Computer Science Department

Capstone Project Proposal

**webb**

Mitch Rodriguez

Course Number and Course Title: CSC 520

Project Advisor

Dr. Bo Hatfield

Project Supervisor

Professor Kentros

webb

webb is a data management web application that has a focus on security. The app allows the customer to be able to import and export private files and information and be able to set encrypted passwords on any imported files with an algorithm that’s not only secure but very hard to break. New accounts created in the application will have strict password requirements, security questions for verification, and a two-factor authentication system for whenever an account is created, or when a user attempts to login at a new IP. The app will also feature something called History, which allows the logged in user to view what changes have been made to and within the account, including login attempts if someone tries to break into the account. webb gives the customer the peace of mind knowing that they are the only ones viewing the information imported within the app. Managing data with security and confidence.

Problem Specification: Whenever a user creates an account for most online platforms, password requirements aren’t very strict, and security preferences for an account aren’t enforced by the platform. Most people aren’t actively looking to make their account more secure, and it’s not because they don’t know how to or don’t care, its partially because they don’t understand what can happen once a hacker breaks into a one of there accounts. As someone who has been working in a customer facing environment for over five years, and a large portion of that time spent is troubleshooting, and assisting customers in resolving issues they have, there’s something that’s abundantly clear that is a similarity shared with hundreds of customers I’ve assisted. There passwords are either too simple and can be easily brute forced with very little computation resources to break their password, or they don’t know their passwords to their respective online platform. In some extreme cases, I have spoken to and helped customers who have had several accounts breached and believe there experiencing someone attempting to steal their identity. From seeing first-hand how bad a lousy password can cause a slew of problems when the user has his or her account breached, I not only have setup multiple thirty-character passwords for my own personal accounts, but I’ve applied these similar security mentalities with webb. A user can easily create an account with webb, however they will need secure account information to do. Password requirements for webb are set the way they are so that a user cannot create a short password, and fourteen seemed like a good number to help prevent. In just one day, an eight-character password can be cracked to break into a user’s account, whereas fourteen characters will be significantly harder to break and take more time. Although fourteen characters doesn’t completely secure a user’s information, the cap limit for a password to be created with webb is one hundred. A user will have plenty of flexibility to create a longer password if they feel like they need it. To make a user’s account as secure as possible, not only will the password be encrypted using a very complicated algorithm, but account creation also asks for the setup of two factor authentication and security questions. So even if a user has a simple fourteen-character password that has been cracked, two factor authentications will stop the hacker from continuing to break into the account. The user can make their account as secure as they want, while keeping even the minimum requirements more secure than most online platforms minimum account requirements. When a user creates and logs into webb, the application opens to its main page which allows the user to import, export, create a folder, set password encryption, check account preferences, and check History. webb allows a user to take a private document or set of documents, import them into webb and have peace of mind knowing their eyes are the only ones viewing that document.

webb is a data management web application that allows users to import, export, and secure documents, files, and photos with having an emphasis on security for both logging into the account and securing files by encrypting them using the Argon2 algorithm. The languages used to help create webb will be written in Python with the Django Framework, HTML and CSS. The front end and back end of webb will be managed and developed with the help of the framework, making development faster and more secure. All passwords, including account and files encrypted, will be secured using the Argon2 algorithm. Both accounts and individual files will be extremely secure, and a database will store users’ information and saved IP addresses for the two-factor authentication system.

A user can get started with webb by first signing up within the web application. Said user will enter their first and last name, their email, and a secure password. The password requirements are as follows: minimum of fourteen characters, maximum characters being one hundred, one capital letter required, one number, and one special character. Once this has been completed, the user will then be prompted to enter the code sent to their email address, which will the initiate the two-factor authentication system. Afterwards, the user will finally be prompted to setup their security questions, which will be for authentication if the user does not remember their passwords. These case sensitive responses will be for no less than three, to no greater than eight security questions. The purpose of having these higher end, high amount choices like for passwords and security questions is so that the user could make their account as secure as they possibly can. Once the user has properly created their account, the user will be in webb, and have access to all features such as import, export, and encrypt.

There are currently no prototypes for the project yet, but the list of deliverables, case diagram and use case checklist will be below.

Student Objectives:

* Learn different encryption methods and find which will be the most secure. As of right now that is Argon2.
* Become more familiar with web development and successfully implement the Django framework in making my capstone.
* Gain more experiences and work with databases and how it will be implemented into my project.
* Be able to use those databases and store customers information and stored IP address within them.
* Have a few accounts set up for webb, with different information and files stored within them.
* Properly implement a two-factor authentication system, with a code being sent to the customers email address, and the customer has sixty seconds to retrieve that code and enter it.
* Become more familiar with how AWS will be implemented into my project.
* Ability to build the application and focus on security, creating the strict requirements for account creation, and learn how Argon2 will encrypt the account password and any imported file password that is set.
* Build web. This is the biggest project I’ve worked on and I’ve never worked on something like this before. Become knowledgeable in Python/HTLM/CSS/Any other programming language I need to learn to create and finish my project.

List of deliverables:

* Two factor authentication system for new accounts and new log ins on different IP addresses.
* Strict requirements for account creation regarding password requirements and security question setup.
* User can import a file or photo within the web application and move the file to and from folders within the app.
* User can export a file or photo, assuming there is not password encryption set on the file.
* User can setup password encryption for files within the application that are already imported.
* Passwords for the account as well as the files with the encryption set on them will be encrypted using the Argon2 algorithm.
* User should be able to log in to the app, open their account settings, and be able to change password, security questions, and amount of security questions, as well as use a different email address for the account, which will then prompt to setup the two-factor authentication system again.
* User will have the ability to access History, which will show any changes that have been made to the account while logged in, as well as if any log in attempt have been made for the account.

Use case checklist:

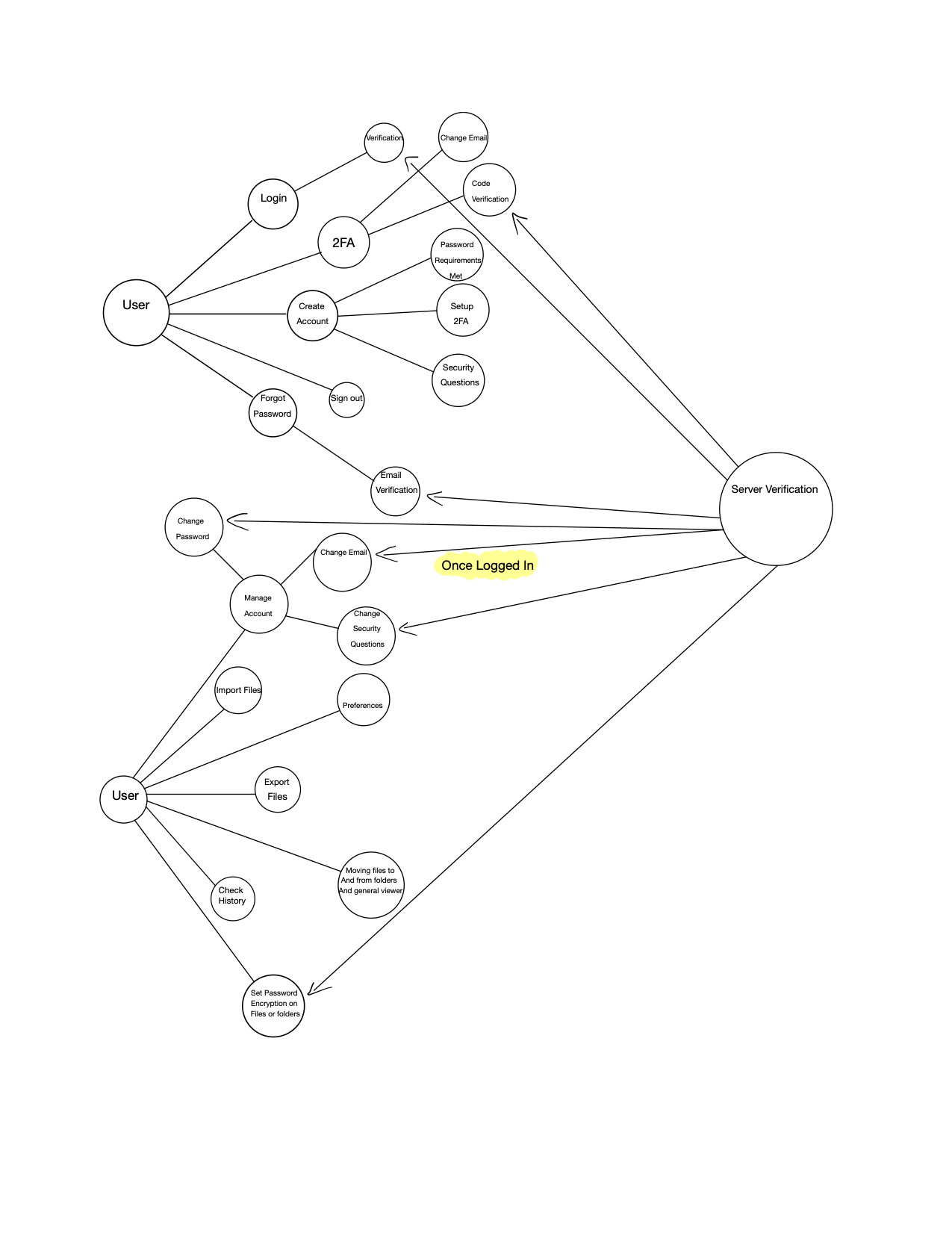
* Login splash screen will consist of two fields, email, and password, with create account and forgot password link on page.
  + Log in successful and user is entered into account.
  + Login is not successful, prompting “incorrect email or password, please try again”.
    - In the rare case that an account has four unsuccessful logins with a registered email, the account will be locked temporarily due to multiple failed attempts.
      * Account lock timers are as follows: 15 minutes for first failed attempt, 30 minutes for second failed attempts, 60 minutes for third failed attempt, and for the final attempt/any attempts after third failed attempt until log in is successful, 180 minutes. After the first failed log in timer, user can attempt password entry twice until next timer and lock engages.
  + Create account
    - Enter email/password (minimum of twelve characters with one number, one capital letter, and one special character required.
      * Entered account credentials do not meet requirements, prompt user to try again.
      * Entered account credentials meet requirements.
        + Prompt user for two factor code sent to entered email.

User enters code correctly and continues.

System prompts user to create three to eight security questions for verification.

* + - * + User enters code incorrectly, and either tries again or requests a new code.
        + User doesn’t enter code in time, new code is resent, and user must enter within sixty seconds.
  + Forgot password.
    - Prompt user to enter email address for account.
      * User enters email properly and is sent a six-digit verification code to enter within the application to allow for new password creation.
        + Code entered correctly, prompts user for new password to be entered twice with same requirements as log in.
        + Code entered incorrectly, prompt user to try again, or resend a code to email with maximum attempts being three.
      * Email entered by user that does not have an account associated with it will receive a prompt stating the entered email is not linked with an account, and prompt to try again.
* Once logged in.
  + Import files.
    - Successfully imported file.
  + Export files.
    - Successfully exported file.
    - Attempt to export file that is encrypted with password, “Please remove password encryption before exporting.”, prompt user to remove password encryption by asking for password to be entered.
  + Check history
    - Shows user what changes have been made to the account including imports/exports/etc as well as if the account had any attempted log ins.
  + Set password encryption on either files or folders.
    - User sets password encryption on individual files and folders containing multiple files.
  + Moving files to and from folders and general viewer
    - Successfully prompts for moving files.
  + Top left of system is systems menu, showing “Import”, “Export”, “History”, and “Preferences”.
  + If files that are encrypted need to be moves to and from folders or out of folders, the password encryption set on the file or folder needs to be entered before continuing.
  + Within preferences will be account settings, like “Change password”, “Edit security questions”, “Change email”, and possibly more.
    - Changing email; for this to be successful to be successful, two factor authentications will be setup again with new email.
    - Changing passwords will require the user to answer the security questions again, ensuring the user requesting password reset is the owner of the account.
    - Editing security questions will require the user to re-enter their password.
* Exit app.

Use case diagram:



References: