**Data Flow Diagram (DFD) for React Counter App**

**Project Description**

The React Counter App is a simple web application that allows users to click a button to increase a counter value. The system processes user input, updates the counter state, and dynamically refreshes the UI.

**Level 0 DFD (Context Diagram)**

The Level 0 DFD provides a high-level overview of the system, illustrating how the User interacts with the React Counter App and how data flows between the system and its state management.

+-----------------------+  
| External Entities |  
| |  
| - User |  
+-----------------------+  
 |  
 v  
+---------------------------+  
| React Counter App |  
| (Process: 1.0) |  
+---------------------------+  
 |  
 v  
+---------------------------+  
| Data Store: State |  
+---------------------------+

**Explanation:**

• **External Entity (User):** The user interacts with the React Counter App by clicking a button.

• **Process (React Counter App):** The system processes the button click and updates the counter value.

• **Data Store (State):**The application state stores and manages the counter value.

**Level 1 DFD (Detailed Breakdown)**

The Level 1 DFD further decomposes the React Counter App process into sub-processes, showing how user interactions trigger state updates and UI re-rendering.

+-----------------------+  
| External Entity |  
| (User) |  
+-----------------------+  
 |  
 v  
+----------------------------+  
| Process: 1.1 - Click Button |  
| (Trigger Event) |  
+----------------------------+  
 |  
 v  
+----------------------------+  
| Process: 1.2 - Update State |  
| (useState Hook) |  
+----------------------------+  
 |  
 v  
+----------------------------+  
| Process: 1.3 - Re-render UI |  
| (React DOM Updates) |  
+----------------------------+  
 |  
 v  
+----------------------------+  
| Data Store: State |  
| (Updated Counter Value) |  
+----------------------------+

**Explanation:**

1**. Process 1.1 (Click Button - Trigger Event):**The user clicks the button, triggering an event.

2**. Process 1.2 (Update State – use State Hook):**The application updates the counter value using React’s state management.

3. **Process 1.3 (Re-render UI - React DOM Updates):** The UI dynamically updates to reflect the new counter value.

**Data Flow**

• The User clicks the button.

• The React Counter App captures the click event.

• The useState Hook updates the State with the new counter value.

• React re-renders the UI to reflect the updated count.