Planning Document: Sprint 3
Team 19 – Sentinel Data Vault
Jiho Choi, Zhaoji Jiang, Adam Petty, Dingfu Sun, Thomas Worns
November 16, 2015

SPRINT OVERVIEW

In this sprint, we will focus on implementing all the remaining crucial functionality of the data vault, as well as miscellaneous features to improve user experience. Our main goals for this sprint are:

1. Integrate Crypto to encrypt and decrypt data entries.
2. Implement sharing functionality.
3. Implement system backup functionality.
4. Implement various UI features.
5. Improve miscellaneous user experience features.

Scrum Master: Adam Petty

Scrum meeting time: MWF 12:30pm

Risks/Challenges: The last remaining features are ones in which we felt would be the most difficult to implement and require additional outside learning. We will have to learn how to make backups of an SQLite database, manage system time tracking, and figure out sharing data between separate user accounts both internally and over email.

CURRENT SPRINT DETAIL

User Story (Crypto):
- As a user, I want my data entries to be encrypted.
- As a developer, I want the ability to choose between two levels of data entry encryption (normal and strong).
- As a developer, I want the ability to change encryption in the future and upscale security.

Task Description	Owner	Time Estimate
Integrate encryption functionality into the creation of a DataEntry.	Zhaoji Jiang	1
Integrate decryption functionality into the retrieval of a DataEntry from the database.	Zhaoji Jiang	1
Implement two levels of encryption in the Crypto class.	Thomas Worns	5

Acceptance Criteria:
- When data entries are stored in the database, their fields must be in an encrypted, non-readable format.
- When data entries are retrieved from the database for display in the data entry browser, they must have human-readable data fields.
- When a data entry is encrypted under strong security settings, the encryption will be different from the encryption under normal security settings.
- When two data entries are created each with the same fields but differing levels of encryption, their information will be encrypted significantly different when stored in the database.

Test Case: Encrypt newly created/modified data entries and store them in the database
Instructions:
1. Create a new data entry for that user.
2. Inspect database.
3. Open the newly-created data entry.
4. Change user high security settings
5. Create another new data entry.
6. Inspect database
7. Open the second newly created data entry

Expected Results
1. The data entry should be successfully created.
2. The data entry in the database should have all data fields encrypted.
3. Data entries being displayed should be in human readable format.
4. The high security settings should change.
5. The new data entry should be successfully created
6. The data entry in the database should have all fields encrypted.

7. The data entry being displayed should be in human readable format.

Test Case: Different encryption levels for normal vs. strongly encrypted data entries
Instructions:
1. Create a new data entry for a user with a specific value in a field.
2. Change security setting.
3. Create a second data entry for the same user with the same specific value in a field.
4. Inspect database.

Expected Results
1. The data entry should be successfully created.
2. The security setting should change.
3. The data entry should be successfully created.
4. The two data entries with the same field should have differently encrypted strings stored in the database.

User Stories (Help & FAQ):
● As a user, I want a help section in the application to provide more detailed FAQs and explanations of functionality.

Task Description	Owner	Time Estimate
Write text tutorials to be placed in the HelpView for each FAQ / Help Topic and redesign existing HelpView as necessary.	Adam Petty	5

Acceptance Criteria:
● When the user clicks "Help" from the Home window, a Help Window will open containing a list of various FAQs / help topics.
● Given the user has clicked on an item in the list of "Help" items in the Help Window, the relevant help information text will be displayed to the right within the window.
● When the user clicks "Close", the help window will close and the user will be returned to the Home window.

Test Case: Viewing help topics and/or FAQs in the Help Window
Instructions:
1. At the Home Window, click the "Help" button.
2. At the Help window, click a help item from the list on the left of the window.
3. Click "Close" to close the Help window.

Expected Results:
1. The Help window will open, showing a list of help topics.
2. Help will be displayed in the right side of the Help window with information related to that topic.
3. The Help window will close and the user returned to the Home window.

User Story (Remaining Data Entries):
● As a user, I want to store a wide variety of data items, including but not limited to: PINs, Shipment tracking numbers, Plane tickets, ID cards, Entry codes, Account numbers, Coupons, Application passwords, Serial numbers, Receipts/Confirmation numbers, prescriptions (only unimplemented types listed).

Task Description	Owner	Time Estimate
Implement remaining Data Entry type UI classes for NewDataEntryView, and their related checks/functionality.	Zhaoji Jiang	6
Implement remaining Data Entry type UI classes for EditDataEntryView, and their related checks/functionality.	Zhaoji Jiang	6
Implement remaining Data Entry type UI classes for DataEntryPanel	Jiho Choi	5

Acceptance Criteria:
● When a user clicks on "New Entry" from the Home window of the data vault, then a list will open with all the possible data entry types that a user can create (including the ones listed above).
● When a user clicks on "New Entry" from the Home window of the data vault and chooses an entry type, then each data type will have its own unique UI for creating that entry.
● When a user clicks "Edit Entry" from the Home window of the data vault, then each data type will have its own unique UI for editing its fields.
● When a user choose a data entry from the Home window of the data vault, then each data type will have its own unique UI for displaying its fields.

Test Case: All available data types show in "New Entry" list and can be successfully created
Instructions:
1. From the Home window of the data vault, click "New Entry".
2. Click the drop-down list of data entry types.
3. Click on a data entry type.
4. In the Create New Data Entry window, enter information into all the fields.
5. Click "Create"
6. Repeat as needed for each data type.
Expected Results:
1. A drop-down list of data entry types will open to select from.
2. All available data entry types that a use can create appear in the data entry list view.
3. The Create New Data Entry window for that specific data entry type opens, showing all required information text fields to be entered.
4. Text fields will show entered information.

5. The Create New Data Entry window closes, the newly-created data entry appears in the list of all user data entries, and the newly-created data entry's information will display in the entry viewer.

Test Case: All data types can be successfully edited
Instructions:
1. From the Home window of the data vault, click "Edit Entry".
2. In the Edit Data Entry window, modify information in one or more of the required fields (without deleting a field and leaving it blank).
3. Click "Save".
4. Repeat as needed for each data type.

Expected Results:
1. The Edit Data Entry window opens, showing all the existing data entry information in their respective text entry fields (able to be edited by the user).
2. Text fields will show existing and altered information.
3. The Edit Data Entry window closes, the user will be returned to the Home window of the data vault, and the modified entry's information will display in the entry viewer when clicked on.

Test Case: All data types can be successfully display
Instructions:
1. From the Home window, choose a stored data
2. Repeat as needed for each data type.

Expected Results:
1. Every data type have its own unique DataEntryPanel for displaying its fields in the home view

User Story (Sort Entry & Search Entry):
● As a user, I want to be able to sort my data entries by name or last modified date (in addition to the standard sorting by type).
● As a user, I want searching functionality to easily find data entries.

Task Description	Owner	Time Estimate
Create a "Search" field and button in HomeView.	Jiho Choi	1
Implement functionality to search for a data entry by name.	Jiho Choi	5
Create a "Sort By" button in HomeView.	Jiho Choi	1
Implement functionality to sort a user's data entries by name and last modified time.	Jiho Choi	5

Acceptance Criteria:
● When the user types a search term into the search field and initiates the search by clicking the "Search" button, entry names that match that term are displayed to the user while those that don't will be hidden.

● When the user deletes the search term from the search field after performing a search, all entries will again be visible to the user.
● When the "Sort By" button is clicked, the data entry browser will display the entries sorted by type (default), name, and last modified time based on what method is selected.

Test Case: Searching for an existing entry by its name
Instructions:
1. On the Home window above the data entry browser, enter a search term for the name of an existing entry into the search field.
2. Click "Search"
3. Delete the search term from the search field.
Expected Results:
1. The search term text will be shown in the search field, but the shown data entries will not change.
2. The data entry browser will only show entries with names that match the search term.
3. All data entries will again be visible to the user.

Test Case: Searching for a non-existent entry
Instructions:
1. On the Home window above the data entry browser, enter a search term for the name of an non-existent entry into the search field.
2. Click "Search"
3. Delete the search term from the search field.
Expected Results:
1. The search term text will be shown in the search field, but the shown data entries will not change.
2. The data entry browser will not show any entries and display "No entries found".
3. All data entries will again be visible to the user.

Test Case: Sorting entries in the data entry browser by name and last modified time
Instructions:
1. On the Home window above the data entry browser, click the "Sort By" button and choose "Entry Name"
2. On the Home window above the data entry browser, click the "Sort By" button and choose "Last Modified Time"
Expected Results:
1. The data entry browser will show show a list of all the user's data entries sorted in alphabetical order by data entries' name.
2. The data entry browser will show show a list of all the user's data entries sorted in chronological order by data entries' last modified time.

User Stories (Forget Password / Password Retrieval):
- As a user I want to be able to regain access to my account if I forget my password.

Task Description	Owner	Time Estimate
Implement PasswordRecoveryView wherein a user enters their username, answers their security question, and enters a security code	Thomas Worns	6
Implement functionality to send an email containing a randomly generated security code after the user enters their username and security question answer into PasswordRecoveryView	Thomas Worns	3
Implement functionality in PasswordRecoveryView to allow users to change their existing password after entering their username, security question answer, and security code	Thomas Worns	6

Acceptance Criteria:
- When a user clicks the "Forgot Password...:" button, the PasswordRecoveryView will pop up.
- When a user enters their username, their security question will pop up.
- When the user answers their security question correctly, an email containing a randomly generated security code will be sent and a window prompting for that security code will be displayed.
- When the user enters the code, they will be taken to a window where they can change their password.

Test Case: I forgot my password
Instructions:
1. Click "Forgot Password" button
2. Enter desired username
3. Answer security question correctly
4. Retrieve and enter security code
5. Change password
6. Log in with new password

Expected Results:
1. When the "Forgot Password" button is clicked, the PasswordRecoveryView will pop up.
2. When the user enters their username, a place where they can answer their security question should display.
3. When the question is answered correctly, an email will be sent to the user.
4. When the security code is entered correctly, a window for changing passwords will be displayed.
5. When a new password has been set, it will be the user's new login password
6. When the new password is used to log in, the log in should be successful.

User Stories (Hide/Show Passwords):
● As a user, I want to toggle between hiding and showing passwords within text fields.

Task Description	Owner	Time Estimate
Implement a toggle next to any password text fields in the data entry browser that will hide/show the characters.	Zhaoji Jiang	2

Acceptance Criteria:
● When a user clicks the hide/show toggle button next to a password text field in the data entry browser, the password text (characters) will change between dots (or asterisks) and their actual ASCII values on alternate clicks of the button.
● When a user clicks the hide/show button and the password field is showing the actual characters, then the characters will become dots (or asterisks).
● When a user clicks the hide/show button and the password field is showing dots (or asterisks), then the characters will become the actual ASCII values of the password.
● When a user copy and pastes a password from the text field, it will be the actual characters of the password, regardless of whether or not the password is hidden/shown.

Test Case: Toggle between hidden and shown passwords
Instructions:
1. Select a data entry from the browser that contains a password field (such as Account Login).
2. Next to the password field, click the Hide/Show button.
3. Click the Hide/Show button again.

Expected Results:
1. The data entry's information will be displayed in the data entry viewer of the Home window. By default, the password text will be hidden (characters as dots/asterisks).
2. The characters of the password will turn from dots/asterisks to their actual ASCII value.
3. The characters of the password will revert to their hidden form (characters as dots/asterisks).

User Story (Manual Backups):
● As a user, I want to manually backup my account and all its associated data to a database file.
● As a user, I want to specify the storage location for backup files.

Task Description	Owner	Time Estimate
Implement a BackupView UI to handle all backup-related functionality.	Dingfu Sun	2

Implement a file chooser.	Jiho Choi	5
Implement functionality to create a database backup file containing a user's account and data entry information.	Adam Petty	6
Implement functionality to validate the owner of a backup file and import the contained entries into the current user account.	Adam Petty	6

Acceptance Criteria:
- When the user clicks "Backup Account" on the Home window, a confirmation will appear asking the user if they want to backup their account.
- When the user confirms the backup, a new backup database file will be created at their specified location (chosen in User Settings) containing their user account and all associated data entries.
- When the backup file is created, the data contained within it will be encrypted.
- When the user clicks "Import Account" on the Home window, a file chooser window will open asking for the user to pick a database backup file to import.
- Given the user has selected a database backup file to import, when the user clicks "Import", the data entries of the backup will be added to the user's account *provided that they enter the correct account credentials of the account being imported*.
- Given the user has not entered the correct account credentials of the account being attempted to import, then the import will not occur and the user will be informed that they are not authorized to access this backup file.

Test Case: Successfully creating an account backup and importing its entries into a new account.
Instructions:
1. From the Backup Window, click "Backup Account"
2. Click "Yes" to confirm.
3. Sign out of the current account by clicking "Sign out" on the Home window.
4. At the Sign in window, sign in to a different account either by creating a new account or signing in to an existing account.
5. From the Home Window, click "Import"
6. Choose a backup file to import and click "Select"
7. Enter the correct account sign in credentials of the account to be imported and click "Verify and Import..."

Expected Results:
1. A confirmation will appear asking the user if they want to create a backup database file of their account.
2. The confirmation will close and a backup database file will be generated at the location specified by the user account settings.
3. The Home Window will close and the Sign in window will open.
4. The Home Window of the new account will open and display the user's data entries (if any).

5. A file chooser window will open asking the user to select a backup file to import from.
6. An account validation window will open asking the user to enter the account credentials of the backup file to be imported.
7. The user entries of the account of the backup file will be added to the current user's data vault.

User Story (Automatic Backups):
- As a user, I want the option for backups to occur automatically on a user-configured interval.

Task Description	Owner	Time Estimate
Implement functionality for the data vault system to automatically control backups based on the user's backup frequency setting.	Dingfu Sun	5

Acceptance Criteria:
- When the user selects a backup frequency (other than "Manual") in the User Settings window, then the data vault will begin automatically creating account backups at the location specified by the user's account settings.
- When the user first turns on automatic backups (by selecting a backup frequency other than "Manual"), the data vault will immediately create an initial backup at the location specified by the user's account settings.
- Given the amount of time specified by the user's backup frequency settings has passed, when the user signs into their data vault, then a backup file of their account will automatically be created at the location specified by the user's account settings.

Test Case: Successfully turning on automatic backups and automatically creating an account backup file.
Instructions:
1. In the Settings Window, select a backup frequency interval other than "Manual" (for testing purposes, select 15 seconds) and click "Save"
2. Check the specified backup location on the user's computer.
3. Wait for a period of time set for the backup frequency (15 seconds for this test).
4. Check the specified backup location on the user's computer.
Expected Results:
1. The Settings Window will close, returning the user to the Home Window.
2. A backup file of the user's account will exist at the location specified by the user's account settings.
3. No new backups will be created.
4. A new backup file of the user's account will have replaced the existing backup file (if one exists) at the location specified by the user's account settings.

<u>**User Story (Shared Folder):**</u>
- As a user, I want to have a permissions-based shared folder that will contain data entries made available to me by different users across multiple accounts.

Task Description	Owner	Time Estimate
Create a "Share" button in the Home window.	Dingfu Sun	1
Implement a "Shared" folder in the data entry browser with the functionality to properly track and maintain data entries that have been shared with the current users by other data vault users, as well as to prohibit modification of its entries.	Jiho Choi	8
Implement a "ShareView" UI window for a user to enter the email (username) of an account they want to share with.	Dingfu Sun	4

Acceptance Criteria:
- Given the user has selected a data entry from the data entry browser, when the user clicks the "Share" button in the Home window, the "Share" window will open.
- Given the user is in the "Share" window, when the user enters the valid email (username) of the user they want to share with and clicks "Share with user", then the specified user will be able to see this entry in their shared folder.
- When the owner of a data entry that has been shared with another account deletes said data entry, then it will also be removed from the shared folder of the other account.
- When the user deletes an entry from their shared folder, the entry will only be removed from the shared folder, but not from the account of the entry owner.
- When the user attempts to edit an entry in their "Shared Folder", they are prevented from doing so (unable to edit the entry).

Test Case: Successfully share a data entry with another account.
Instructions:
1. User A selects an data entry and clicks the "Share" button.
2. User A enters the valid email (username) of a User B in the "Share with" field and clicks "Share with user" button.
3. Sign out as user A; sign in as User B.
4. Click the "Shared Folder" and select the newly-shared data entry.
5. Click the "Edit Entry" button.

Expected Results:
1. The "Share" window will open with a field to enter the email (username) of the account to share the entry with (i.e. to have the entry added to its shared folder).
2. The "Share" window is closed and User A is returned to their Home window.
3. User A is signed out and User B is signed in to their data vault Home window.
4. User B is able to see the shared data entry and view its information
5. User B is unable to edit the entry.

- As a user, I want to be able to share a data item securely with another non-local user over email. (e.g. sharing a WiFi password)

Task Description	Owner	Time Estimate
Implement the "Send Entry" button in the "Share" window.	Dingfu Sun	1
Implement functionality to convert a data entry into a send-able format, and then send an email containing the encrypted information of the data entry to be shared.	Dingfu Sun	5
Implement functionality in the ShareView to display a place for the user to input entry information to be decrypted.	Thomas Worns	3

Acceptance Criteria:
- When the user hits the "Send Entry" button in the "Share" window, an email will be sent to the user-specified address (username) with the data entry's encrypted information.
- The email sent must contain all the fields of the data entry in an encrypted format.
- Given the user as received an encrypted data entry, when the user copies and pastes the entry fields into the decrypt area of the "Share" window and clicks "Decrypt", then the data entry's information will be displayed in human-readable format.

Test Case: Successfully sharing an encrypted data entry; decrypting a received data entry
Instructions:
1. User A selects a data entry from the data entry browser of the Home window and then clicks the "Share" button.
2. User A types in the valid email (username) of User B – the account to receive the data entry – into the "Share with" field of the Share window and clicks "Send Entry" button.
3. User A signs out of the data vault; User B signs in to the data vault.
4. User B copies the encrypted content of the received data entry email (instructions will be on the email), pastes it into the decrypter field of the Share window, and clicks "Decrypt"

Expected Results:
1. The "Share" window will open with a field to enter the valid email (username) of the account to share the entry with (i.e. to have the encrypted entry sent to their address).
2. User A will see a success message informing them that their encrypted data entry has been sent to User B.

3. User A will be returned to the Sign in window; User B will be signed in to their own data vault.
4. User B will now see the contents of User A's data entry in human-readable format.

User Stories (Password Change Reminder):
● As a user, I want to be reminded to change my data vault password after a set period of time.

Task Description	Owner	Time Estimate
Implement a pop-up window indicating that the user needs to change his/her data vault password.	Dingfu Sun	1
Implement a method of detecting data vault password changes and tracking the time between these changes.	Dingfu Sun	5
Implement functionality to store user password reminder settings in the database.	Zhaoji Jiang	1

Acceptance Criteria:
● Settings must contain a new option for setting the time interval between password change reminders, including Weekly, Monthly, Biannually, and Annually.
● Settings must contain an option for the user to turn off password change reminders.
● Given the user-set amount of time has passed since the user has changed their data vault password, when the user signs into their data vault, a pop-up window will be displayed informing them that they need to change their data vault password in the Settings window.

Test Case: Successfully remind the user to change their data vault password
Instructions:
1. Create a new data vault account.
2. Immediately sign in to the new account.
3. Sign out of the account.
4. Wait; do not change password for the user-set amount of time for password change reminders (use the "1 minute" option for testing purposes).
5. Sign back into the account.
Expected Results:
1. New data vault account is created.
2. The new user will be taken to a fresh Home window and will not see any pop-up reminder.
3. The Home window will close and the Sign in window will open.
4. Time passes.

5. A pop up-window will appear reminding the user that they have not changed their data vault password in [user-set amount of time (1 minute for testing purposes)].

User Stories (User Account Avatar Pictures):
   ● As a user, I want to have an avatar picture associated with my account.

Task Description	Owner	Time Estimate
Implement AvatarSelectionView where a user can select a picture they would like to use from a set of account pictures.	Thomas Worns	4
Implement a UI element of HomeView to load and display user account avatars.	Thomas Worns	3
Implement functionality to store user account picture choice data in the database.	Zhaoji Jiang	1
Implement a "Change Picture" button in the SettingsView.	Zhaoji Jiang	1

Acceptance Criteria:
   ● When the user clicks "Change Avatar" in the Settings window, an Avatar Selection window will open displaying the available pictures to choose from.
   ● When the user selects a picture, that picture is displayed as the user's avatar picture on the Home window.
   ● If the user has not selected an avatar picture, then a default one will be displayed in its place until the user chooses one.

Test Case: Successfully select and view an account avatar.
Instructions:
   1. Click "Change Avatar" in the Settings window.
   2. Select an avatar picture from the given set and click "Set Avatar".
   3. Click "Save" in the Settings window.
Expected Results:
   1. The Avatar Selection window will open displaying the available avatar pictures to choose from.
   2. The Avatar Selection window will close, returning the user to the Settings window.
   3. The Settings window will close, returning the user to the Home window. The newly-chosen avatar will be seen on the Home window.

User Stories (Virtual Keyboard):
- As a user, I want to have a built-in virtual keyboard for text input.

Task Description	Owner	Time Estimate
Implement a system-wide virtual keyboard to enter text information.	Adam Petty	15
Create a button to activate the virtual keyboard from the Home window.	Zhaoji Jiang	1

Acceptance Criteria:
- Given the virtual keyboard is not open, when the user clicks the virtual keyboard icon, then a separate, moveable keyboard will open.
- Given the virtual keyboard is open, when the user clicks the virtual keyboard icon, then it will close.
- Given the user's cursor is in a text-entry field, when the user clicks on a key on the virtual keyboard, the clicked-on characters will appear in the text-entry field.

Test Case: Successfully open, close, and enter text using the virtual keyboard.
Instructions:
1. From the Home window, click the virtual keyboard icon.
2. Click on a text-entry field anywhere within the data vault (e.g. "New Entry").
3. Enter text by clicking on characters of the virtual keyboard.
4. Click the virtual keyboard icon again.

Expected Results:
1. The virtual keyboard will open. It is usable from any window and moveable.
2. The cursor will appear in the text-entry field.
3. The clicked-on characters will appear in the text-entry field.
4. The virtual keyboard will close.

User Stories (UI Refinements, Code Optimization, Miscellaneous):
- As a user, I want the application to run smoothly and be highly responsive.
- As a user, I want a visually appealing interface, largely free of the clutter seen in existing managers.
- As a user, I want an intuitive and easily navigable interface.

Task Description	Owner	Time Estimate
Redesign PasswordChangeView, SecurityView, and SettingsView.	Dinfu Sun	2
Attempt to merge data entry creation and modification windows into the data entry viewer of the Home window.	Dingfu Sun	2
Unify styles across all windows (colors, spacing, typefaces, language, layout, UI elements, alignment,	Dingfu Sun	2

etc.).		
Attempt to reduce the overall number of windows that are on screen/pop-up at one time (e.g. eliminating warning message pop-ups and moving them to the current window).	Dingfu Sun	2
Optimize code where possible (e.g. sending emails via threads, more efficient database accessing, minimize database accesses, speed up encryption, better resource management, etc.)	Zhaoji Jiang	5
Implement ability for user to set the number of failed sign in attempts to trigger an account wipe.	Zhaoji Jiang	1
Implement functionality to store account-wipe-attempts and counter in the database.	Zhaoji Jiang	1
Add options to set account wipe and high security on/off at account creation.	Zhaoji Jiang	1
Add ToolTips (context-based help) to all newly added features/settings/fields.	Zhaoji Jiang	1

Acceptance Criteria:
- The required fields to be entered must be made more clear to the user in PasswordChangeView.
- SecurityView must have guidelines for how to use the Password Generator and Strength Checker.
- SettingsView must contain all newly implemented settings with user-adjustable settings where applicable.
- Data entry creation and modification must be performed within the data entry viewer of the Home window rather than a separate windows of their own.
- All windows system-wide must follow the same general style guidelines: colors, spacing, typefaces, language, layout, UI elements, and alignment.
- The number of separate windows that open over the course of system usage must be reduced, and warning messages must be contained within their own window (e.g. showing red error text next to a problematic field).
- Emails must be sent in their own program threads to prevent the system from halting during this process.
- In general, the system must run quicker more smoothly due to fewer, more efficient database accesses, minimized resources leaks, and faster encryption (as long as security is not sacrificed).
- UI / UI elements must be more intuitive and obvious in their function and layout.
- The user must be able to set the number of failed sign in attempts to trigger an account wipe in Settings.

- The user must be able to set account wipe and high security on/off from the Account Creation window during sign up.
- All features/settings/fields implemented whose purpose or function is not abundantly clear must have context-based help in the form of ToolTips.

REMAINING BACKLOG
Functional:

1. As a user, I want to have two-factor authentication for account access. (*If time allows*)
2. As a user, I want to have the ability to attach images to a data item. (*If time allows*)
3. As a user, I want to have an overall security rating based off total password strength. (*If time allows*)
4. As a user, I want to have automatic internet form filling option. (*if time allows*)