**Name: Roma Shirodkar**

**Div: D15B**

**Roll No: 54**

**Practical 5**

**Aim:** To apply navigation, routing and gestures in Flutter App

**Steps:**

**Set Up Pages:**

* Ensure you have a **Login Page** and a **Home Screen** as separate widgets.

**Use Navigator Class for Navigation:**

* Flutter uses the Navigator widget to manage a stack of screens (routes).
* You can push, replace, or pop screens using the Navigator.

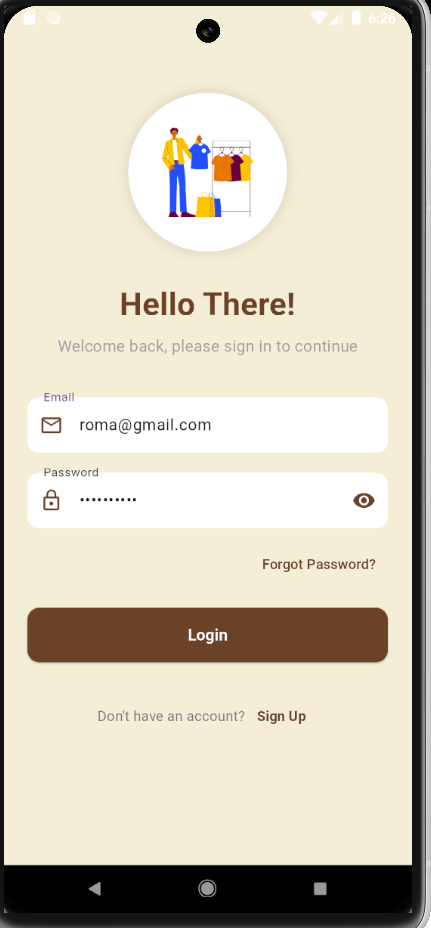
**Navigation Methods:**

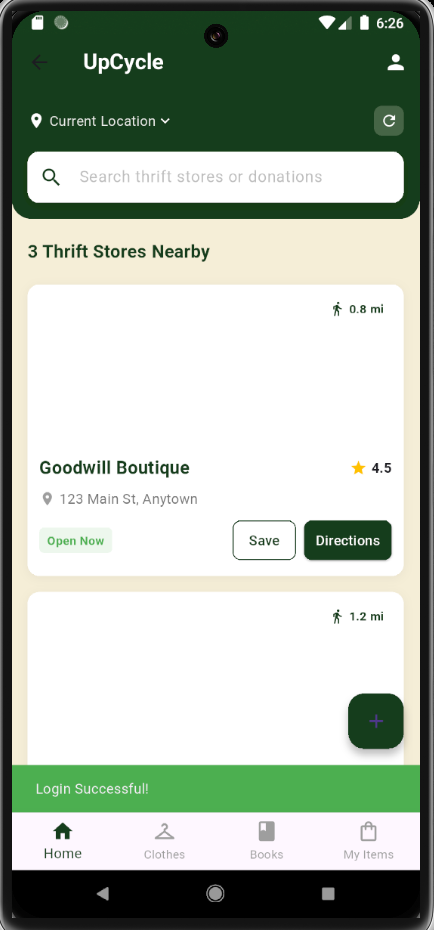
* **Navigator.push()**: Adds a new screen on top of the current one, allowing users to go back.
* **Navigator.pushReplacement()**: Replaces the current screen with a new one, preventing the user from navigating back.
* **Navigator.pop()**: Removes the current screen and returns to the previous one.

**Navigate on User Action:**

* Trigger navigation when a user interacts (e.g., after clicking a "Login" button).
* Use Navigator.pushReplacement() for login flows to prevent returning to the login page.

Output:





Gestures

**Wrap the Card with GestureDetector**

* Use Flutter’s GestureDetector widget to capture user gestures.
* Wrap the thrift store card widget inside the GestureDetector to detect interactions.

**Detect Double Tap Gesture**

* Set the onDoubleTap property of GestureDetector to trigger an action when the user double-taps.

**Navigate to the Detail Page**

* Use Navigator.push() to navigate to the thrift store's detail page.
* Pass relevant information (like store details) via the constructor or arguments.

**Manage Data Passing (Optional)**

* If each card represents a unique thrift store, pass its details (name, image, etc.) to the detail page.

