PROJECT PROPOSAL
Password Safe
Jayden Lupold, Logan Krause, Dary Demchuk, & Lian Welch

Table of Contents

Project Proposal

Project Abstract

"Personal Safe" password manager will be developed as a lightweight application to securely store and manage passwords. It will allow users to create a master account, then add, retrieve, and organize all their login credentials, which will all be encrypted and decrypted locally on their own personal device. There are plenty of password managers like LastPass and 1Password, but unlike "Personal Safe", many of them rely on cloud storage which introduces risks if their servers are compromised. This project is different because all encryption happens on the client side, meaning data never leaves the user's device in a plain form. The intended users are everyday people who want better security or people who are just serious about their privacy. Ultimately, this project will provide a security tool for users wanting to protect their passwords.

High Level Requirement

From a user's point of view, "Personal Safe" will function as a secure and simple password management tool to safely store and manage all of their login credentials in one place. The application will require the user to create a master account, granting access to all stored passwords. Once logged in, the user can:

- o Add new credentials by entering a website name, username, and password.
- o View their saved credentials.
- o Copy stored credentials to their clipboard.
- o Delete or edit saved passwords if needed.
- o Generate strong passwords using an integrated password generator and check password strength using a built-in strength meter.
- o Store all data locally in an encrypted JSON file.
- o Export and import encrypted backups, maintaining privacy even during data transfer.

All encryption and decryption processes will occur locally on the client side, ensuring full control over their personal data. The interface of the password manager will be simple and intuitive, created using Python's tkinter GUI framework.

Conceptual Design

The initial design concept for "Personal Safe" is a client-side password manager created with Python 3 using tkinter for GUI and JSON/AES encrypted local storage. It would run on multiple platforms including windows, macOS, and Linux. The initial design would ensure that any credentials would remain fully under the user's control and balances security with ease of use. Some of the features would include a master account, adding/deleting credentials, a password generator, a strength meter, secure local storage, the ability to copy to clipboard, and optional encrypted import/export for backups.

Use Case Diagram

Proof of Concept

https://github.com/cis3296f25/final-project-04-personal-safe.git

Background

"Personal Safe" will enable users to generate, store, and manage passwords securely, with any encryption and decryption performed locally on their device. This approach ensures that any sensitive data never leaves the device in plain text, while reducing risks from breaches of centralized servers. The main features will include a master account login, password storage and retrieval, a password generator, a strength meter, and an optional encrypted export for any backups.

While proprietary products like LastPass and 1Password are useful, they heavily rely on cloud storage, which can make them targets. Other open-source options like Bitwarden and KeePass provide flexibility, but they come with harder setups and even partially rely on external infrastructure. This project won't reuse any of their source code but it will likely need to borrow some design inspiration. Its main difference between those other products is the focus on full client-side encryption and user-controlled storage, which combines usability with even more protection.

Required Resources

This password manager will be developed using Python 3, which I'm familiar with. I'll also use the tkinter library to create a simple GUI for entering and viewing passwords, and the json and base64 libraries to store encrypted passwords locally. For encryption I'll implement a XOR-based method for this prototype, but I'll most likely be added stronger encryption later using the cryptography library if I need to. The project can be run on any computer without the need for special hardware, and any necessary Python libraries are either built-in or installed relatively easily.

Project Design

Vision

"Personal Safe" is a password manager that is a lightweight application to securely store and manage passwords. This tool is for any users wanting to protect their passwords and who want better security. There are plenty of password managers like LastPass and 1Password, but unlike "Personal Safe", many of them rely on cloud storage which introduces risks if their servers are compromised. This project is different because all encryption happens on the client side, meaning data never leaves the user's device in a plain form.

Personas

1. John Doe

    a. Name: John Doe

    b. Age: 16

    c. Occupation: High school student

    d. Educational Background: In high school

    e. Technology Experience: Uses social media, iMessage, and video games.

f. Demographics: Teenager who is becoming more responsible for their own accounts, loves to play videos games and lives in the suburbs. g. Why they'd use Personal Safe: This person is creating accounts for the first time and wants to ensure that all his new passwords are secure.

2. Jane Smith

    a. Name: Jane Smith

    b. Age: 68

    c. Occupation: Retired

    d. Educational Background: BS in Education from Temple University

    e. Technology Experience: Experienced in social media and basic software required for teaching but has not used technology regularly in five years.

    f. Demographics: Lives in a nursing home, many children and grandchildren who visit often, in a small city that supports their community.

    g. Why they'd use Personal Safe: Smith is getting nervous the more she becomes out of touch with developing technology and wants to ensure that all her accounts are safe, especially since she has been checking them less since retiring.

3. Mary Donohue

    a. Name: Mary Donohue

    b. Age: 40

    c. Occupation: Secretary for a doctor's office.

    d. Educational Background: Associate's degree in business from SNHU

    e. Technology Experience: Proficient in all aspects of Microsoft office and Epic software.

    f. Demographics: Married with no kids in a busy city with a high crime rate. Is active in their community garden.

    g. Why they'd use Personal Safe: She is concerned that the system her hospital uses is getting outdated and fears HIPPA violations as technology advances. She would use Personal Safe as a safety net for the sensitive information that she has access to at her front desk.

4. Frank Don

    a. Name: Frank Don

    b. Age: 25

    c. Occupation: Lawyer

    d. Educational Background: Law degree from Harvard

    e. Technology Experience: Extensive database organization and understanding, as well as uses personal laptops for all meetings and accessing files.

    f. Demographics: Lives in the heart of Washington DC, single and travels on the weekends.

g. Why they'd use Personal Safe: They keep sensitive client data on their personal computer that they take with them when they travel, using Password Safe would ensure that this confidential data is more secure.

5. Julie Steve

a. Name: Julie Steve

b. Age: 55

c. Occupation: Personal Tutor

d. Educational Background: BS in Chemistry from UCLA

e. Technology Experience: Extensive data organization experience, proficiency in zoom and Microsoft.

f. Demographics: Lives in a rich southern rural neighborhood. Married to the mayor with many people in her town asking for favors.

g. Why they'd use Personal Safe: Many of her clients are embarrassed to need tutoring and don't want to be the topic of conversation throughout the town, so protecting her passwords ensures that none of her nosey neighbors can access confidential information.
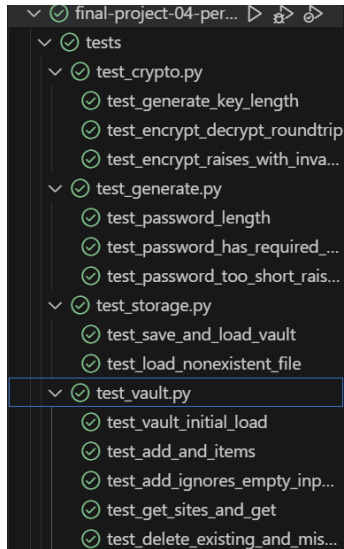
Class Diagram

Class Diagram
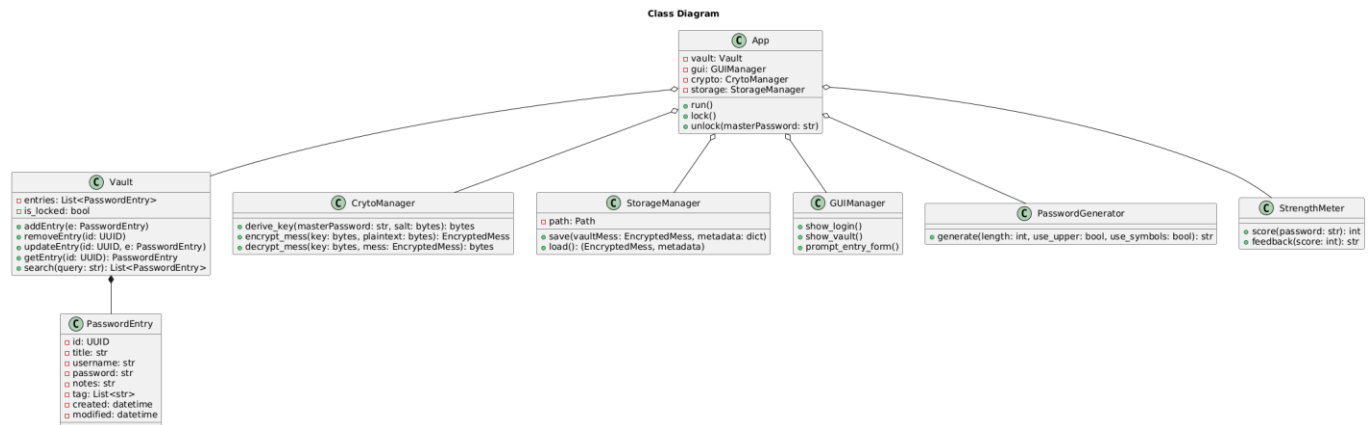
Sequence Diagram

Sequence Diagram
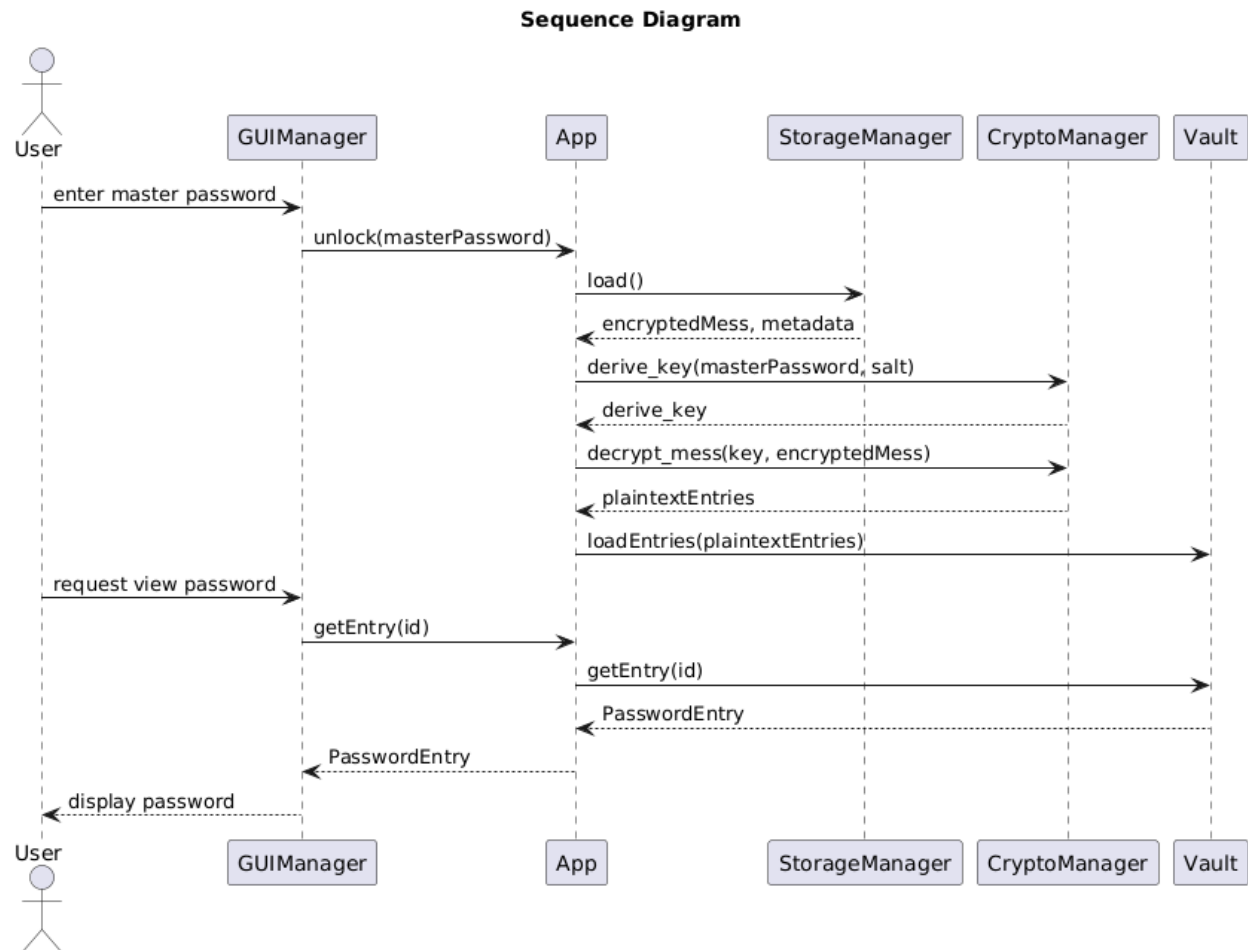
Automated Test Results

Output from the unit test runs:

Output of the coverage report:

```
22    ============================== tests coverage ===============================
23    _____ coverage: platform win32, python 3.13.9-final-0 _____
24
25    Name                        Stmts   Miss  Cover   Missing
26    ------------------------------------------------------------
27    crypto.py                     36      0   100%
28    generate.py                   14      0   100%
29    main.py                       15     15     0%   1-21
30    masterPassword.py             55     55     0%   1-72
31    storage.py                    32      1    97%   49
32    tests\test_crypto.py          25      0   100%
33    tests\test_generate.py        18      0   100%
34    tests\test_storage.py         15      0   100%
35    tests\test_vault.py           29      0   100%
36    ui.py                        125    125     0%   1-161
37    vault.py                      25      0   100%
38    ------------------------------------------------------------
39    TOTAL                        389    196    50%
40    ============================ 13 passed in 0.89s ================================
```

Project Progress

Week 2 Progress
Scrum Master: Dary Demchuk

Sprint Goal: This week's sprint goal is to modify and adjust the working model.

Backlog Features

1. Create requirements.txt file
   - Estimated velocity – 1, 30 minutes XS
   - Effective velocity– 1, 30 minutes XS
2. Style the front end
   - Estimated velocity– 5, days M
   - Effective velocity– 3, one day M
3. Edit Password
   - Estimated velocity– 5, days M
   - Effective velocity– 5, days M
4. Create two sequence diagrams
   - Estimated velocity– 1, 30 minutes XS
   - Effective velocity– 1, 30 minutes XS

Week 3 Progress

Sprint Goal:

Backlog Features

Tasks in Sprint	Size	Task Status at end of Sprint	Assigned To

Estimated Velocity (At beginning of Sprint)	0
Calculated Velocity (At end of Sprint)	0

Week 4 Progress

Sprint Goal:

Backlog Features

Tasks in Sprint	Size	Task Status at end of Sprint	Assigned To

Estimated Velocity (At beginning of Sprint)	0
Calculated Velocity (At end of Sprint)	0