**4. Proposed Methodology**

**4.1 Proposed System**

The proposed application is web-based, developed on Wamp 2.2e using php and sql requires internet connection and will provide a platform for donors and seekers after they successfully register into the system. If a user wishes to donate a wastage of food, he/she can send a message in application. This message will be shown as notification in donations tab to other users. This message will be stored in backend in the database. Once a notification is sent, the orphanages who wish to claim the donations can reply to the donor and contact him/her. The user interface of this system will be simple and user-friendly, and the targeted system is web. At present, we are aiming to avoid the major wastage that usually happens in India and that is foodstuffs. Also the application will be beneficial if donors and seekers are located near each other. The use case diagram shown above describes 3 actors – Donor, Agent and Admin. The Donor performs operations like Registration and Login into the System. He can also put up items for donation and view all donation requests (items required by organizations). The Admin and Donor both can view the Agents’s location. The Admin can also monitor and update the database. The Admin and Agent both can view the Donor’s location. The Agent can also perform operations like requesting for items, viewing requested items and claiming donations.



This is the way our system will work.

**4.2 Proposed Algorithms**

**4.2.1 Floyd- Warshall Algorithm**

The Floyd- Warshall algorithm is applied to distance recalculation. This algorithm was chosen due to the fact that we are using metric system and there the negative values of edges are not used. The algorithm (Floyd-Warshall) also computes straight the vertices distance, which is less time consuming than i.e. Dijkstra Algorithm (which computes distances always for each vertex)

**4.2.2** **Heuristic Algorithm for Routing Optimization**

Heuristics algorithm is based on the model of undirected weighted graph. The constructive heuristics algorithm is applied in the developed system and uses nearest-neighbor (NN) approach. The algorithm that is used to solve TSP has several iterations.

1) Donater is in initial point as the current vertex.

2) Search the edge that has the least weight between the current vertex and the unvisited vertex V.

3) V is set as the current vertex.

4) Mark V as visited vertex.

5) Repeat step 2.

6) If all vertices have been visited, stop the iteration and go back to initial point.

A simple web-based application was developed to test the heuristic algorithm for routing optimization. This testing application also utilizes Google Maps API. In this application, user can input several locations/addresses, and then Google Maps API will return the coordinate of those locations. This function can be seen in Fig 4.



**4.3 Project Development Model**

**4.3.1 Spiral Model**

For this FWMS project, we are using spiral development methodology. Spiral model is a

software development process that combines both prototyping and design in stages; its

basic concepts are as follows:



Focus is on risk assessment and on minimizing project risk by breaking a project into smaller segments and providing more ease of change during the development process, as well as providing the opportunity to evaluate risks and weight consideration of project continuation throughout the life cycle.

Each trip around the spiral traverses four basic quadrants:

1) Determine objectives, alternatives and constraint of the iteration.

2) Evaluate alternatives, Identify and resolve risks.

3) Develop and verify deliverables from the iteration.

4) Plans next iteration.

**Identification:**

This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase. This also includes understanding the system requirements by continuous communication between the customer and the system analyst.

.

**Design:**

Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and final design in the subsequent spirals.

**Construct or Build:**

Construct phase refers to production of the actual software product at every spiral. In the baseline spiral when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback. Then in the subsequent spirals with higher clarity on requirements and design details a working model of the system called build is produced with a version number. These builds are sent to customer for feedback.

**Evaluation and Risk Analysis:**

Risk Analysis includes identifying, estimating, and monitoring technical feasibility and management risks, such as schedule slippage and cost over-run. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback.

**4.4 System Architecture**

**4.4.1 Input Design**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**4.4.1.1 Objectives**

1.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3.When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**4.4.2 Output Design**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2.Select methods for presenting information.

3.Create document, report, or other formats that contain information produced by the system.

**4.4.2.1 Objectives**

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**4.5 Flow Chart**

**4.5.1 User**

duser.png

**4.5.2 Agent**

**dagent.png**

**4.5.3 Admin**

**dadmin.png**

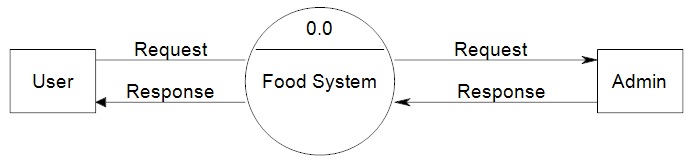
**4.5.4 Usecase Diagram**

**fusecase.png**

**4.6 Data Flow Diagram (DFD)**

|  |  |
| --- | --- |
| **Symbol** | **Description** |
| [dfd symbol](https://meeraacademy.com/wp-content/uploads/2016/09/arro.jpg) | **Data Flow** – Data flow are pipelines through the packets of information flow. |
| [dfd process](https://meeraacademy.com/wp-content/uploads/2016/09/process.jpg) | **Process :**A Process or task performed by the system. |
| [dfd entry symbol](https://meeraacademy.com/wp-content/uploads/2016/09/Entity.jpg) | **Entity :** Entity are object of the system. A source or destination data of a system. |
| [dfd source symbol](https://meeraacademy.com/wp-content/uploads/2016/09/source.jpg) | **Data Store :** A place where data to be stored |

**4.6.1 Context level DFD for Online food system**

[](https://meeraacademy.com/wp-content/uploads/2016/09/fooddfd0.jpg)

**4.6.2 1st level DFD for User**

**REPORT**

**USER ID**

**ADMIN**

USER

USER

**PASSWORD**

**DONOR**

**ADDRESS**

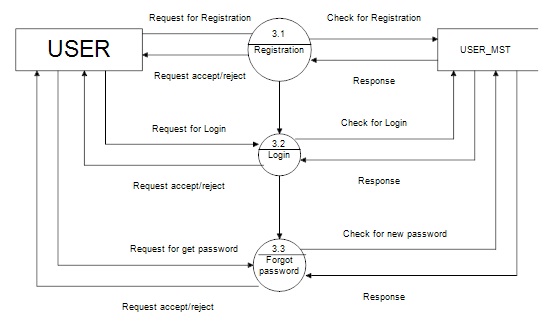
**AGENT**

**SIGN UP**

**LOGIN**

**GIVE ALERTS**

**4.6.3 2nd level DFD for user (3.0)**

[](https://meeraacademy.com/wp-content/uploads/2016/09/fooddfd2user.jpg)

**4.6.4 Admin side DFD for online food system**

