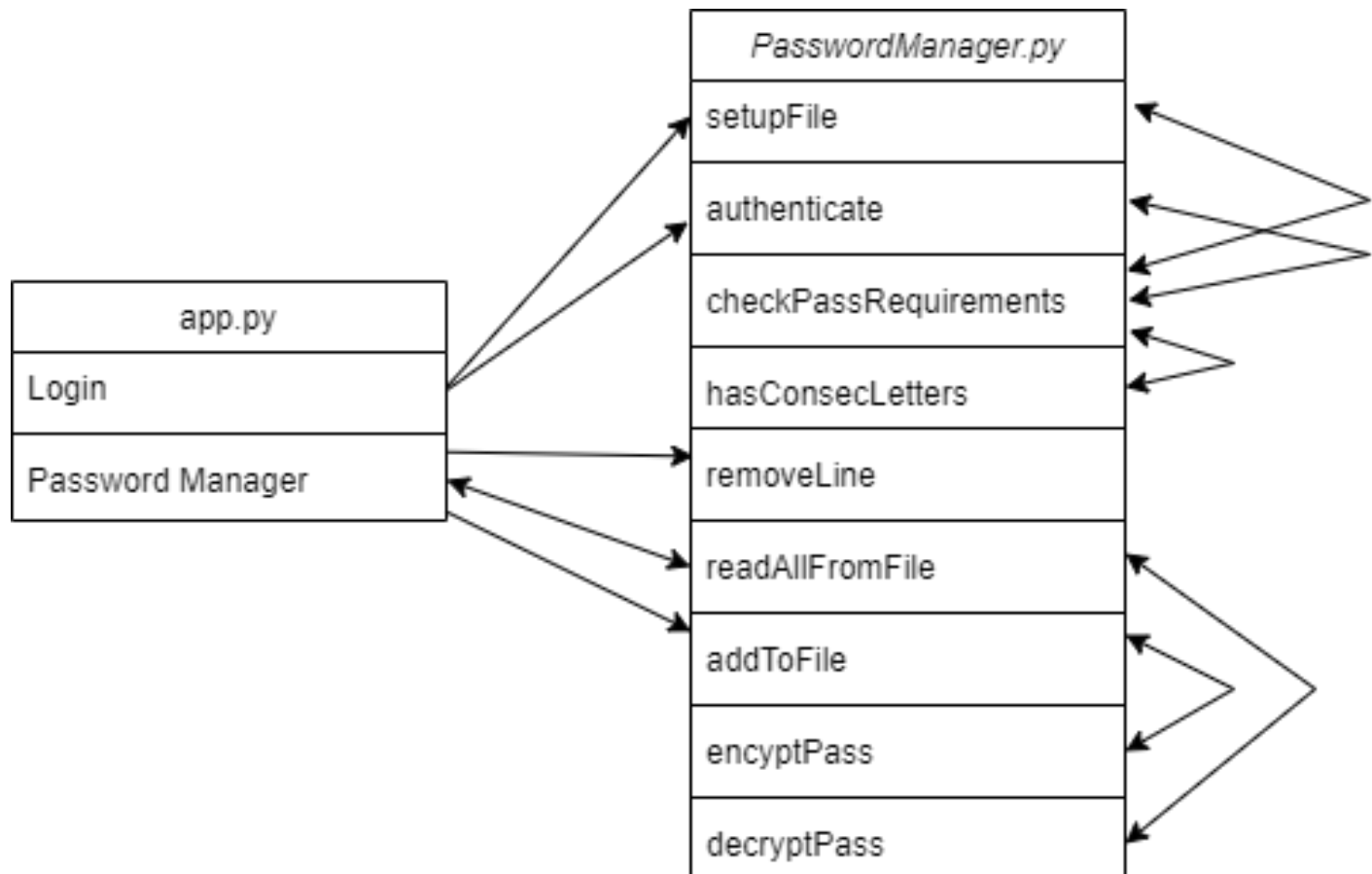
Code

Our github can be found here:
https://github.com/BrandonM001/CMSC413CourseProject
Brandon also included a zip file with his submission.

Function Relationships

References

- https://realpython.com/introduction-to-flask-part-2-creating-a-login-page/
- https://pypi.org/project/scrypt/
- https://pypi.org/project/cryptocode/
- https://github.com/holgern/py-scrypt
- https://github.com/gdavid7/cryptocode
- https://www.geeksforgeeks.org/file-handling-python/
- https://pythonhow.com/python-tutorial/flask/Adding-CSS-styling-to-your-website/
- https://app.diagrams.net/