# **Data Flow Diagram (DFD) for Dynamic Pages**

## **Level 0 DFD (Context Diagram)**

### **Diagram:**

+----------------------+

| External Entity |

| (User) |

+----------------------+

|

v

+----------------------+

| React App (SPA) |

| (Process: 1.0) |

+----------------------+

|

v

+----------------------+

| Data Store (Local) |

+----------------------+

### **Explanation:**

* **User** interacts with the application by clicking navigation links or searching for pages.
* **React App (SPA)** processes these requests and loads the corresponding page dynamically.
* **Data Store (Local Storage / Predefined Data)** stores static posts and navigational details.

## **Level 1 DFD (Breaking Down Process 1.0)**

### **Diagram:**

+----------------------+

| External Entity |

| (User) |

+----------------------+

|

v

+---------------------------------+

| Process: 1.1 - User Navigation |

| (Navbar & Routes) |

+---------------------------------+

|

v

+---------------------------------+

| Process: 1.2 - Search Handling |

| (Redirects to Routes) |

+---------------------------------+

|

v

+---------------------------------+

| Process: 1.3 - Post Management |

| (Fetch & Display Posts) |

+---------------------------------+

|

v

+----------------------+

| Data Store (Local) |

+----------------------+

### **Explanation of Processes:**

1. **Process 1.1 - User Navigation**
   * The **Navbar component** allows users to navigate to:
     + Home /
     + About /about
     + Contact /contact
     + Specific Post /post/:id
     + If an invalid URL is entered, they are redirected to the **404 page**.
2. **Process 1.2 - Search Handling**
   * Users enter search terms in the search bar.
   * If the term matches **home, about, contact, or post {id}**, the app redirects to that page.
   * If the term does not match any valid page, the app **redirects to the 404 page**.
3. **Process 1.3 - Post Management**
   * If a user searches for **post {id}**, it checks if the post exists in the stored posts.
   * If found, it redirects the user to /post/{id} and displays the content.
   * If not found, it redirects to the **404 page**.

## **Level 2 DFD (Detailed Breakdown of Post Management Process 1.3)**

### **Diagram:**

+----------------------+

| User Searches Post |

+----------------------+

|

v

+--------------------------------+

| Check if Post Exists Locally |

+--------------------------------+

| |

| Yes | No

v v

+----------------+ +----------------+

| Display Post | | Redirect to |

| Content | | 404 Page |

+----------------+ +----------------+

### **Explanation:**

1. **User Searches for a Post**
   * The user types post {id} in the search bar.
2. **Check if Post Exists Locally**
   * The app searches for the post in the predefined storedPosts list.
3. **If the Post Exists**
   * The post content is displayed.
4. **If the Post Does Not Exist**
   * The user is redirected to the **404 Page**.

## **Summary**

The user navigates using the **navbar** or **search bar**.  
 The app **redirects** based on valid or invalid searches.  
 **Stored posts** are used for displaying content.  
 If a **post doesn’t exist**, the user is sent to the **404 page**.