

REPUBLIQUE DU CAMEROON
PAIX-Travail-Patrie

MINISTRE DE L'ENSEIGNEMENT
SUPERIEUR

FACULTE D'INGENIERIE
ET DE TECHNOLOGIE



REPUBLIC OF CAMEROON
Peace-Work-Fatherland

MINISTER OF HIGHER
EDUCATION

FACULTY OF ENGINEERING
AND TECHNOLOGY

*****UNIVERSITY OF BUEA*****

DEPARTMENT OF COMPUTER ENGINEERING
COURSE TITLE: INTERNET PROGRAMMING AND MOBILE
PROGRAMMING
COURSE CODE: CEF440

TASK

- **REQUIREMENT ANALYSIS FOR
A PLATFORM TO REDUCE/STOP THE
WASTAGE OF FOOD**

INSTRUCTOR: Dr KAMENI VALERY

Group 12 Members:

S\N	NAMES	MATRICULE
I.	AYAMBA BLESS BISSONG	FE20A018
II	NGONG MARINUS NGONG	FE20A077
III	NDIBONG ENOW EBAI-TABOT	FE20A072
IV	MBI ENOW LEONARD APELGRYN	FE20A063
V	AMINJA SIMPLICE AJONG	FE20A009

Table of content

1	INTRODUCTION AND TERMINOLOGIES.....
	• Problem Statement.....
	• Purpose.....
	• Scope.....
	• Objective.....
2	OVERALL DESCRIPTION
	• Product Perspective.....
	• Product Functions.....
	• User Characteristics.....
	• Constraints.....
	• Assumptions and Dependencies.....
3	SYSTEM FEATURES
4	EXTERNAL INTERFACE REQUIREMENTS
	• User Interfaces.....
	• Hardware Interfaces.....
	• Software Interfaces.....

- Communications Interfaces.....

5 METHODS OF COLLECTING SOFTWARE REQUIREMENT

6 FUNCTIONAL REQUIREMENTS

7 NON-FUNCTIONAL REQUIREMENTS

- Performance Requirements.....
- Safety Requirements.....
- Security Requirements.....
- Software Quality Attributes.....

8 REQUIREMENTS ANALYSIS

- Requirement types.....
- Data requirements.....
- Functional requirements.....
- non-functional requirements.....
- Performance requirement.....
- System requirements.....

9 TECHNIQUES USE TO IDENTIFY SOFTWARE REQUIREMENT

- Data flow programs (DFP).....
- Use cases.....
- User stories.....

10 ANALYSIS TOOLS

11 ACTIVITIES INVOLVED IN REQUIREMENT ANALYSIS

12 PARTIES INVOLVED

- Charity organizations.....
- Orphanages.....
- Restaurants.....
- Sellers or buy and sellers.....
- Homes users.....

13 DESIGN AND IMPLEMENTATION

- use case diagram of the system.....
- activity diagram of the system.....

14 CONCLUSION

1 INTRODUCTION

Food waste is a major issue that has significant economic, environmental, and social impacts. In order to address this issue, many organizations are turning to food wastage management system software to help them track and reduce food waste. This report will provide an overview of the requirements for such a software, including data, functional, non-functional, and system requirements. By implementing a food wastage management system software, organizations can gain greater insight into their food waste and take steps to reduce it, resulting in cost savings and a more sustainable operation.

1 PROBLEM STATEMENT

Food waste is a major issue with significant impacts. One challenge in addressing food waste is the lack of accurate data on the types and amounts of food being wasted. A food wastage management system software can help organizations collect and analyze data on food waste, providing them with the insights they need to reduce waste and improve their operations.

2 PURPOSE

Is to help organizations collect and analyze data on food waste in order to identify key waste reduction opportunities and take effective action to reduce waste. By implementing a food wastage management system software, organizations can gain greater insight into their food waste and take steps to reduce it, resulting in cost savings and a more sustainable operation.

3 SCOPE

Researching and evaluating different food wastage management system software options

- Identifying the data, functional, non-functional, and system requirements for the software
- Implementing the chosen food wastage management system software
- Collecting and analyzing data on food waste using the software
- Identifying key waste reduction opportunities based on the data collected
- Taking action to reduce food waste based on the insights gained from the software

4 OBJECTIVES

- To implement a food wastage management system software that meets the data, functional, non-functional, and system requirements for business and organizations.
- To collect and analyze data on food waste using the software in order to identify key waste reduction opportunities.
- To take action to reduce food waste based on the insights gained from the software.
- To achieve cost savings and improve the sustainability of the organization's operations by reducing food waste.

2 OVERALL DESCRIPTION

I. Product Perspective:

The proposed system is a web-based application designed to reduce or stop food wastage in a community. It would allow market sellers, restaurant owners, charity organizations, and orphanages to register and upload their surplus food stocks at a cheaper price. Users would be able to register, view the different food options, and purchase food from the system at all times.

II. Product Functions:

- Permit market sellers to register and upload their food stocks into the system at a cheaper price.
- Permit restaurant owners to register their restaurants and display the different meals prepared in the system along with their prices.
- Permit users to register, view the different food uploaded by restaurants and market sellers, and purchase food from the system at all times.
- Permit charity organizations and orphanages to register into the system and purchase food from restaurants and market sellers.
- Integrate a payment system using an API from a payment provider.

III. User Characteristics:

- The users of this system would include market sellers, restaurant owners, charity organizations, orphanages, and individual users. These users would need to have access to a device with an internet connection in order to use the system.

IV. Constraints:

Some potential constraints for this project could include:

- Limited resources for development and maintenance of the system.
- Ensuring that all parties involved are able to use the system effectively.
- Ensuring that the payment system is secure and reliable.

V. Assumptions:

- It is assumed that market sellers, restaurant owners, charity organizations, and orphanages in the community would be willing to participate in the system and upload their surplus food stocks at a cheaper price.
- It is assumed that users would be interested in using the system to purchase food at a cheaper price.
- It is assumed that a payment provider would be available to integrate with the system for processing payments.

VI. Dependencies:

- The development of this system would depend on the availability of resources such as funding and personnel.
- The success of this system would depend on the participation of market sellers, restaurant owners, charity organizations, and orphanages in the community.
- The system would depend on a reliable internet connection for users to access and use it.

3 SYSTEM FEATURES

I. Registration:

The system should allow market sellers, restaurant owners, charity organizations, and orphanages to register and create profiles.

II. Food Stock Upload:

Market sellers and restaurant owners should be able to upload their food stocks into the system at a cheaper price. Restaurant owners should also be able to display the different meals they prepare along with their prices, name, and location.

III. User Access:

Users should be able to register and view the different food options uploaded by restaurants, market sellers, charity organizations, and orphanages.

IV. Food Purchase:

Users should be able to purchase food from the system at all times from all parties that upload food into the system.

V. Charity and Orphanage Support:

Charity organizations and orphanages should be able to register and purchase food from restaurants and market sellers. They should also be able to provide physical locations in the community where users can eat for free.

VI. Payment System:

The system should use an API for any payment system that permits payment.

VII. Technology Stack:

The system should be designed with a MySQL database, HTML, CSS, JavaScript and PHP with their respective frameworks.

4 EXTERNAL INTERFACE REQUIREMENTS

I. User Interfaces:

The user interface of the system would allow users to easily register, view and purchase food from market sellers, restaurant owners, charity organizations and orphanages. It would display the different meals prepared by restaurants along with their prices, name and location. The interface would be designed to be user-friendly and intuitive to use.

II. Hardware Interfaces:

The hardware interfaces of the system would include the devices used to access the system such as computers, tablets or smartphones. These devices would need to meet certain requirements in terms of processing power, memory and connectivity to ensure smooth operation of the system.

III. Software Interfaces:

The software interfaces of the system would include the APIs used to integrate with payment systems and other external systems. The system would be designed using a MySQL database, HTML, CSS, JavaScript and PHP with their respective frameworks.

IV. Communications Interfaces:

The communications interfaces of the system would include the protocols used for data transfer between the different components of the system such as HTTP or HTTPS. These protocols would need to be secure and reliable to ensure smooth operation of the system.

5 METHODS OF COLLECTING SOFTWARE REQUIREMENTS

I. Brainstorming:

Brainstorming sessions with users and stakeholders can help generate new ideas and identify potential requirements.

II. Document Analysis:

Analyzing existing documentation such as user manuals, process flows, and system specifications can provide valuable insights into user needs and requirements.

III. Workshops:

Workshops can be used to bring together users and stakeholders to collaboratively define and refine requirements.

IV. Use Cases:

Developing use cases that describe how users will interact with the system can help identify requirements.

V. User Stories:

User stories are short, simple descriptions of a feature or function from the perspective of the user. They can be used to capture and prioritize requirements.

VI. Interviews:

Conducting interviews either one-on-one or in small groups can help gather information about user needs and problems.

VII. Focus Groups:

Bringing together user representatives from different areas to present information about needs and problems can help gather a broad range of perspectives.

VIII. Prototype:

Creating a mock-up or prototype of the solution for users to interact with and give feedback and input can help refine the requirements.

IX. Surveys:

Surveys can be used to gather information from a large number of users quickly and efficiently based on each user's perspective.

6 REQUIREMENT ANALYSIS

1 Requirement types

- I. **Business requirements:** These outline measurable project goals for the business, users and other stakeholders.
- II. **User requirements:** These are specified by the users of the product from their perspective.
- III. **Software requirements:** These identify and clarify the why, what and how of a business's application.
- IV. **Functional requirements:** These specify what is expected from the product by the user.
- V. **Non-functional requirements:** These are not related to the functionality of the product but rather to its usability, look and feel, security, reliability, performance etc.

2 Data requirement

- I. The ability to collect data on the types and amounts of food being wasted
- II. The ability to analyze data to identify key waste reduction opportunities
- III. Compliance with regulatory requirements for data collection and reporting

3 System requirements

- I. A system that can track food waste and identify areas where the wastage occurs
- II. A system that can help businesses and individuals reduce their food waste by providing tips and suggestions
- III. A system they can connect businesses with local food banks or other organizations that can use excess food
- IV. It should be able to help the businesses and individuals compost their food waste
- V. Compatibility with the hardware and operating system used in the kitchen

- VI. Internet connection for data collection and analysis

7 FUNCTIONAL REQUIREMENTS

- I. User connects to the site online, and login, view food, change credentials, and view what's new and you can get.
- II. A system that can track food waste and identify areas where the wastage occurs
- III. A system that can help businesses and individuals reduce their food waste by providing tips and suggestions
- IV. A system they can connect businesses with local food banks or other organizations that can use excess food
- V. It should be able to help the businesses and individuals compost their food waste
- VI. The system should provide analysis on food waste reduction effort
- VII. The system should notify the availability of food other organizations

8 NON- FUNCTIONAL REQUIREMENT S

- I. The system should be easy to use and navigate.
- II. The system should be scalable to accommodate growth in the future.
- III. The system should be secure and protect user data.
- IV. The system should be reliable and available 24/7.
- V. It should be User-friendly and with easy-to-use interfaces.
- VI. It should be reliable and accurate data collection and analysis
- VII. Scalability to handle increasing amounts of data as the business grows

9 SOME TOOLS FOR SOFTWARE ANALYSIS

These tools help identify customers' needs and provide a high level of customization for enterprise requirements management. They also offer a variety of requirement analysis techniques such as business process modeling notation e.g., GANTT CHARTS, FLOWCHARTS and GAP ANALYSIS.

I. Jama Software:

Jama Software is a requirements management tool that helps teams define, track, and validate requirements throughout the product development process. It provides features such as traceability, collaboration, and test management to help teams deliver high-quality products on time and within budget.

II. Visure Requirements:

Visure Requirements is a requirements management tool that provides a collaborative platform for managing requirements throughout the product development process. It offers features such as traceability, version control, and reporting to help teams ensure that their requirements are complete, consistent, and testable.

III. ReqSuite RM:

ReqSuite RM is a requirements management tool that helps teams manage their requirements in a structured and efficient manner. It provides features such as templates, traceability, and collaboration to help teams define and manage their requirements throughout the product development process.

IV. Zoho Projects:

Zoho Projects is a project management tool that helps teams plan, track, and collaborate on their projects. It offers features such as task management, time tracking, and reporting to help teams stay organized and on track.

V. SpiraTeam:

SpiraTeam is an application lifecycle management tool that helps teams manage their requirements, test cases, and defects in one integrated platform. It provides features such as traceability, collaboration, and reporting to help teams deliver high-quality products on time and within budget.

VI. Xebrio:

Xebrio is a project management tool that helps teams manage their projects from start to finish. It offers features such as task management, time tracking, and collaboration to help teams stay organized and on track.

VII. Atlassian Jira:

Atlassian Jira is an issue tracking and project management tool that helps teams plan, track, and release software. It provides features such as agile boards, dashboards, and reporting to help teams deliver high-quality software on time and within budget.

VIII. IBM Engineering Requirements:

IBM Engineering Requirements (formerly known as DOORS) is a requirements management tool that helps teams capture, trace, analyze, and manage their requirements throughout the product development process. It provides features

such as traceability, version control, and collaboration to help teams ensure that their requirements are complete, consistent, and testable.

IX. Innoslate:

Innoslate is a system engineering tool that helps teams manage their requirements, models, tests, and other artifacts in one integrated platform. It provides features such as traceability, simulation, and verification to help teams deliver high-quality systems on time and within budget.

10 ACTIVITIES INVOLVED IN REQUIREMENT ANALYSIS

I. IDENTIFY CUSTOMER'S NEEDS

This involve identifying the customers' requirements and expectations for the software

II. EVALUATE SYSTEM FOR FEASIBILITY

This involve assessing whether the software can be developed within the given constraints such as time, budget, and resources.

III. PERFORM ECONOMIC AND TECNICAL ANALYSIS

This involve analyzing the economic and technical feasibility of the software

IV. ALLOCATE FUNCTIONS TO SYSTEM ELEMENTS

This involve breaking down the requirements int smaller components and assigning them to different parts of the system.

V. ESTABLISH SCHEDULE AND CONSTRAINTS

This involve setting up a timeline for the development of the software and identifying any constraints that may affects its development.

I. CREATE SYSTEM DEFINITIONS

This involve defining the system requirement in detail.

11 SOME TECHNIQUES USE TO IDENTIFY SOFTWARE REQUIREMENT

I. Data Flow Programs (DFP):

Data-flow analysis is a technique used by software engineers to analyze the way values of variables change over time as a program is executed. The data gained from this process may be used for optimizing or debugging the software [1](#).

II. Use Cases:

A use case is a technique for capturing the potential requirements of a new system or software change. Each use case provides one or more scenarios that convey how the system should interact with the end user or another system to achieve a specific business goal.

III. User Stories:

A user story is an informal, natural language description of one or more features of a software system. User stories are often written from the perspective of an end user or user of a system and are often used in agile software development methodologies.

12 PARTIES INVOLVE

I. Charity organizations:

Charity organizations would be able to register into the system and purchase food from market sellers and restaurants at a cheaper price. They would also be able to give physical locations in the community where users can eat.

II. Orphanages:

Orphanages would play a similar role to charity organizations in the system. They would be able to register and purchase food from market sellers and restaurants at a cheaper price.

III. Restaurants:

Restaurant owners would be able to register their restaurants into the system and display the different meals they prepare along with their prices, name and location. They would also be able to upload the food they prepare into the system.

IV. Market Sellers:

Market sellers would be able to register into the system and upload their food stocks at a cheaper price. This would help reduce food waste and provide an additional source of income for them.

V. Home users:

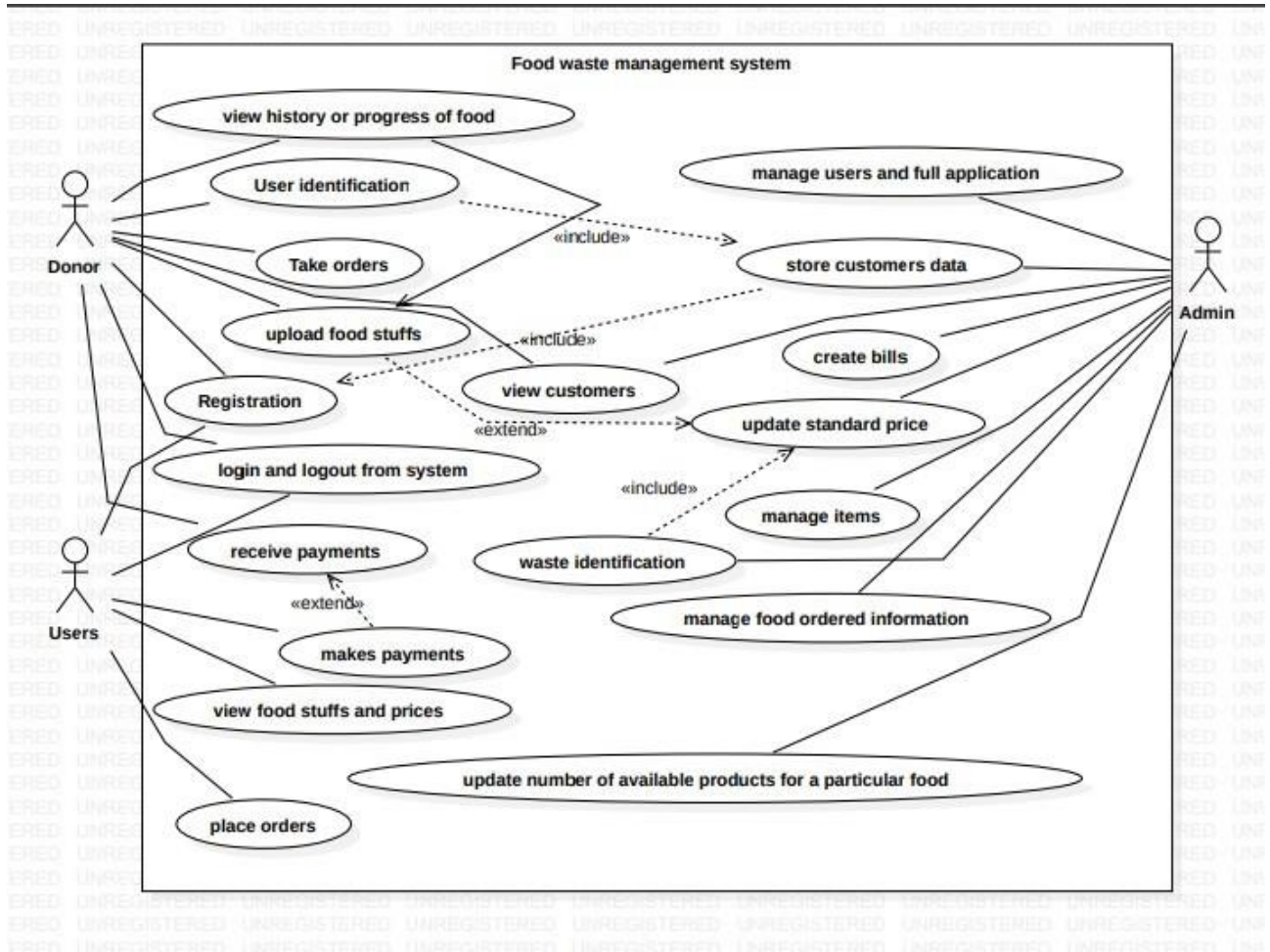
Home users would be able to register into the system and view the different food uploaded by restaurants, market sellers, charity organizations and orphanages. They would be able to purchase food from these parties at all times through the system.

VI. Admin:

The admin is there to ensure that all activities in the system are running as required and all other compliances are perfect.

13 DESIGN AND IMPLEMENTATION

I. use case diagram of the system



14 CONCLUSION

This system's aims to reduce food waste and help feed undernourished communities by allowing market sellers, restaurant owners, charity organizations and orphanages to register and upload their food stocks at a cheaper price. Users would be able to register and purchase food from these parties at all times through the system. The system would be designed with a user-friendly interface and would use secure and reliable hardware, software and communication interfaces. By bringing together these different parties and providing a platform for them to sell their surplus food, the system has the potential to make a positive impact on both the environment and the community.