Retail-Hub-App Data Flow

1. Introduction

This document provides a detailed Data Flow Diagram (DFD) representation of the Retail-Hub-App React application.

It explains how user registration, form validation, API interaction, and state management are handled dynamically.

2. System Overview

The React application consists of the following key components:

- App.js: The main file that sets up React Router and renders the SignUpForm component.
- SignUpForm.js: A controlled form that handles user registration with validation and API interaction.
- useForm Hook: Manages form state and validation using React Hook Form and Yup schema.
- Axios: Handles API requests to send registration data to the backend.

3. Data Flow Diagram (DFD)

Level 0 (Context Diagram)

At the highest level, the system consists of external users interacting with the sign-up form.

External Entities:

- User: Provides registration details (username, email, password, confirm password).

Processes:

- React Application: Handles form validation, state updates, and API requests.

Data Stores:

- Backend API: Stores registered user information.

```
   +--------------------+
   |     User          |
   +--------------------+
          |
          v
   +--------------------+
   |  React App (UI)   |
   +--------------------+
          |
          v
   +--------------------+
   |  Backend API      |
   +--------------------+
```

Level 1 DFD (User Registration Process)

This level breaks down how user registration is handled from input to API interaction.

```
+--------------------+       +--------------------+
|  User Enters Data | -----> |  Validate Form     |
+--------------------+       +--------------------+
        |                          |
        v                          v
+--------------------+       +--------------------+
|  Show Errors      | -----> |  Submit to API     |
|  if Any           |        |  if Validation Pass |
+--------------------+       +--------------------+
```

Process Explanation:

1. The user enters registration details in the form fields.
2. React Hook Form validates inputs against the Yup schema.
3. If validation fails, error messages are displayed.
4. If validation passes, the data is sent to the backend API via Axios.

Level 2 DFD (API Interaction)

```
+----------------------+       +----------------------+
|  Send Data to API   | -----> |  Backend Processes  |
+----------------------+       +----------------------+
          |                              |
          v                              v
+----------------------+       +----------------------+
|  Receive Response   | -----> |  Show Success/Error  |
+----------------------+       +----------------------+
```

Process Explanation:

1. The form data is sent to the backend API using Axios.
2. The backend processes the request and returns a response.
3. Based on the response, the UI displays success or error messages.

4. Explanation of Data Flow

1. The user interacts with the form by entering registration details.
2. React Hook Form manages input validation and state updates.
3. Axios sends the validated data to the backend for processing.
4. The application updates the UI based on the API response.

5. Conclusion

This document outlines the detailed data flow in a React-based user registration system.

The application ensures data validation, dynamic UI updates, and smooth API interaction.