**Experiment No. 6: Draw Data Flow Diagram for Your Project**

### **Aim:**

To design a Data Flow Diagram (DFD) for the project, representing the flow of information and processes involved in the system.

### **Theory:**

A Data Flow Diagram (DFD) is a graphical representation of data flow within a system. It illustrates how data moves between processes, data stores, and external entities (sources/sinks) in a system. DFDs help in understanding the system's functionality and identifying bottlenecks or inefficiencies.

**Key Components of DFD:**

1. **Processes:** Represented as circles or ovals, they depict actions performed on data.
2. **Data Stores:** Represented as open rectangles, they show where data is stored within the system.
3. **External Entities:** Represented as squares, they denote sources or sinks that interact with the system.
4. **Data Flows:** Represented as arrows, they show the flow of data between entities, processes, and data stores.

**Levels of DFD:**

* **Level 0 (Context Diagram):** Provides an overview of the system with a single process.
* **Level 1:** Breaks down the main process into sub-processes.
* **Level 2 and Beyond:** Further decomposes sub-processes into detailed steps.

### **Learning Objective:**

* To understand the role of Data Flow Diagrams in system design and analysis.
* To learn to represent processes, data stores, and data flows effectively.
* To develop analytical skills for visualizing the flow of information in the system.

### **Learning Outcome:**

At the end of this experiment, students will be able to:

1. Analyze the data flow in their project.
2. Develop Level 0, Level 1, and, if needed, Level 2 DFDs for the project.
3. Demonstrate the ability to use DFDs for identifying system functionality and data dependencies.

### **Course Outcomes (COs):**

* **CO3:** Identify and formulate functional and non-functional requirements for software systems.
* **CO5:** Evaluate and mitigate software project risks and apply configuration management practices to maintain project integrity.

### **Cognitive Levels of Attainment as per Bloom’s Taxonomy**

* **L3 (Apply):** Apply knowledge to create DFDs for the project.
* **L5 (Evaluate):** Evaluate the data flow and ensure its correctness and completeness.

### **Programme Outcome (POs):**

* **PO2: Problem Analysis:** Analyze complex system requirements to identify data flow and processes.
* **PO3: Design/Development of Solutions:** Design DFDs to represent the system’s processes and data interactions.
* **PO5: Engineering Tool Usage:** Use modern tools to create accurate DFDs.
* **PO10: Project Management and Finance:** Plan and document system processes for better project management.

### **Programme Specific Outcome (PSOs):**

* **PSO1:** Apply software engineering principles to design sustainable IT solutions for system data flow.
* **PSO2:** Integrate data flow representation into effective system modeling and real-time problem-solving.

### **Result & Discussion:**

* **Result:** Designed a Level 0 and Level 1 DFD for the project.
* **Discussion:**
  + The DFD successfully depicted the data flow and system processes.
  + Level 1 DFD provided detailed insights into sub-processes, aiding in system analysis.

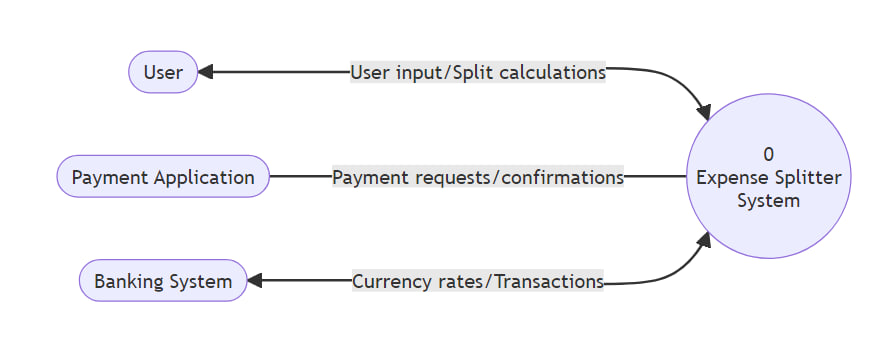
### **Conclusion:**

Drawing Data Flow Diagrams is a crucial step in understanding and designing system processes. Through this experiment, students gained the ability to represent system functionality and data flow visually, ensuring clarity in system design and development.

### **Steps for the Experiment**

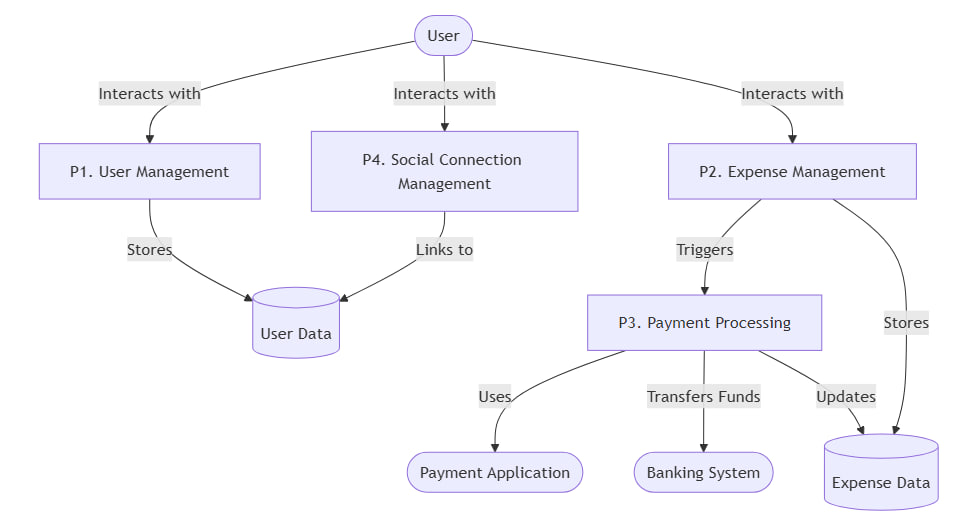
#### ****1. Level 0: Context Diagram Development****

1. Identify the main system as a single process.
2. List external entities interacting with the system.
3. Identify data flows between the system and external entities.
4. Create a context diagram showing the overall data flow.



#### ****2. Level 1: DFD Development****

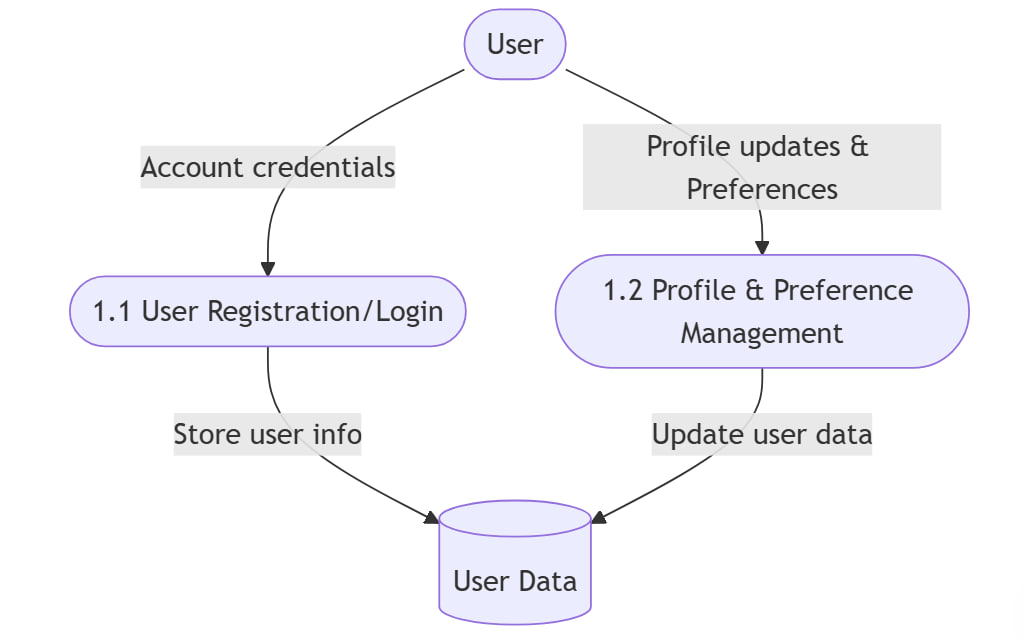
1. Decompose the main process into sub-processes.
2. Identify data stores used by the system.
3. Map data flows between sub-processes, data stores, and external entities.
4. Ensure completeness and correctness in data representation.



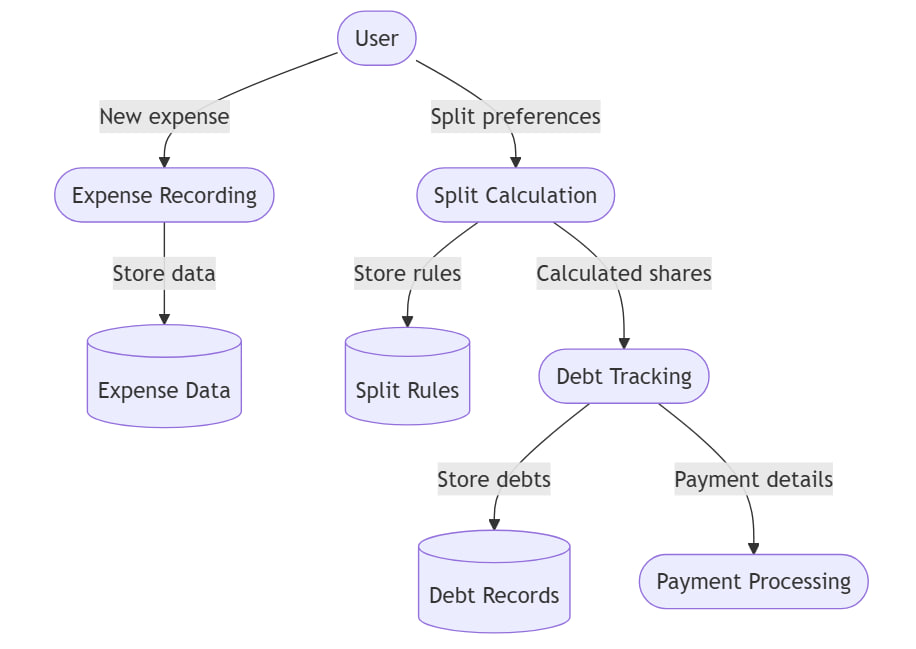
#### ****3. Optional: Level 2 and Beyond****

1. Further break down sub-processes from Level 1 into smaller steps.
2. Refine data flows and interactions for more granularity.

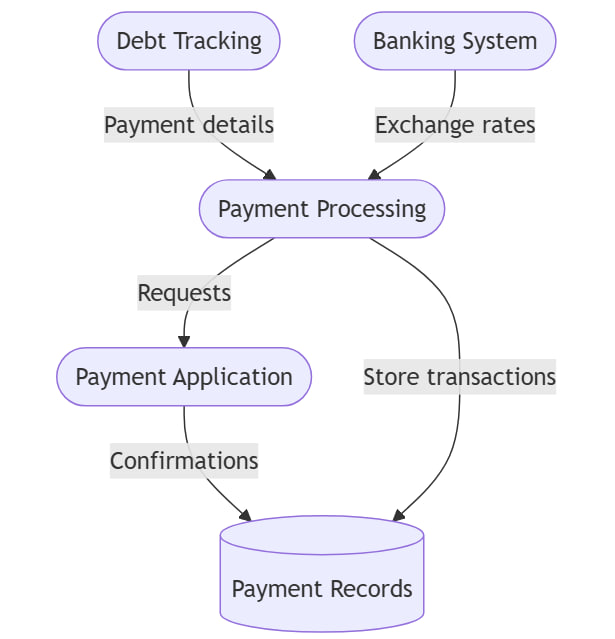
**P1. User Management**



**P2. Expense Management**



**P3. Payment Processing**



**P4. Social Connection Management**

