

# User Management System - Process Flow Documentation

## Introduction

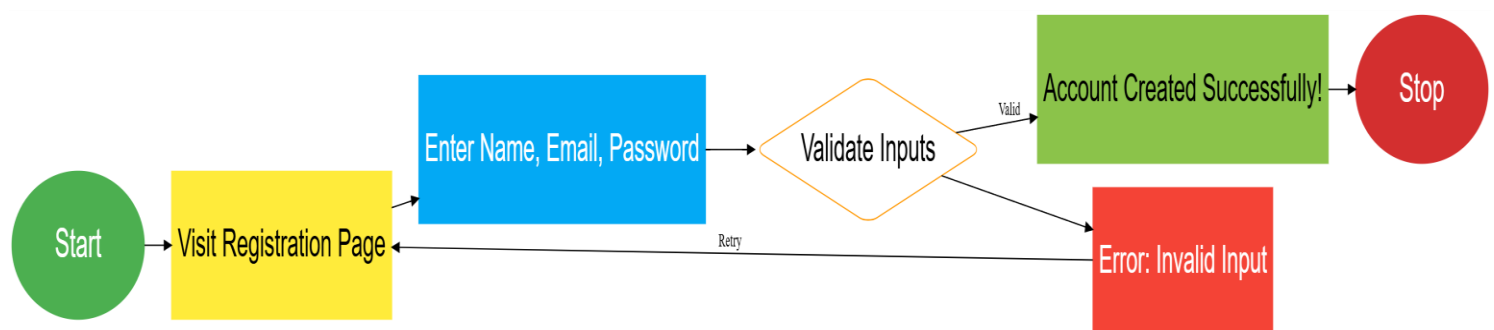
The User Management System is designed to manage user accounts efficiently by providing a seamless registration, authentication, and profile management experience. This system enables users to register by providing essential information such as personal details, educational background, and employment history. It includes a secure login mechanism with token-based authentication to ensure authorized access, along with functionalities for password recovery, profile viewing, and updates. The application also allows users to change their password, ensuring improved account security, and supports a logout feature to end sessions securely. This report documents the end-to-end process flow of the system, from user registration to login, profile management, and session control, including flowcharts to illustrate each operation's workflow.

## User Story 1: Basic Registration

As a new user, I want to register for an account by providing my name, email, and password, so that I can create a basic account and log in to access further features.

### Acceptance Criteria:

- User visits the registration page.
- User inputs the required details: Basic Information: Name, Email, Password.
- System validates the inputs: Ensures email is unique and properly formatted. Password meets security requirements.
- If valid: The system creates a new user account with only basic details.
- User receives a success message: 'Account created successfully!'
- If invalid: The system shows error messages for fields that failed validation (e.g., 'Email already exists').

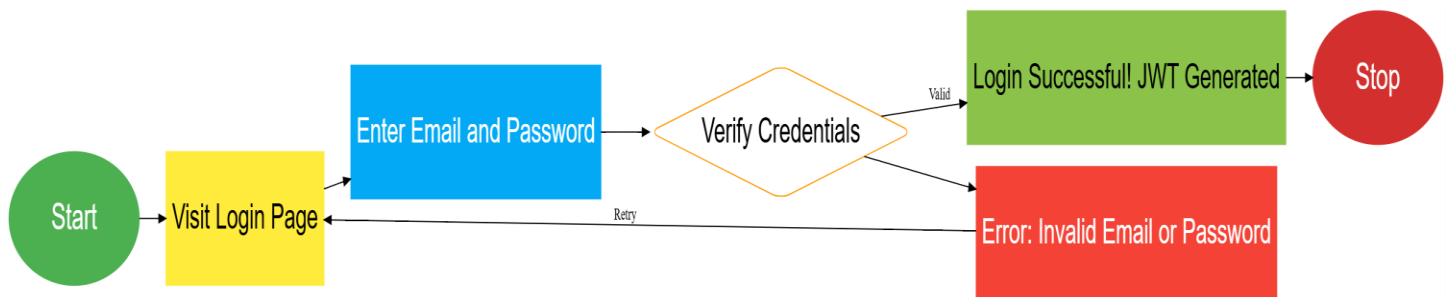


## User Story 2: Login and Token Generation

As a registered user, I want to log in with my email and password, so that I can receive an authentication token to securely access my data.

### Acceptance Criteria:

- User visits the login page.
- User inputs their email and password.
- System verifies the credentials: Checks that the email exists in the system. Checks that the password matches the stored hash for that user.
- If valid: System generates a JWT (JSON Web Token) for the user, which includes their User ID.
- User receives the token and is logged in, with a success message: 'Login successful!'
- If invalid: System shows an error message: 'Invalid email or password.'

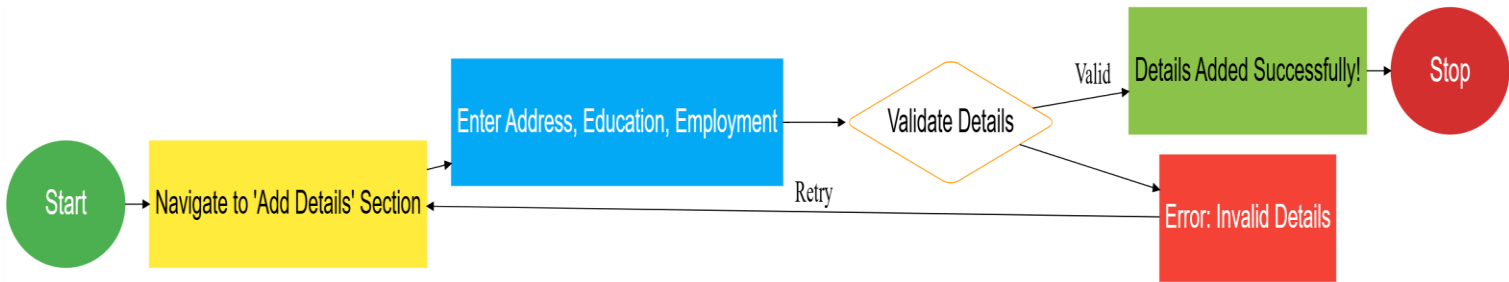


## User Story 3: Add Additional Details (Authenticated)

As an authenticated user, I want to add additional information like address, education, and employment details, so that I can have a complete profile in the system.

### Acceptance Criteria:

- User has logged in and received an authentication token.
- User navigates to the 'Add Details' section for address, education, or employment.
- User provides Address, Education, and Employment details.
- The token is used to retrieve the User ID, linking the details to the correct user.
- The system validates each section's input for required fields.
- If valid: System saves the details for the authenticated user.
- User receives a success message: 'Details added successfully!'
- If invalid: System shows appropriate error messages for the fields that failed validation.

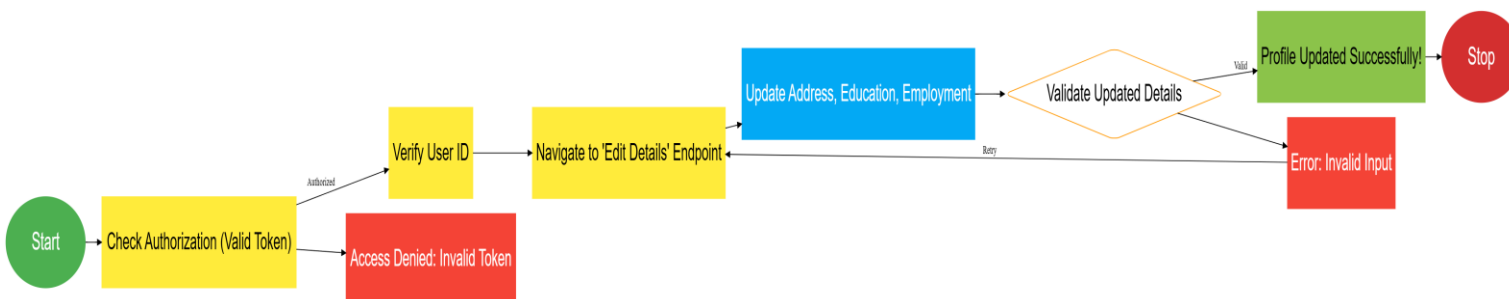


#### User Story 4: Update Profile Details (Authenticated)

As an authenticated user, I want to update my address, education, and employment details, so that I can keep my profile information up to date.

##### Acceptance Criteria:

- User is logged in with a valid authentication token.
- User navigates to the 'Endpoint ' for address, education, or employment.
- User updates relevant fields for Address, Education, or Employment.
- The token is used to retrieve the User ID, ensuring updates apply only to the authenticated user's records.
- The system validates the updated input.
- If valid: System saves the updated details for the authenticated user.
- User receives a success message: 'Profile updated successfully!'
- If invalid: System shows appropriate error messages for fields with incorrect input.

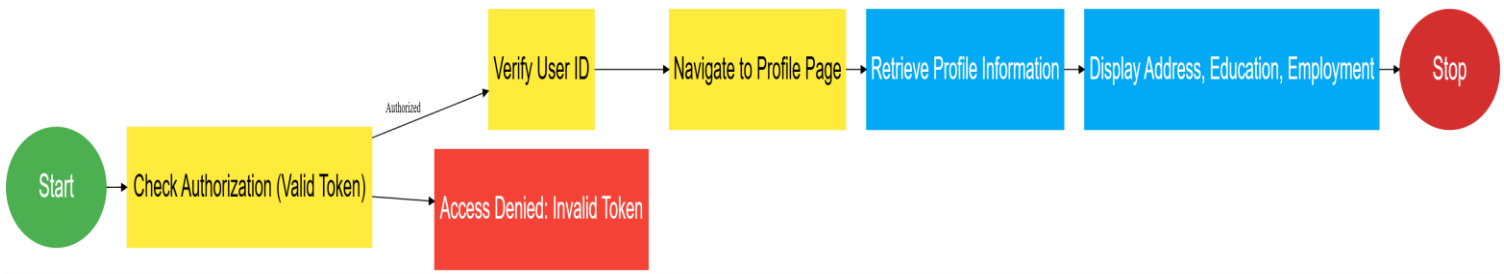


#### User Story 5: View Profile (Authenticated)

As an authenticated user, I want to view my complete profile, including my address, education, and employment details, so that I can see my information in one place.

##### Acceptance Criteria:

- User is logged in with a valid authentication token.
- User navigates to the profile page.
- System uses the token to identify the User ID.
- System retrieves and displays all profile details, excluding sensitive information like the password.
- User can see all their details in a single view.

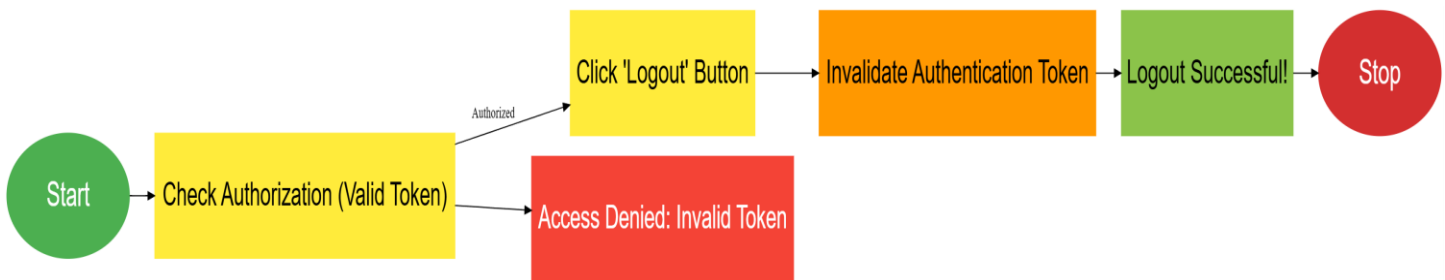


### User Story 6: Logout

As an authenticated user, I want to log out of my account, so that my session is terminated, and I can ensure security.

#### Acceptance Criteria:

- User clicks on the 'Logout' button.
- System invalidates the current authentication token.
- User receives a confirmation message: 'Logged out successfully.'
- Further access to protected resources requires the user to log in again.

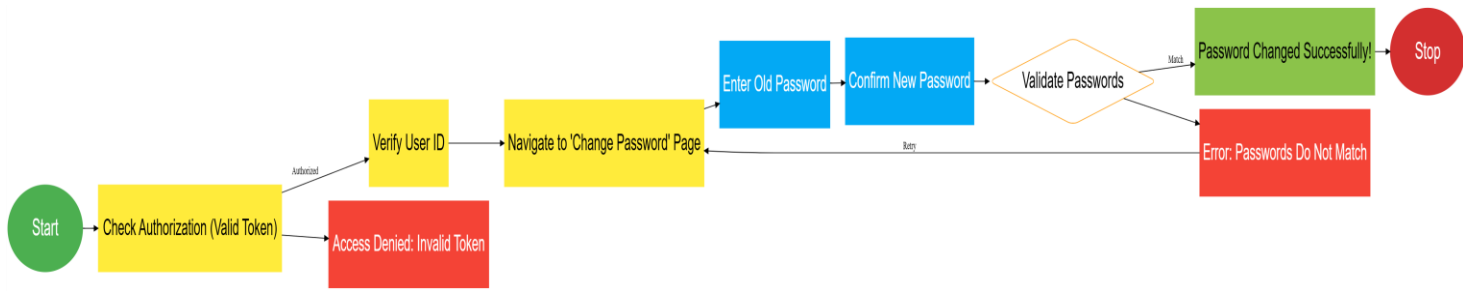


### User Story 7: Change Password (Authenticated)

As an authenticated user, I want to change my account password, so that I can keep my account secure.

#### Acceptance Criteria:

- User is logged in with a valid authentication token.
- User navigates to the 'Change Password' page.
- User inputs current and new passwords.
- System verifies the current password and validates the new password.
- If valid: System updates the password and shows a success message.
- If invalid: System shows an error message.



## User Story 8: Authorization for All Protected Endpoints

As an application developer, I want to ensure that only authenticated users with valid tokens can access or modify sensitive data, so that unauthorized access to the system is prevented.

### Acceptance Criteria:

- All endpoints that allow viewing or updating user data are protected.
- Unauthorized access returns an error message.
- If the token is expired or invalid, the user is denied access to protected routes.

