**MyPass Project Report**

**Team Members:**

**Abdulkader Qayed (worked alone)**

**Github URL**

**https://github.com/kadercoder0/MyPass**

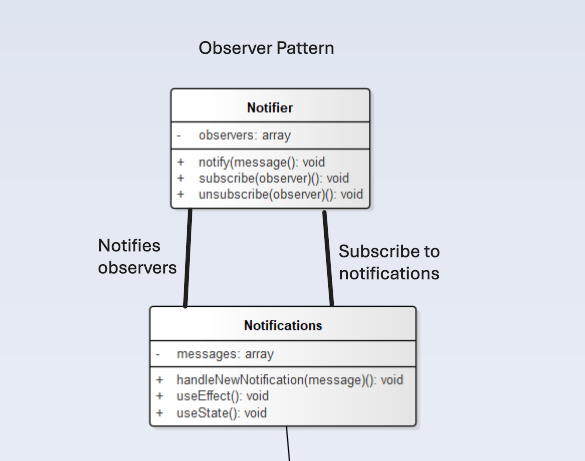
**Video Presentation URL  
https://youtu.be/kPNU2iwEUT0**

**Class Diagrams**

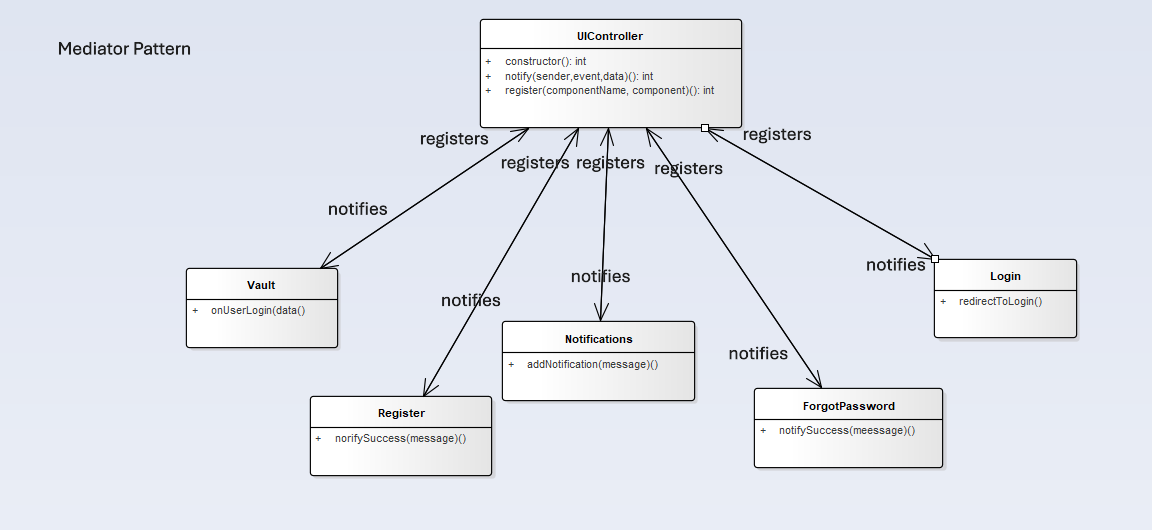
A screenshot of a computer

Description automatically generated

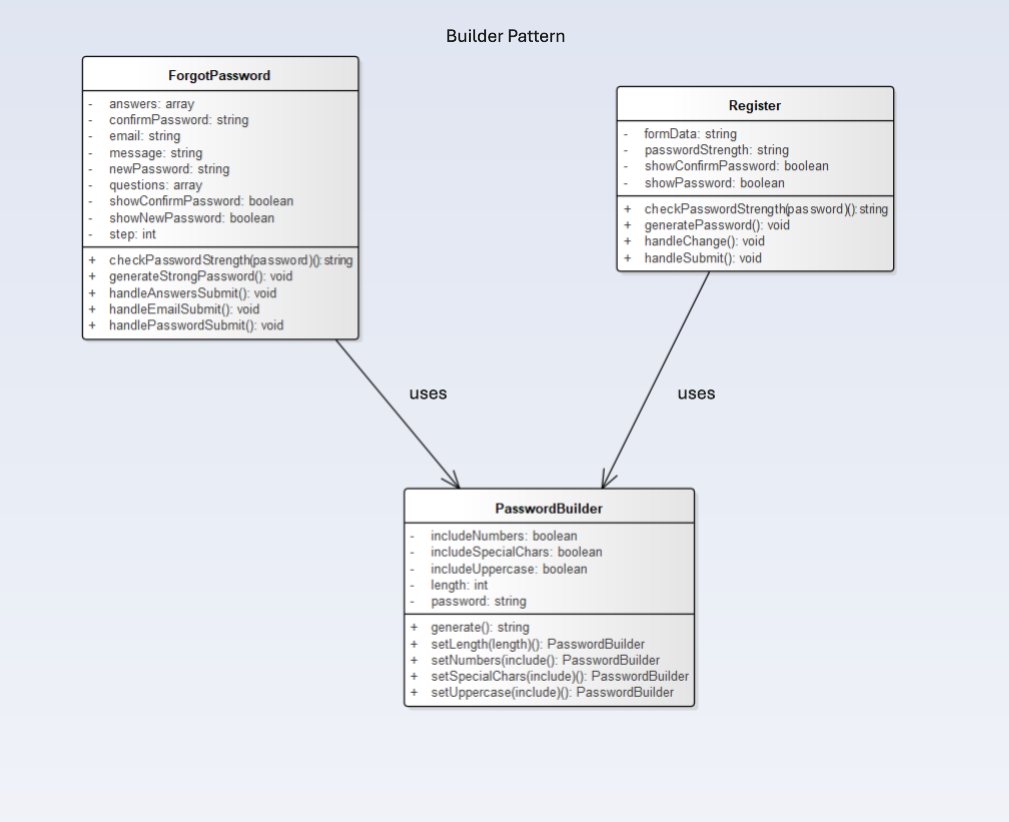
The SessionManager class is implemented as a singleton to ensure that only one instance of the session exists at any given time. It manages user sessions and interacts with other classes like Login and Vault, ensuring consistent access to session-related data across the application.



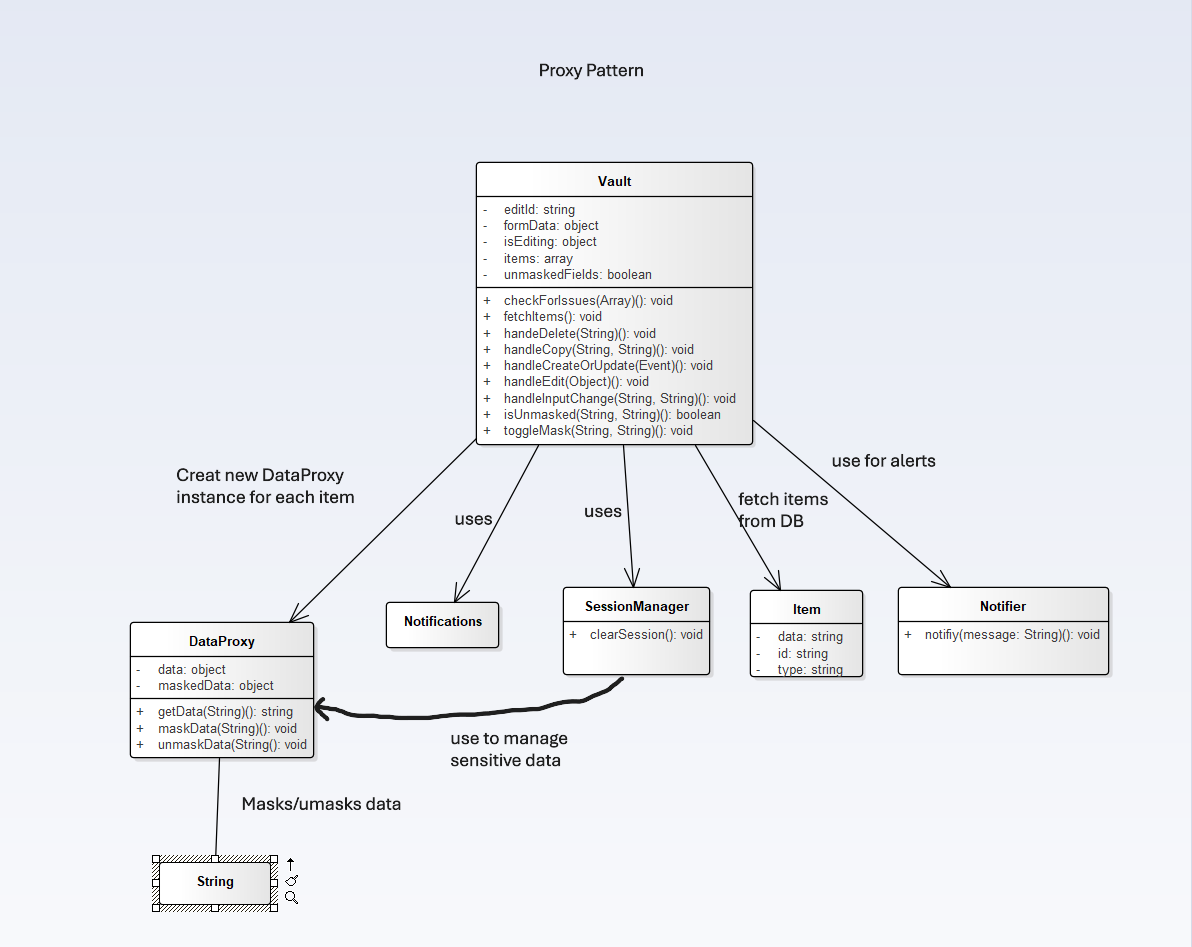
The Notifier class notifies observers, such as the Notifications class, when a new message needs to be sent. Observers can subscribe to notifications and react accordingly, allowing the system to handle multiple alerts or messages dynamically.



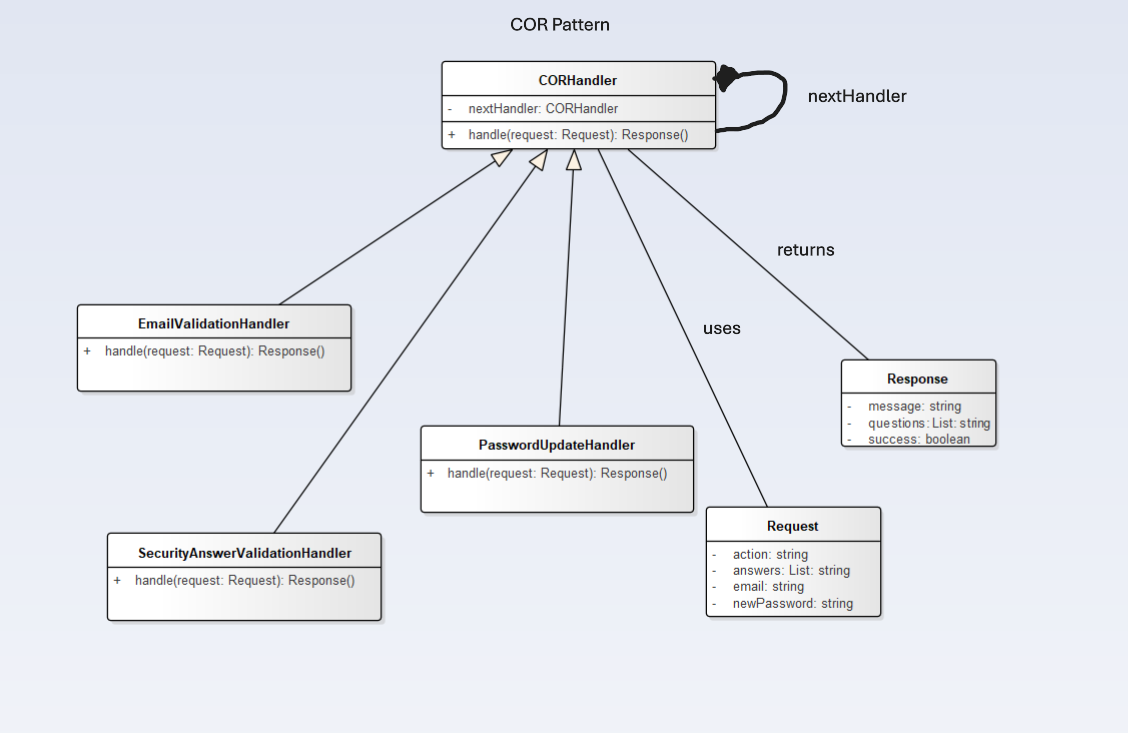
The UIController class acts as a mediator that manages communication between different components like Vault, Register, Login, ForgotPassword, and Notifications. It ensures loose coupling between these components by facilitating their interaction through a centralized interface.



The ForgotPassword and Register components use the PasswordBuilder class to generate secure passwords. The builder pattern is used to construct passwords step by step, providing flexibility in password creation based on user inputs.



The Vault class uses the DataProxy to manage sensitive data. The DataProxy controls access to the underlying data, ensuring that sensitive information is masked or unmasked as needed, while interacting with other components like SessionManager, Item, and Notifier.



The CORHandler chain processes requests in a sequential order. Different handler classes, such as EmailValidationHandler and PasswordUpdateHandler, manage various stages of request validation, and each handler either processes the request or passes it to the next handler.

**Database Schema**

**Entities:**

1. **User**
   * id (Primary Key)
   * email
   * password
   * security\_question\_1
   * security\_answer\_1
   * security\_question\_2
   * security\_answer\_2
   * security\_question\_3
   * security\_answer\_3
   * created\_at
2. **Vault Item**
   * id (Primary Key)
   * user\_id (Foreign Key to User id)
   * type (Could be CreditCard, Login, Identity, SecureNote, etc.)
   * data (Contains a serialized JSON string with sensitive data)
   * created\_at

A screenshot of a computer

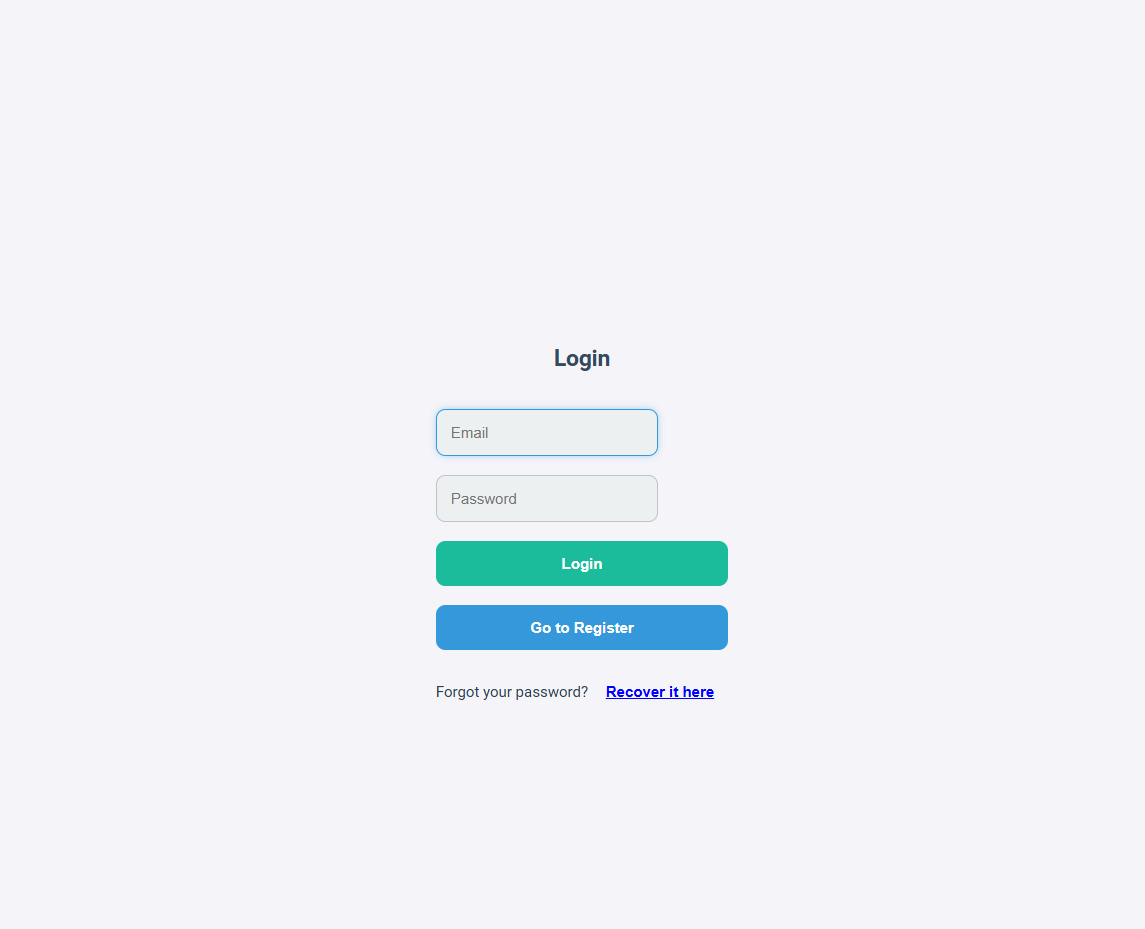
Description automatically generatedA screenshot of a computer

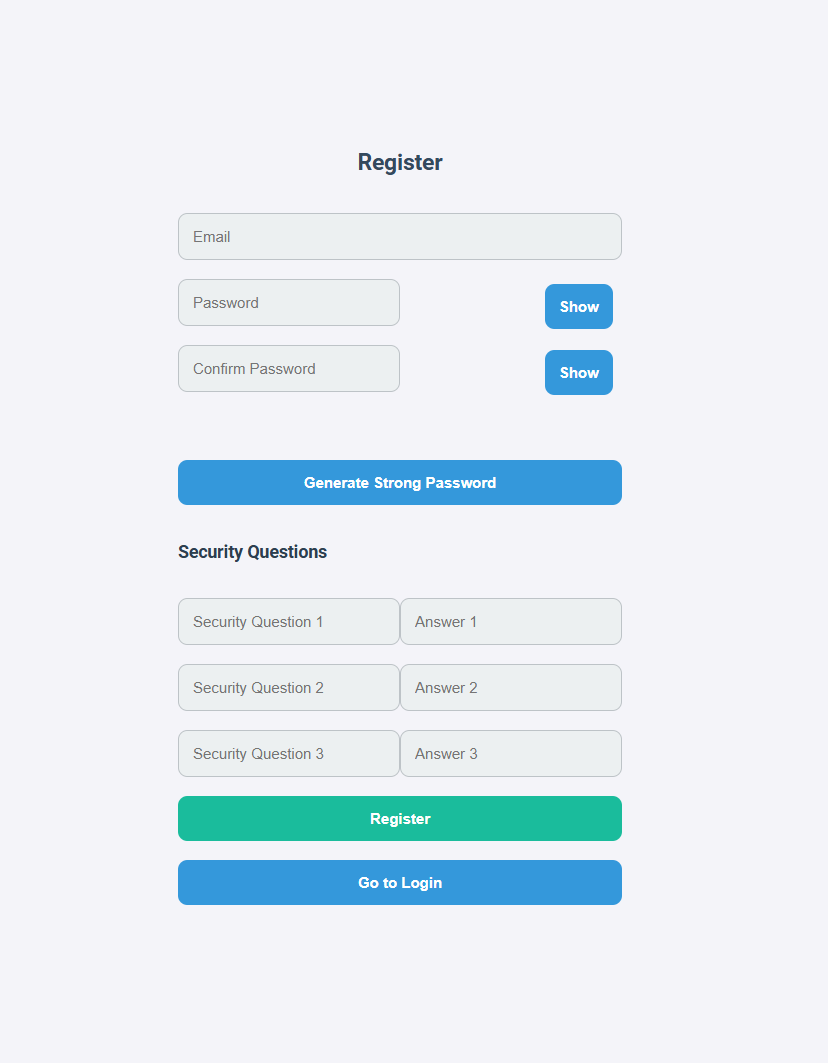
Description automatically generated

In the MyPass project's database schema, the User table stores essential user information, such as their email, password, security questions, and answers, along with a timestamp indicating when the account was created. The Vault Items table holds various sensitive data related to a user, like credit card information, login credentials, or personal documents, with each item linked to the respective user through a foreign key. The relationship between the User and Vault Items tables is one-to-many, meaning each user can have multiple vault items associated with them.

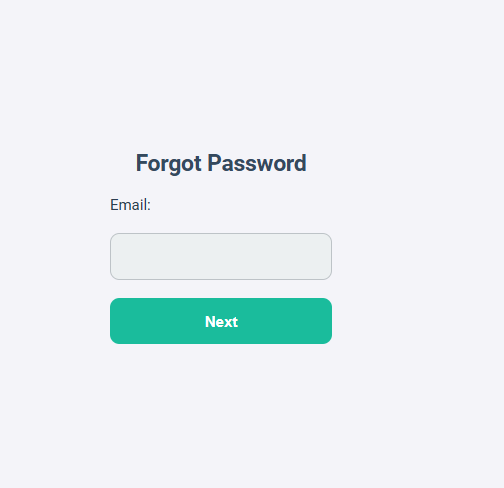
**UI Screenshots**

**Login Page**

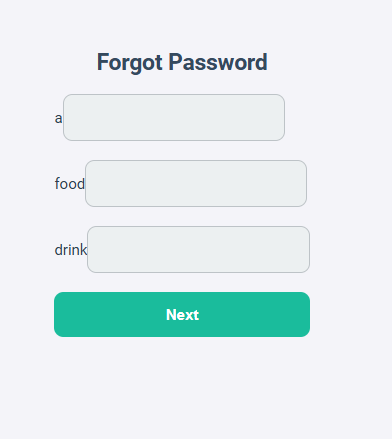
****

**Register Page**

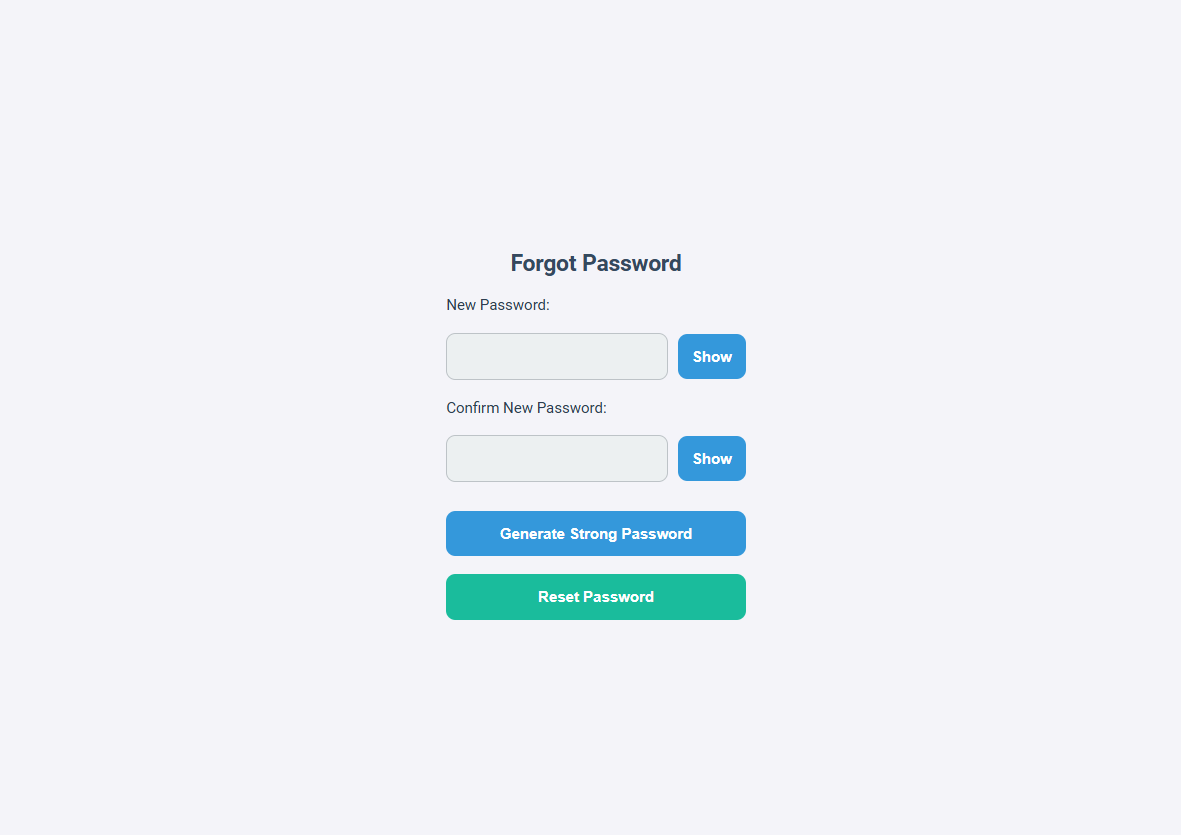
**Forgot Password Part 1**

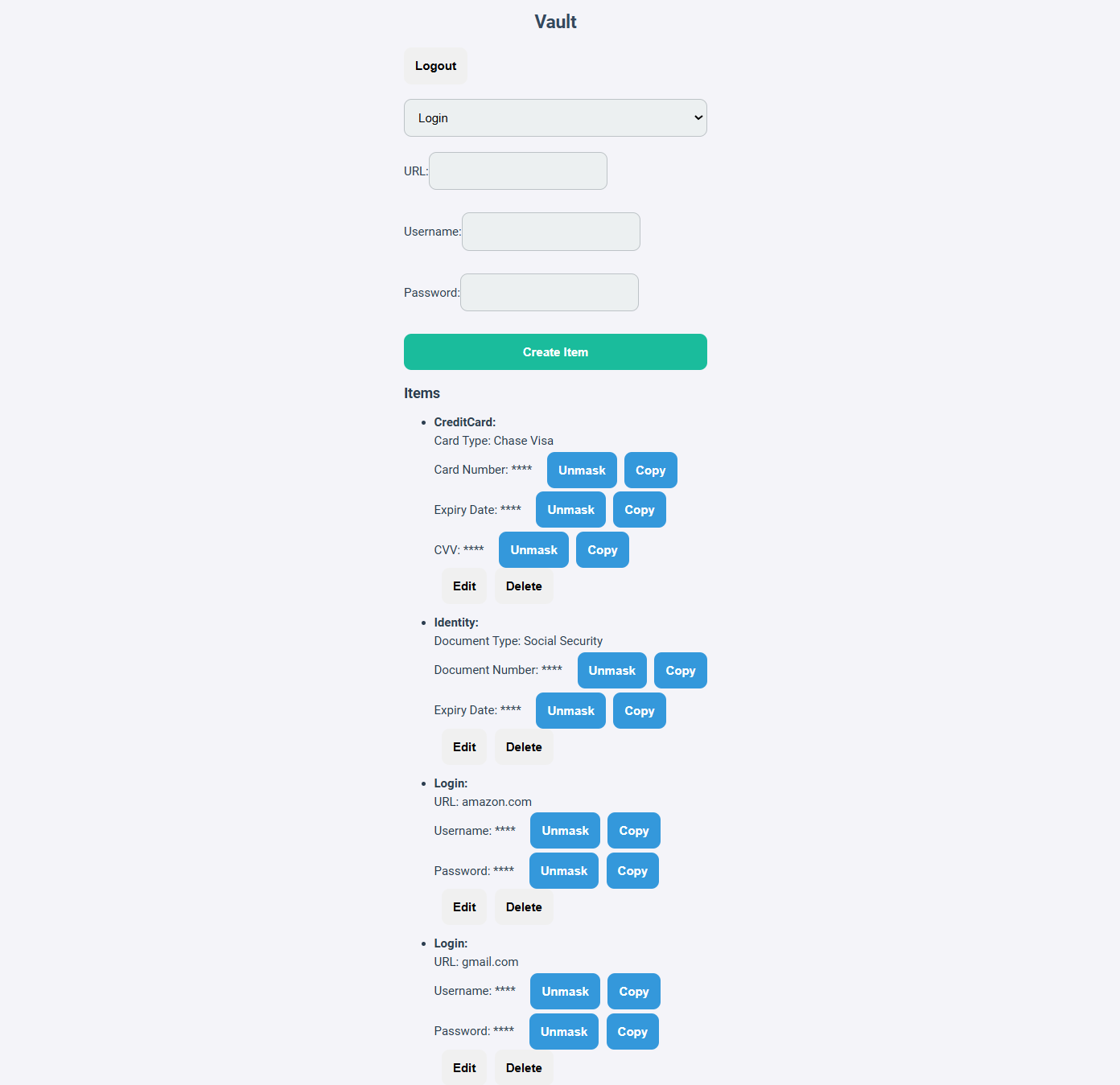
****

**Forgot Password Part 2**

****

**Forgot Password Part 3**

****

**Vault Page  
  
**