|  |  |
| --- | --- |
| Date | 29-05-2025 |
| Team ID | LTVIP2025TMID28821 |
| Project Name | TO SUPPLY LEFTOVER FOOD TO POOR |
| Maximum Marks | 4 Marks |

**Data Flow Diagram (DFD)**

**Data Flow Diagram (DFD)**

**1. Introduction to Data Flow Diagrams**

A Data Flow Diagram (DFD) is a graphical representation of the flow of data through a system. It illustrates how data enters, is processed, stored, and exits the system, making it easier to understand system behavior, user interaction, and system boundaries.

For the *“To Supply Leftover Food to the Poor”* initiative, the DFD plays a vital role in modeling the movement of information between the key actors—**Food Donors**, **NGOs**, **Volunteers**, and **System Administrators**—and the centralized platform (*NourishBridge*). This structured view supports better system design, ensures data integrity, and guides future development.

**2. Objectives of the DFD**

The objectives of creating the DFD for this project are:

* To visualize how data flows between users and the system.
* To define system boundaries and identify external entities.
* To support developers in understanding processing steps.
* To ensure clarity in data handling, storage, and retrieval.
* To aid scalability and troubleshooting of the application.

**3. Key System Entities and Data Stores**

Before diving into diagram levels, we define the main actors and data storage components:

**External Entities**

* **Food Donor**: Restaurant or individual posting surplus food.
* **NGO Coordinator**: Charitable organization accepting and distributing food.
* **Volunteer**: Person assigned to pick up and deliver the food.
* **System Administrator**: Oversees user accounts and platform health.

**Major Data Stores**

* **User Profiles**: Information on Donors, NGOs, and Volunteers.
* **Donation Records**: Each posted donation and its lifecycle status.
* **Pickup Logs**: Collection, transit, and delivery confirmations.
* **Impact Reports**: Analytical data used for dashboards and reports.

**4. Level 0 DFD – Context Diagram**

This diagram represents the entire system as a single process. All external entities interact with the system boundary, showing only data inputs and outputs.

**Processes & Data Flow (High-Level Overview)**:

* **Food Donor ➝ System**: Donation details, account info.
* **System ➝ Donor**: Confirmation, status updates, reports.
* **NGO ➝ System**: Accept donation, update delivery status.
* **System ➝ NGO**: Alerts for available food, donation details.
* **Volunteer ➝ System**: Pickup status, delivery logs.
* **Admin ➝ System**: Approval and monitoring commands.
* **System ➝ Admin**: User requests, platform health reports.

This level abstracts internal processes to keep the focus on **system boundaries and data exchange**.

**5. Level 1 DFD – Functional Breakdown**

At this level, the internal processes of the system are revealed to show how data is manipulated and transferred among actors.

**Key Processes**:

* **1.0 User Registration & Profile Management**
  + Inputs: Registration forms, verification documents.
  + Outputs: Verified profiles, user access rights.
* **2.0 Donation Management**
  + Inputs: Posted donation (food type, quantity, address, time).
  + Processes: Store, match, notify NGOs.
  + Outputs: Confirmed donations, updates to donors.
* **3.0 Matching & Notification System**
  + Inputs: New donation entries.
  + Processes: Match algorithm checks distance and NGO capacity.
  + Outputs: SMS and email notifications.
* **4.0 Volunteer Coordination**
  + Inputs: Accepted donation by NGO.
  + Processes: Assign volunteer, track pickup and delivery.
  + Outputs: Pickup logs, real-time updates.
* **5.0 Admin Oversight & Reporting**
* Inputs: Data from all modules.
* Processes: Generate dashboards, approve users, audit activity.
* Outputs: Reports, impact summaries.

**6. Sample Data Flow Descriptions**

Here are a few sample flows to show how data travels between components:

* **Flow A**: Donor submits donation ➝ System records donation ➝ Matching engine notifies NGOs ➝ NGO accepts ➝ Volunteer assigned ➝ System updates all parties.
* **Flow B**: NGO completes delivery ➝ Updates system ➝ System logs delivery ➝ Dashboard updates donor impact metrics.
* **Flow C**: Admin logs in ➝ Reviews pending NGO registrations ➝ Verifies documents ➝ Approves access ➝ Updates user profile in database.

**7. Security & Validation Considerations**

To ensure data integrity:

* All submissions are validated (e.g., food donation entries must include pickup window).
* Only verified users can access dashboards or take actions.
* Pickup and delivery steps require volunteer authentication and timestamped confirmations.

**8. Benefits of the DFD Approach**

Implementing a DFD structure offers multiple operational benefits:

* Promotes clarity during development and deployment.
* Ensures each module aligns with specific data objectives.
* Makes onboarding of developers and testers more efficient.
* Enables easier monitoring and troubleshooting.
* Supports scalability by visually modeling potential bottlenecks.

**9. Conclusion**

The Data Flow Diagram for *“To Supply Leftover Food to the Poor”* acts as a visual framework that ties together the project's mission, user needs, and system functionalities. By clearly defining how information moves across various modules and actors, the DFD ensures transparency, efficiency, and accountability at every stage of the platform’s operation. As the platform scales, the DFD will also evolve—serving as a blueprint for continuous improvement and innovation.