# List of features and how to use them (& how they link to other functions)

1. **Accounts Functionality**:
   * **Create Account**
     + Description: This feature allows users to create a new account with their personal information
     + How to Use: Users can access the account creation page through the "Sign Up" link on the login page of the website. They need to fill out the required fields such as name, email, and password, and then submit the form to create their account.
       1. Once the user makes their account, they are taken back to the login page to login to their newly created account.
     + Links to Other Functions: This function interacts with the security features for data encryption, password hashing, and input sanitization to ensure secure account creation and updates.
   * **Profile/Edit Profile**:
     + Description: This feature allows users to view their profile which displays their name, roles, biography, contact information, profile picture and gallery. and to make edits to
     + How to Use: After a user logs in to their account, they are taken to their profile page. To edit the profile, click on the “Edit Profile” button which allows users to edit all the fields on their profile except for their profile picture and gallery (will be implemented in the future). The “Cancel” button on the editing form will make the form disappear.
     + Links to Other Functions: Data on the profile page is populated from the account creation page. Any edits made on the profile page update the corresponding data in the database. Upon saving changes, the profile page fetches and displays the updated information.
   * **Sign in/Sign out:**
     + **Sign in**: Allows users to sign in to their accounts.
       1. How to Use: Users enter their email and password on the sign in page and click the "Sign in" button to access their account.
       2. If a user attempts to sign in with invalid information 5 times, they will be blocked from signing in for 10 minutes.
     + **Sign out**: Allows users to log out of their accounts.
       1. How to Use: Users can sign out by clicking the "Sign Out" button located in the navigation bar.
     + Links to Other Functions: Signing in and out are essential for accessing secured areas of the website and interacting with personalized content. These functions are also linked to the security features for password hashing and session management.
2. **Security Functionality**:
   * **Data Encryption**: Encrypts sensitive data to protect it from unauthorized access.
     + How it Works: Data encryption is applied to sensitive user information such as passwords, ensuring that even if the data is intercepted, it remains unreadable without the decryption key.
   * **Password Hashing**: Converts user passwords into irreversible hashes for secure storage.
     + How it Works: Passwords are hashed using a secure hashing algorithm before being stored in the database. This ensures that even if the database is compromised, plaintext passwords cannot be retrieved.
       1. Both data encryption and password hashing are linked to the accounts functionality by ensuring that sensitive user information, such as passwords, is securely stored and managed, enhancing overall account security.
   * **Input Sanitization**: Cleans and validates user input to prevent common security vulnerabilities.
     + How it Works: Cleans and validates user input to prevent common security vulnerabilities such as NoSQL injection and protection against malicious HTML input.
     + Linked to various functionalities across the application. This prevents common security vulnerabilities, ensuring that user input data is clean and safe to process. This also indirectly supports features like profile editing.
   * **Rate Limiting**: Limits the number of requests a user can make within a certain timeframe to prevent abuse.
     + How it Works: Rate limiting tracks user requests and blocks requests that exceed the defined limits.
     + Linked to user authentication and API endpoints, rate limiting prevents abuse and ensures fair usage of system resources, maintaining system stability and preventing potential attacks on authentication mechanisms and backend APIs.
   * How External Resources Are Incorporated: External resources such as security libraries or functions (validator library, bcrypt for password hashing, and rate limiting functions) are integrated into the backend codebase to enhance the security of the application.
3. **Sidebar Functionality**:
   * **Navigate Website**: Provides navigation links to different pages of the website.
     + How to Use: Users can click on the sidebar links to navigate to specific sections such as hubs, profile, sign out, and email signup.
   * **Display Hubs**: Shows a list of hubs or communities within the platform.
     + How to Use: Users can view the list of hubs in the sidebar and click on a hub to access its details. They can also join a hub using a code.
   * Links to Other Functions: The sidebar functionality is linked to various sections of the website and enhances user navigation and accessibility.
4. **Notifications Functionality**:
   * **Email List Signup**: Allows users to subscribe to email notifications from our app.
     + How to Use: Users can sign up for email notifications by filling out the email signup form on the website.
     + Links to Other Functions: Email list signup integrates with SendGrid to manage and send email notifications to subscribers.
5. **Hubs Functionality**:
   * **Display Members and Hub Info**: Shows information about hub members and details about each hub.
     + How to Use: Users can navigate to the "Hubs" section and view the list of members and hub information.
   * **Links to Subpages**: Provides links to subpages or specific sections within each hub.
     + How to Use: Users can click on the subpage links in a hub to access related content or features.
   * **Admin Page**: Enables hub administrators to manage hub settings and members.
     + How to Use: Admins can access the admin page from the hub interface and perform actions like admitting/denying member requests, kicking/banning members, and updating hub information.
   * Links to Other Functions: The hubs functionality interacts with user accounts, security features, and notifications (e.g., sending notifications to hub members).
6. **Resources/Designer Pages Functionality**:
   * **File Uploads/Downloads**: Allows users to upload and download files/resources.
     + How to Use: Users can upload files to share with others or download resources provided by the platform.
   * **Scripts**: Manages script uploads and displays script PDFs for users.
     + How to Use: Users can upload and download script files or view existing script PDFs on the scripts page.
     + Scripts can be edited by drawing/highlighting on the document and can be downloaded with the edits included.
   * Links to Other Functions: This functionality interacts with file storage systems, user accounts, and notifications [in the future].
7. **Unit Testing Functionality**:
   * **Automated Tests for Files, Accounts, etc.**: Performs automated tests to ensure the functionality and integrity of files, user accounts, and other system components.
     + How to Use: Developers can run automated tests to check for errors, bugs, or vulnerabilities in different parts of the system.
   * Links to Other Functions: Unit testing is a development and quality assurance practice that supports overall system stability and reliability.

# How external resources are integrated

1. **React (Frontend Library):**
   * Used for building user interfaces.
   * Incorporated through component-based architecture for a modular and scalable frontend.
2. **Axios (HTTP Client for Node.js):**
   * Used for making HTTP requests to the backend API.
   * Incorporated into frontend components to fetch and send data from/to the server.
3. **SendGrid (Email API):**
   * Integrated for sending marketing emails, notifications, and password reset emails (to be implemented).
   * Configured with API keys stored securely in environment variables for email functionality.
4. **MongoDB (Database):**
   * Used for storing user profiles, account information, and other application data.
   * Integrated into the backend to handle CRUD operations and ensure data persistence and reliability.