DFD for StopWatch

Data Flow Diagram (DFD) for Stopwatch Application

A **Data Flow Diagram (DFD)** represents how data flows within the **Stopwatch Application**, showing interactions between the **User, Stopwatch System, and Data Storage** (localStorage for dark mode settings).

Level 0 DFD (Context Diagram)

At the highest level, the Stopwatch App consists of a **single process** representing the entire system. The **User** interacts with the app by **starting, stopping, and resetting the stopwatch** or **toggling dark mode**.

Diagram Structure

┌───────────┐ ┌───────────—--

│ User │──────▶ │ Stopwatch System│

│ │ │ (Process: 1.0) │

│ | ◀—──── │ Display Time │

│ │ | Store Dark Mode |

└───────────┘ └────────────┘

Explanation

* **External Entity: User** → The user interacts with the system to **start, stop, reset the stopwatch, and toggle dark mode**.
* **Process (Stopwatch System)** → Handles **timer logic and dark mode toggle**.
* **Data Store (localStorage)** → Saves **dark mode preferences** for future sessions.

Level 1 DFD (Decomposition of Stopwatch System)

Now, let's break down the **Stopwatch System (Process 1.0)** into detailed sub-processes.

Diagram Structure

┌───────────┐ ┌──────────────┐

│ User │────Start/Stop────▶ │ Process: 1.1 │

│ │ │ Manage Stopwatch │

│ │◀───Display Time─── │ (useState, useEffect) │

│ │ └───────────────┘

│ │ ┌────────────-──┐

│ │──Toggle Dark Mode─▶ │ Process: 1.2 │

│ │ │ Handle Dark Mode │

│ │◀──Apply Dark Mode──│ (localStorage) │

│ │ └──────────────┘

└───────────┘

Explanation

Process 1.1 - Manage Stopwatch

* The **user clicks Start** → The timer **increments every second** using setInterval().
* The **user clicks Stop** → The timer **pauses**.
* The **user clicks Reset** → The timer **resets to 0**.
* The **system updates the display** to show the current stopwatch time.

Process 1.2 - Handle Dark Mode

* The **user clicks the Dark Mode button** → The system toggles dark mode.
* The system **stores the dark mode setting in localStorage**.
* On the next visit, the app **retrieves and applies** the dark mode preference.

Data Flow Explanation

1. The **User** interacts with buttons (Start, Stop, Reset, Dark Mode).
2. The **Stopwatch System** handles these actions:
   * **Updates and displays time** (Process 1.1).
   * **Toggles dark mode and stores preference** (Process 1.2).
3. The **updated stopwatch time** is displayed in the UI.
4. The **dark mode preference** is stored and applied on the next visit.

Retail-app

Level 0 (Context Diagram)

At the highest level, the system allows users to sign up, log in, and access a success page.

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

| User |

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

|

v

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

| Authentication System |

| (Process: 1.0) |

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

|

v

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

| Local Storage (User DB) |

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

Explanation:

1. The **User** interacts with the **Authentication System**.
2. The **Authentication System** processes the signup and login requests.
3. The system stores user data in **Local Storage** (acting as a temporary user database).
4. Upon successful login, the user is redirected to the **Success Page**.

Level 1 (Detailed DFD Breakdown)

Now, let's break down the **Authentication System** (Process 1.0) into individual sub-processes.

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

| User | | User |

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

| |

v v

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

| Process 1.1: Signup | | Process 1.2: Login |

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

| |

v v

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

| Local Storage (DB) | | Local Storage (DB) |

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

| |

v v

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

| Redirect to Home Page| | Redirect to Success |

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

Explanation of Sub-Processes:

1. **Process 1.1 - Signup:**
   * The user enters their details in the signup form.
   * The data is validated and stored in **Local Storage**.
   * The user is redirected to the **Home Page**.
2. **Process 1.2 - Login:**
   * The user enters login credentials.
   * The system retrieves stored data from **Local Storage** and verifies the credentials.
   * If valid, the user is redirected to the **Success Page**; otherwise, an error message is displayed.