**To Supply Leftover Food to the Poor**

**1. Project Overview**

This project aims to develop a system that connects food donors (restaurants, caterers, event organizers, individuals) with volunteers and NGOs to distribute leftover food to the poor and needy. The platform will streamline the process of collecting, storing, and distributing food safely and efficiently, reducing food wastage and supporting hunger alleviation efforts.

**2. Project Objectives**

* To reduce food wastage by utilizing surplus food.
* To develop a digital platform to connect donors and distributors.
* To promote community engagement in fighting hunger.
* To ensure timely delivery of leftover food to the poor.
* To track and monitor donations and distributions for transparency.

**3. Student Outcomes**

By working on this project, students will:

* Gain experience in real-world software development and system integration.
* Learn teamwork, project planning, and time management.
* Understand social responsibility and how technology can solve real-life problems.
* Apply knowledge of databases, UI/UX design, and networking.
* Gain hands-on experience with full-stack development and deployment.

**4. System Requirements**

**Hardware Requirements**

|  |  |
| --- | --- |
| **Component** | **Specification** |
| Processor | Intel i5 or higher |
| RAM | Minimum 8 GB |
| Storage | 256 GB SSD or more |
| Internet | Stable broadband connection |
| Devices | Smartphone (Android/iOS) for app users |

**Software Requirements**

|  |  |
| --- | --- |
| **Software/Tool** | **Purpose** |
| Operating System | Windows 10 / macOS / Linux |
| Backend | Node.js / Python (Flask/Django) |
| Frontend | React.js / HTML-CSS-JS / Flutter |
| Database | MySQL / MongoDB |
| Hosting | Firebase / AWS / Heroku / Netlify |
| Version Control | Git & GitHub |
| IDE | VS Code / PyCharm |
| Testing Tools | Postman, Jest, Selenium |

**5. Project Duration:** 300 hrs

|  |  |  |  |
| --- | --- | --- | --- |
| **Phase No.** | **Phase Name** | **Description** | **Page No.** |
| 1 | Requirement Analysis & Planning | Gather requirements, identify stakeholders, define workflows, and prepare timelines. | 3 |
| 2 | Salesforce Development – Backend Configurations | Configure custom objects, workflows, and API integrations in Salesforce. | 3-4 |
| 3 | UI/UX Development & Customization | Design donor portal, NGO dashboard, and volunteer interfaces with Lightning pages. | 4 |
| 4 | Data Migration, Testing & Security | Migrate data, test functionality, and ensure security with role-based access controls. | 4-5 |
| 5 | Deployment, Documentation & Maintenance | Deploy solution, prepare documentation, and set up maintenance protocols. | 5 |

**Requirement Analysis & Planning**   
In this phase, the primary goal is to clearly define the scope and objectives of the project. The team begins by identifying all key stakeholders, which include food donors such as restaurants, event organizers, and households; distribution partners like NGOs and community kitchens; volunteers who manage pickup and delivery; and administrators who oversee the entire operation. Detailed requirements are collected to understand the flow of food donations from the source to the final beneficiaries. During this stage, workflows are mapped to ensure clarity on how requests will be logged, assigned, and fulfilled. Along with functional workflows, technical needs such as the use of Salesforce, integration with mapping tools, and communication channels are documented. A tentative project plan is created, with timelines, milestones, and resource allocation carefully outlined. This phase sets the foundation for the rest of the development process by ensuring all stakeholders have a shared vision of the project.

**Salesforce Development – Backend Configurations**   
The second phase involves building the backbone of the solution within Salesforce. Custom objects and fields are created to handle critical data such as donor details, food availability records, pickup requests, beneficiaries, and delivery logs. Relationships between these objects are defined to ensure smooth data flow within the platform. Automation tools like Process Builder and Salesforce Flow are configured to trigger alerts, assign volunteers automatically, and update records in real time. For example, when a donor logs available food, a notification can instantly be sent to the nearest volunteer for pickup. Integration with third-party APIs is also planned in this stage, particularly for maps (to optimize delivery routes) and communication services like SMS and email (to send timely notifications to donors, NGOs, and volunteers). By the end of this phase, the system has a strong and reliable backend ready to support real-world workflows.

**UI/UX Development & Customization**   
Once the backend is ready, the project shifts focus to designing and customizing the user experience. The interfaces are developed with accessibility and ease of use in mind, given the diversity of stakeholders. Donors get a simple portal or app interface where they can quickly log the type and quantity of food they wish to donate, along with pickup location and timing. NGOs and volunteers are provided with dashboards that give them real-time visibility of food requests, pickup assignments, and delivery progress. Administrators are given comprehensive monitoring tools to track the entire ecosystem. Salesforce Lightning pages are customized with layouts, themes, and components to make navigation smooth and visually engaging. Mobile accessibility is prioritized to ensure that volunteers can manage their tasks easily while on the move. This phase ensures that the technical capabilities developed earlier are presented in a user-friendly and practical way, enabling seamless interaction for all stakeholders.

**Data Migration, Testing & Security**   
The fourth phase is critical for ensuring that the system runs smoothly and securely. Any existing data, such as lists of partner NGOs, registered volunteers, and donor organizations, is migrated into Salesforce. The migration process is carefully planned to ensure accuracy and avoid duplication. Once the data is in place, the system undergoes extensive testing. Functional testing checks whether each feature, such as logging a donation or sending notifications, works correctly. Performance testing ensures that the system can handle multiple requests simultaneously, especially during peak donation times. Usability testing is also performed to confirm that all user groups find the system intuitive. Security is a top priority in this phase, as sensitive information about donors and beneficiaries is stored in the system. Measures such as role-based access control, record-level security, and data encryption are implemented. Compliance with privacy regulations is ensured so that the system can be trusted by both individuals and organizations.

**Deployment, Documentation & Maintenance**   
In the final phase, the fully tested system is deployed into a live Salesforce environment. The deployment is carefully managed to minimize downtime and ensure that all configurations and integrations function as expected in the production environment. Once live, the system is made accessible to all stakeholders, and initial monitoring is carried out to address any unforeseen issues quickly. Comprehensive documentation is prepared at multiple levels: user manuals for donors, NGOs, and volunteers to guide them through daily operations; administrator guides for managing and configuring the system; and technical documentation for future developers who may enhance the solution. A structured maintenance plan is also established, which includes monitoring system performance, applying updates, and adding new features as needs evolve. Future improvements such as predictive analytics for food demand and AI-driven optimization for volunteer assignment are considered for long-term sustainability. This phase ensures that the project does not end at deployment but continues to grow and improve as it serves society.