**Level 0 (Context Diagram)**

At the highest level, the system consists of a single process that represents the Form Handling App.

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

| External Entity |

| (User) |

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

|

v

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

| Form Handling App |

| (Process: 1.0) |

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

|

v

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

| Data Store: Form |

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

**Explanation:**

* **External Entity (User):** The user interacts with the **Form Handling App** by entering their name and email.
* **Process (Form Handling App):** The system processes the user's input and submits the form.
* **Data Store (Form):** Stores the user's entered information.

**Level 1 DFD (Decomposition of Process 1.0 - Form Handling App)**

Now, let’s break down the **Form Handling App** process into more detailed steps.

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

| External Entity |

| (User) |

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

|

v

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

| Process: 1.1 - Input Name & Email |

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

|

v

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

| Data Store: Form State |

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

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

| Process: 1.2 - Submit Form |

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

|

v

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

| Process: 1.3 - Display Submitted Data |

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

**Explanation:**

1. **Process 1.1 (Input Name & Email):**
   * The user enters their **name** and **email** into the input fields.
   * The **useState** hook updates the state dynamically as the user types.
2. **Process 1.2 (Submit Form):**
   * When the user clicks the **"Submit"** button, the **handle Submit** function is triggered.
   * The page prevents default form submission (e.preventDefault()) and updates the submitted state.
3. **Process 1.3 (Display Submitted Data):**
   * If the form is submitted, the system replaces the input fields with a confirmation message.
   * Displays the user’s entered **name** and **email**.

**Data Flow**

* The **User** interacts with the form by entering their details.
* The **React State** stores the input dynamically.
* Upon submission, the system updates the UI to display the submitted details.

**Additional Notes:**

* This **React-based form** handles input state using use State.
* **Conditional rendering** ensures that once submitted, the form fields disappear, and the submitted data is shown.
* Future enhancements could include **form validation, API integration, and error handling**.