| **Team Number** | 04 | **Section** | 05 |
| --- | --- | --- | --- |
| **Team Members** | 1. Lovepreet Uppal 2. Karan Partap Singh Virk 3. Nhat Le 4. Wen Luo | | |
| **Software Name** | Passaver | | |

## Problem Statement

Computers and the Internet are being used extensively and the security of users' privacy has become a big issue. One of the oldest ways to provide security is using passwords.

Choosing and storing passwords are challenging tasks for many users and cause breaches. Since the number of websites that we use every day is increasing, this causes most users to choose simple memorable passwords, store them improperly, or reuse them for many websites/applications that increase the risk of being cracked.

The existing password managers are not reliable because there is always the risk of leaking our information to the developers of the software!

## Functional Requirements

This product has these features:

1. The application shall be multiuser.

2. The user shall log in/signup/reset password when running the application. The user's login password (aka master password) shall be reset by one security question that is set during signup.

4. The application shall store the username, password, website/application name, specified email, creation date, expiration date

5. The application shall generate a password based on the given criteria.

6. The passwords shall be encrypted before storing and decrypted before   
retrieving. For security purposes, the code of encryption/decryption will be provided by the client (the professor).

7. The password shall be copied in the clipboard by a button.

8. The application shall warn the user about the expired passwords right after successful login to the system.

9. The user shall search based on the website/application name or username.

10. The user shall modify the information.

## References

None

## Use Cases

| **Use Case Name** | | Login | |
| --- | --- | --- | --- |
| Goal | | | |
| This use case allows users to login to the application | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User click on the “Login” button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User types in the userId | | System display text area for userId |
| 2 | User types in the password | | System display text area for password |
| 3 | User presses Login button | | System checks the database for user credentials. |
| 4 |  | | System prompts user to the home page |
| 5 |  | | System display error if data dismatch |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| The user clicks on the "Forgot Password" button  The user clicks on the "Sign-Up" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User click forgot password | | System prompts the User to the forgot password page |
| 2 | User click on sign-up | | System prompts the User to the sign-up page |
| *3* |  | |  |

| **Use Case Name** | | Sign-up | |
| --- | --- | --- | --- |
| Goal | | | |
| This use case describes how a user sign-up for the system. | | | |
| Participating Actors | | | |
| user, system | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User runs the application  User click on “Sign-up" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | The user clicks on the sign-up button on the main display page | | System prompts the user the sign-up page |
| 2 | The user enters UserId, Password, Confirm Password, | | System provides UserId, Password, Confirm Password |
| 3 | The user chooses security question | | System display options for security questions |
| 4 | The user enters Answer | | System provides the text area for security question Answers |
| 5 | The user clicks on Sign Up button | | System creates an account for a new user |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| The user clicks on "Login" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | The user clicks the Login button | | System prompts the user to the log-in page |
| 2 | The user enters the invalid confirm password | | System displays error message |
| *3* |  | |  |

| **Use Case Name** | | Forgot Password | |
| --- | --- | --- | --- |
| Goal | | | |
| This describes how the user can change the password on the login page. | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| Users click on the “Forgot Password” Button on the profile page. | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on Forgot Password | | System display Reset Password Page |
| 2 | User clicks on UserId | | System display text field for UserId |
| 3 | User click on options for security Question | | System record the option |
| 4 | User click on Answer | | System provides text field for Answer |
| 5 | User click on New Password | | System display text field for New Password |
| 6 | User click on Reset Password | | System record the new password into database |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| The user clicks on "Login" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on “Login” butt | | Return user to home page |
| 2 | Enter incorrect old password | | Show error message |
| *3* | Enter incorrect confirmed password | | Show error message |

| **Use Case Name** | | Search for app/web name/email | |
| --- | --- | --- | --- |
| Goal | | | |
| Allow users to search for a password that they stored. | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User clicks on Search button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User enters the keywords to the search field and click “Search” button | | Find the emails, website names that match the keywords, then display them on the page. |
| 2 |  | |  |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| The user clicks on "Cancel" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on “Cancel” button while entering keywords | | Clearing the search field |
| 2 |  | |  |

| **Use Case Name** | | Save a new password | |
| --- | --- | --- | --- |
| Goal | | | |
| Creating a new password for a website or application | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User clicks on “Save New Password” button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on “Save New Password” button | | Show user the page to enter information |
| 2 | User enters the website’s name, username, email and generate the password for the website | | Store the website’s name, username, email and password to database |
| 3 | User click on Expired after | | System saves expiration date |
| 4 | User click on generate pass | | System generates a random password |
| 5 | User click on Save Info button | | System record new data into database |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| The user clicks on "Cancel" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on “Cancel” button | | Return user to the home page |
| 2 |  | |  |

| **Use Case Name** | | Edit detail | |
| --- | --- | --- | --- |
| Goal | | | |
| Allow user to update their password, email, name, username of a website | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User clicks on an entry listed on the Saved Passwords page | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on the Edit button on the home page | | Show user the page to update information |
| 2 | User enters the website’s name, username, email and generate the password for the website | | update the website’s name, username, email and password to database |
| 3 | User click on Expired after | | System saves expiration date |
| 4 | User click on generate pass | | System generates a random password |
| 5 | User click on Update Info button | | System record new data into database |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| The user clicks on "Cancel" button | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User clicks on “Cancel” button | | Return user to the Saved Passwords page |
| 2 |  | |  |
| *3* |  | |  |

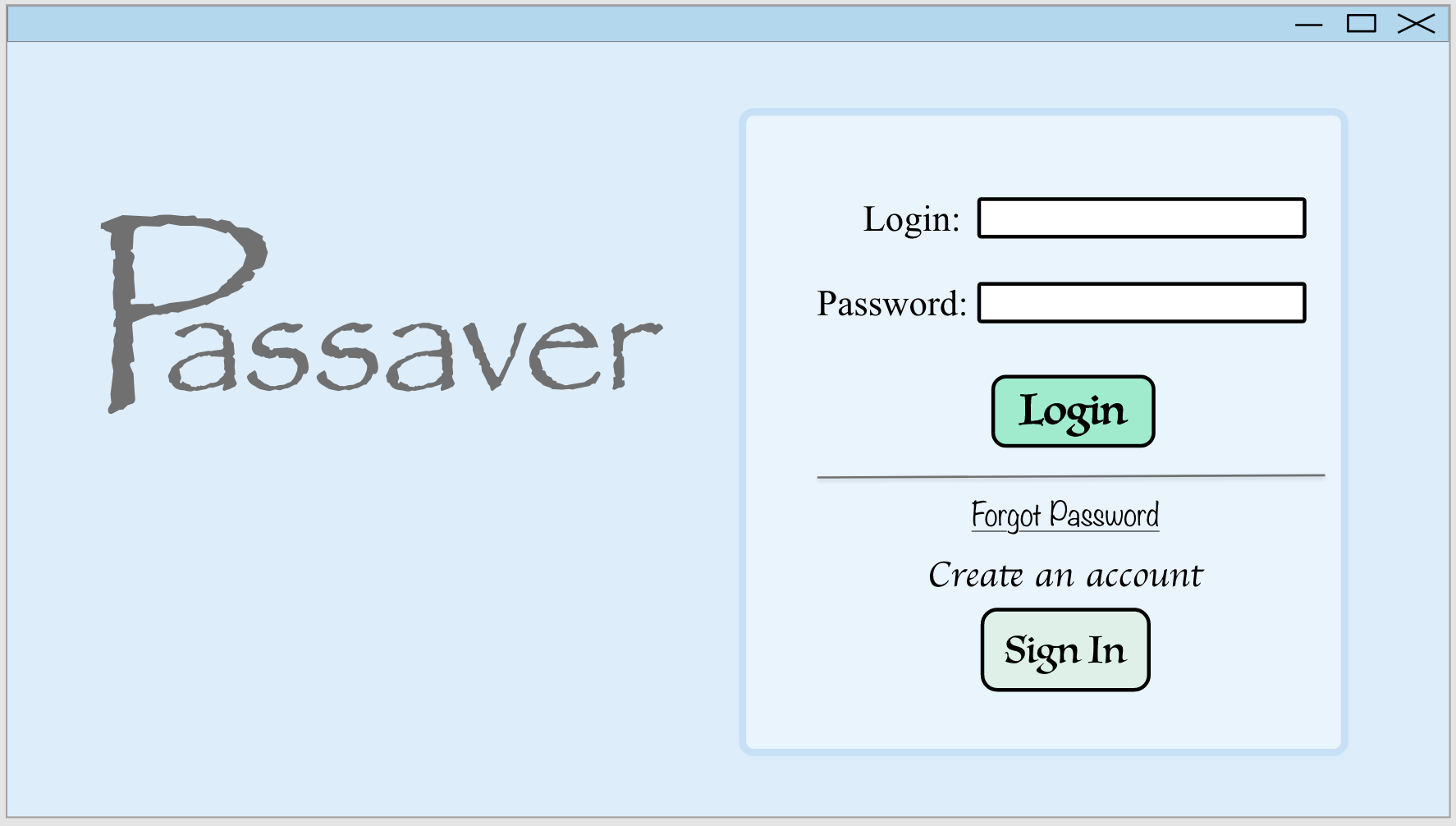
| **Use Case Name** | | Expired Password | |
| --- | --- | --- | --- |
| Goal | | | |
| This use case describes how a user will be reminded about a password expiration | | | |
| Participating Actors | | | |
| System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| A user logged into the software | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User log-in into the application | | System displays a reminder regarding password expiration for an application/website |
| 2 | User Click “Go to Home” button | | System routes the user to the “saved passwords” page |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| N/A | | | |
| **Steps** | **Action** | | **System Response** |
| 1 |  | |  |

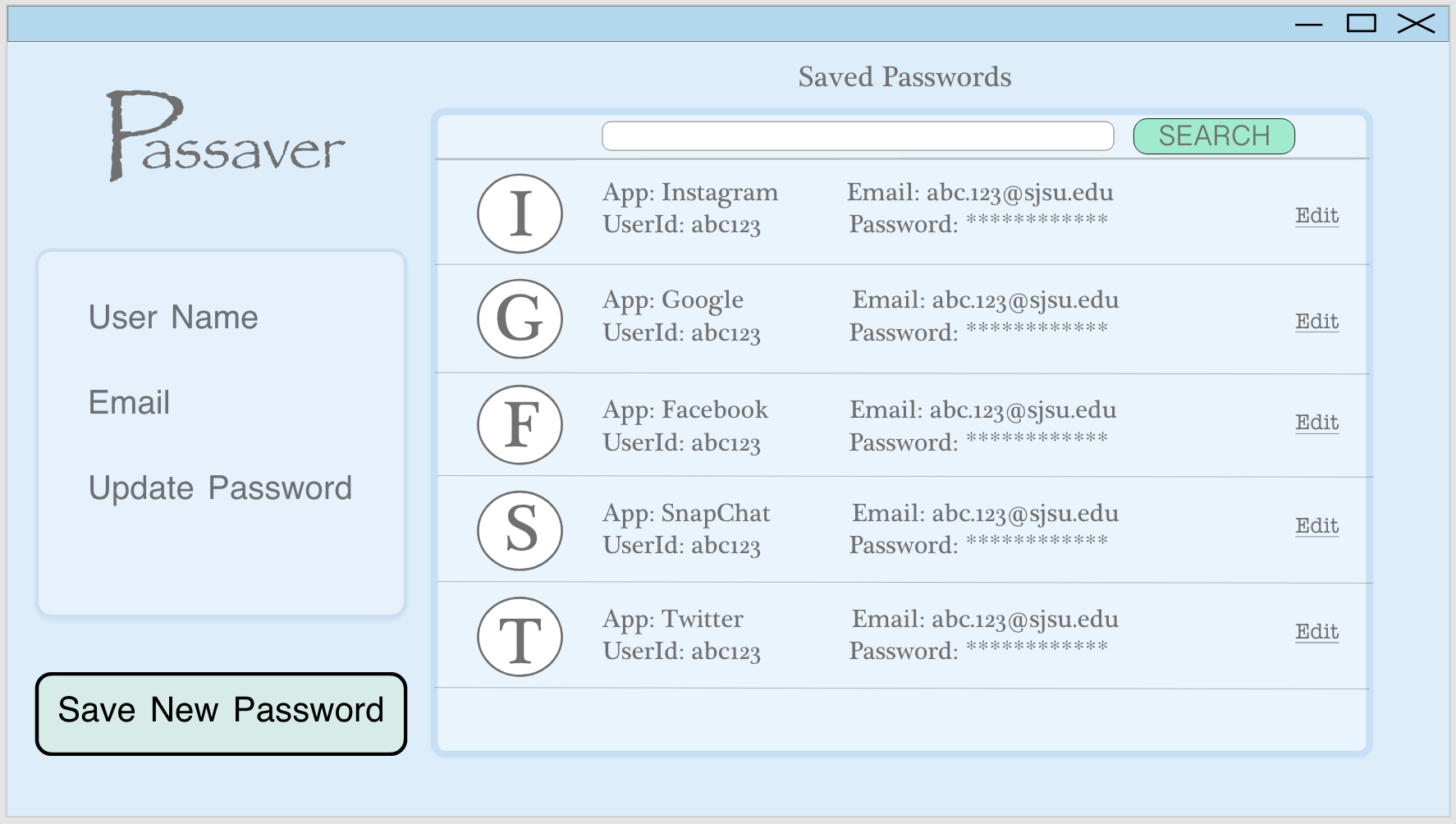
| **Use Case Name** | | Delete Details | |
| --- | --- | --- | --- |
| Goal | | | |
| This use case describes how a user will delete its password and other related information | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User click on the “Delete” Bottom | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User click on the bottom “Delete” on the homepage | | System delete any website related information. |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| N/A | | | |
| **Steps** | **Action** | | **System Response** |
| 1 |  | |  |

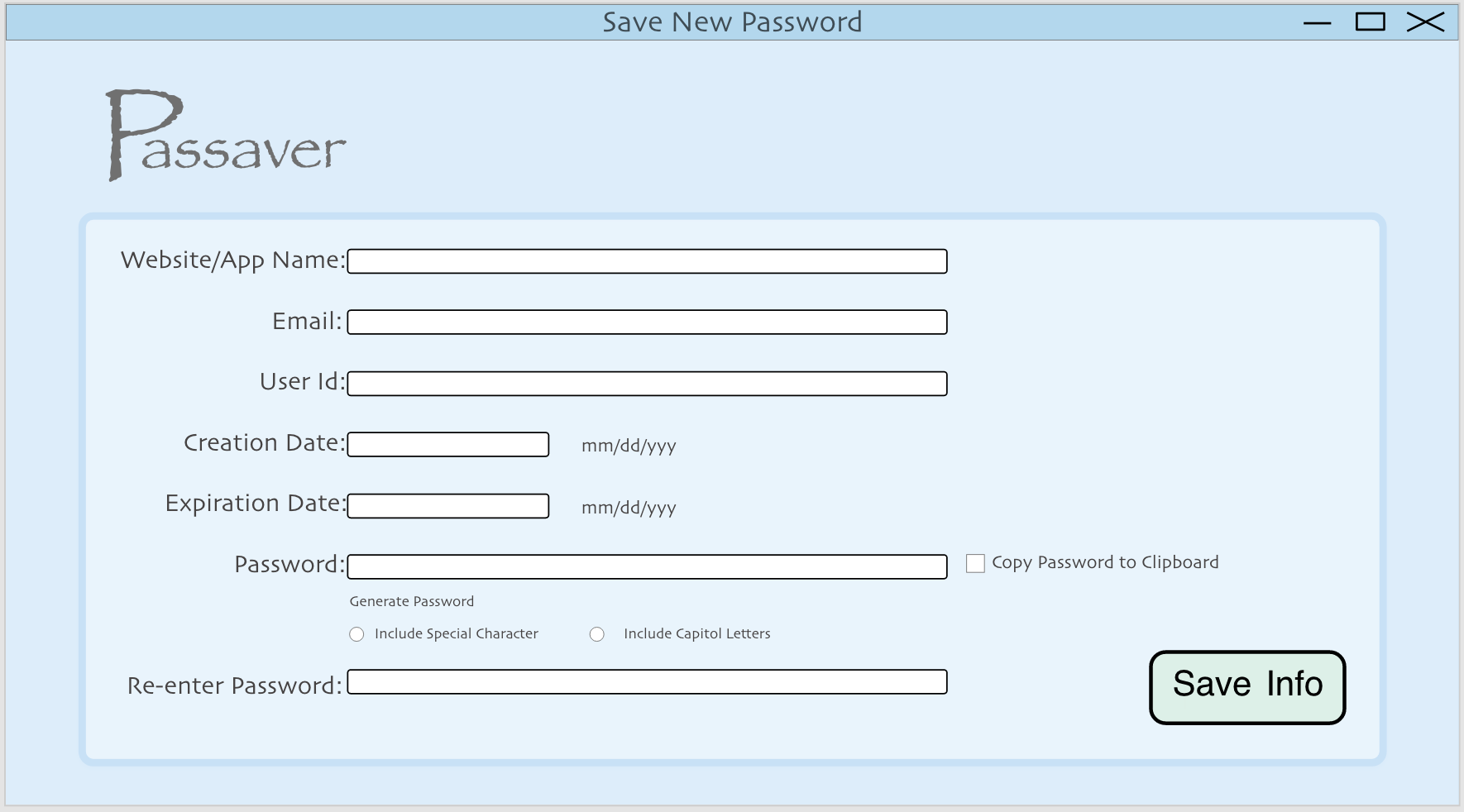
| **Use Case Name** | | Copy To Clipboard | |
| --- | --- | --- | --- |
| Goal | | | |
| This use case describes how a user will copy its password to the clipboard | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User click on the “Copy To Clipboard” Bottom | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User click on the bottom “Copy To Clipboard” on the homepage | | System copy the password from database |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| N/A | | | |
| **Steps** | **Action** | | **System Response** |
| 1 |  | |  |

| **Use Case Name** | | Logout | |
| --- | --- | --- | --- |
| Goal | | | |
| This use case describes how a user will log out a specific account | | | |
| Participating Actors | | | |
| User, System | | | |
| Glossary | | | |
|  | | | |
| Primary Flow of Events | | | |
| Trigger | | | |
| User click on the “Log Out” Bottom | | | |
| **Steps** | **Action** | | **System Response** |
| 1 | User click on the bottom “Logout” on the homepage | | System log out the current user session |
| 2 |  | | System route the current page to “Login” page |
| Alternate Flow of Events | | | |
| **Alternate Trigger** | | | |
| N/A | | | |
| **Steps** | **Action** | | **System Response** |
| 1 |  | |  |

## Mockup







## Glossary

| None |  |
| --- | --- |
|  |  |
|  |  |