

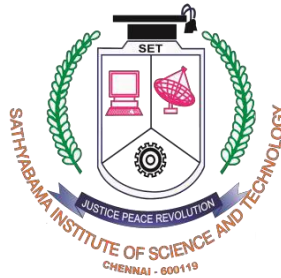
PROFESSIONAL TRAINING REPORT
At
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology Degree in

Information Technology

By

SNEHA.J
(REG.NO.37120071)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119. TAMILNADU.

AUGUST 2019



SATHYABAMA
Institute of science and technology
(DEEMED TO BE UNIVERSITY)
(Established under Section 3 of UGC Act,
1956) Jeppiaar Nagar, Rajiv Gandhi Salai,
Chennai 600119
www.sathyabamauniversity.ac.in



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Professional Training Report is the bonafide work of **SNEHA.J (REG.NO.37120071)** who underwent the professional training in “**FOOD WASTE MANAGEMENT**” under our supervision from June 2019 to august 2019.

INTERNAL GUIDE

Mrs.JEYANTHI.M.E., Ph.D.,

HEAD OF THE DEPARTMENT

Dr. R. SUBHASHINI, M.E., Ph.D.

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

I **SNEHA`J(Reg.No.37120071)** hereby declare that the Professional Training Report on “**FOOD WASTE MANAGEMENT** ” done by me under the guidance of **Mrs.JEYANTHI,M.E., Ph.D.**, at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind Encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr.T.Sasikala M.E., Ph.D, Dean**, School of Computing, **Dr.SUBHASHINI M.E., Ph.D.** , Head of the Department of Information and Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Mrs.JEYANTHI M.E., Ph.D., for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE



TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO.
		i
	ABSTRACT	ii
	LIST OF FIGURES	iii
	LIST OF TABLES	iv
	LIST OF ABBREVIATIONS	
	INTRODUCTION	
1	1.1Synopsis	9
	1.2Objective	10
		10
2	SYSTEM ANALYSIS	11
	2.1 PROBLEM DEFINITION	11
	2.1.1Existing System	11
	2.1.2Proposed System	11
	2.2FEASIBILITY ANALYSIS	12
	2.2.1Technical Feasibility	12
	2.2.2Operational Feasibility	13
	2.2.3Economical Feasibility	13
3	SOFTWARE REQUIREMENT SPECIFICATION	14
	3.1REQUIREMENTS ANALYSIS	
	3.1.1Purpose	
	3.1.2 scope	14
		15

	3.1.3 overview	15
	3.2General discriptions	
	3.2.1 product function	16
	3.2.1user characteristics	16
	3.3System Requirement	
	3. 2.1Hardware Requirements	16
	3.2.2Software Requirements	
4	SOFTWARE DESCRIPTION	
	4.1 Features	17
	4.2 Components	
	4.2.1 The common language runtime	19
5	SYSTEM DESIGN	
	5.1 Use case diagram	21
	5.2 sequence diagram	21
	5.3 activity diagram	22
	5.4 class diagram	23
	5.5 Architecture diagram	23
	5.6 Entity relationship diagram	24

6	IMPLEMENTATION	
	6.1 Modules description	
	6.1.1 Admin module	24
	6.1.2 donar module	24
	6.1.3 agent module	25
	CONCLUSION AND FUTURE ENHANCEMENT	
	7.1Conclusion	27
	REFERENCE	
7	APPENDIX	
	A.SCREENSHOTS	29
	B.SOURCE CODE	30

ABSTRACT

This project is used to manage wastage foods in a useful way. Every day the people are wasting lots of foods. So we have to reduce that food wastage problem through online. If anyone have wastage foods they are entering their food quantity details and their address in that application and then the admin maintain the details of food donator.

The donator can create the account and whenever they are having wastage food they can login and give request to the admin. And the admin also maintain the buyer (orphanage, poor people,..) details too. After the admin view the donator request and give the alert message like time to come and collect the food.

And the admin collect foods from donator through their nearby agent then provide to nearest orphanages or poor people. After receiving the food from the agent by admin and give alert message to that donator. If the donator need any detail about the orphanage with helping thought they can give request to the admin and collect the orphanage details.

This project is food redistribution is an enormously successful social innovation that tackles food waste and food poverty. The user's details are maintained confidential because it maintains a separate account for each user.

LIST OF FIGURES		
FIGURE NO.	FIGURE NAME	PAGE NO.
5.1	Use case diagram	21
5.2	Sequence diagram	21
5.3	Activity diagram	22
5.4	Class diagram	23
5.5	Architecture diagram	23
5.6	Entity relationship diagram	24

TABLE NO.	LIST OF TABLES TABLE NAME	PAGE NO
6.1.1	Insertion page	24
6.1.2	search page	24
6.1.3	delete page	25

LIST OF ABBREVIATION

ABBREVIATION

EXPANSION

CLS	COMMON LANGUAGE RUNTIME
CLR	COMMON LANGUAGE SPECIFICATION
SQL	STURCTURE QUERY LANGUAGE
XML	EXTENSIBLE MARKUP LANGUAGE

CHAPTER 1

INTRODUCTION

Hunger is a severe global problem and rather contradictory topic relative to the considerable amount of food waste [1]. In 2017, according to Food and Agriculture Organization of the United Nations (FAO), 820.8 million (10.9%) people was undernourished [2, p. 4]. Despite a reduction from 14.5% in 2005 to 10.9% in 2017, it is still vital to ensure further reductions thus creating an elevated quality of life on a global scale [2, p. xiv]. A part of lowering the percentage will require that new ideas are examined based on both existing and new technologies.

1.1SYNOPSIS

According to the National Research Council (2006), hunger is defined as a short-term, physical discomfort due to chronic food shortage, or in more serious cases, a life-threatening lack of food. When talking about world hunger, some related terms could be food insecurity and malnutrition. Food insecurity is when the access to food is limited or unreliable, with causes such as natural disasters, climate change, or war.

Malnutrition refers to an inadequate intake of the necessary nutrients, which could mean both overnutrition and undernutrition. Undernutrition is the most common topic when talking about global hunger. The most important type of undernutrition is protein-energy malnutrition, which refers to a lack of protein and calories. The problem with protein-energy malnutrition, is that protein is needed for key body functions, as well as the growth and maintenance of the muscles.

The consequences of malnutrition can be wasting or stunting. Wasting is when a person is having a very low weight compared to their height. Stunting is the term for reduced growth rate and human development compared to the standard height for age. It is a very slow process and it develops over a long period of time without the proper nutrition.

The total number of undernourished people worldwide is estimated to have reached 820.8 million people in 2017. As seen in figure 1.1 on the following page this number has mostly been decreasing between 2005 and 2014, but since 2015 the number has started to increase, with around 20 million more people suffering from undernourishment each year.

1.2OBJECTIVE

The overall objectives of the waste management assessment are summarised below:

1. The administrator maintain the agent details as well as the donator details. The administrator collect the food from the agent.
2. The administrator give the orphanage details directly to the donator
3. The donator give the wastage of food to the orphanage. The donator give the request to the admin for the purpose of to collect the wastage food.
4. The donator view the orphanage details and agent details.
5. The agent maintain the orphanage details. It can also maintain the donator details. The agent give the request to the admin for collect the food from the donator.
6. After collect the food the agent give the alert message for the donator.

CHAPTER 2

2. SYSTEM ANALYSIS

2.1.1 EXISTING SYSTEM

In existing system if anyone have extra food because of any function or in their home it will be become waste because instantly there is no way to share with anyone if they are having lots of food . even if they want to give that extra food to any orphanage or poor people they don't have time or don't have an idea about that So that we have create a application for sponsor that extra food to poor people or nearby orphanage.

2.1.2 PROPOSED SYSTEM

the admin collect foods from donator through their nearby agent then provide to nearest orphanages or poor people. After receiving the food from the agent by admin and give alert message to that donator. If the donator need any detail about the orphanage with helping thought they can give request to the admin and collect the orphanage details.

This project is food redistribution is an enormously successful social innovation that tackles food waste and food poverty. The user's details are maintained confidential because it maintains a separate account for each user

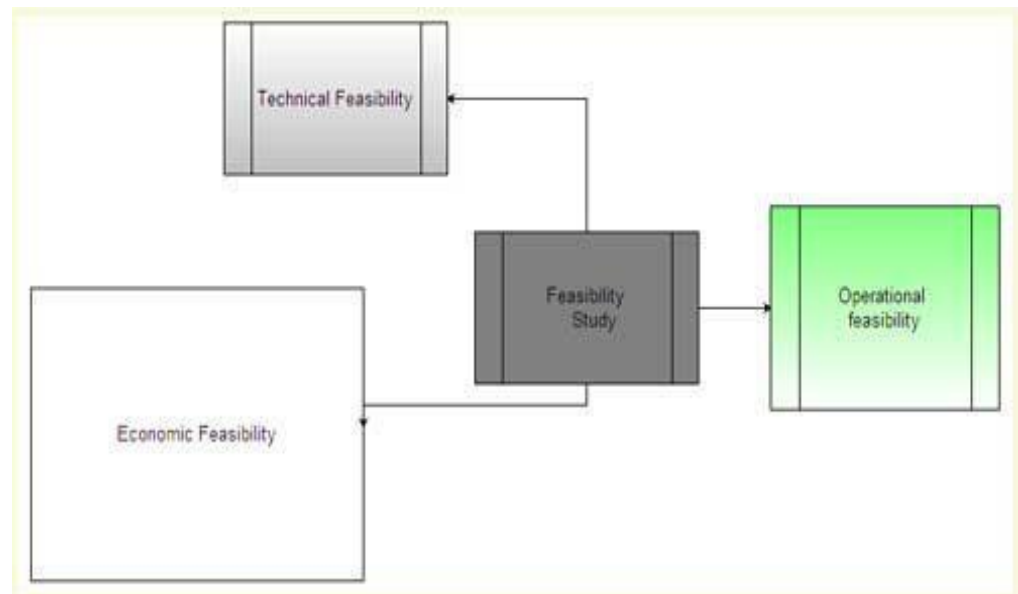
2.2 FEASIBILITY ANALYSIS

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

There are three main types of feasibility. They are as follows:

- TECHNICAL FEASIBILITY
- ECONOMIC FEASIBILITY

➤ OPERATIONAL FEASIBILITY



2.2.1 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

It measures of the availability of technical resources (hardware components or technical equipment). It also studies the availability of the technical manpower for the project.

Technical feasibility also performs the following tasks:

- Analyzes the technical skills and capabilities of the software development team members
- Determines whether the relevant technology is stable and established. This statement was agreed by our application. The software which is used can able to solve the problem in the existing using the technology.
- Technical skills such as .net languages usage and the connectivity becomes to learn more.
- Backend software SQL server 2008 R2, in this software installation, become more complications such as having the server name, local host number, database name then the connectivity become more tedious before installation.

2.2.2 OPERATIONAL FEASIBILITY

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility generally proposed to have the following properties.

Operational feasibility also performs the following tasks:

- Determines whether the problems anticipated in user requirements are of high priority
- Determines whether the solution suggested by the software development team is acceptable
- Analyzes whether users will adapt to a new software
- Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

Operational Feasibility study is necessary as it ensures that the project developed can be successfully implemented. Everything is provided in easy access manner so that any novice users of information very efficiently.

2.2.3ECONOMICAL FEASIBILITY

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on. For this, it is essential to consider expenses made on purchases and activities required to carry out software development. In addition, it is necessary to consider the benefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below.

- Cost incurred on software development to produce long-term gains for an organization
- Cost required to conduct full software investigation
- Cost of hardware, software, development team, and training

CHAPTER 3

restaurants involving aerobic and anaerobic digestion processes for recycling food waste.

The food service industry includes food waste from hotels and restaurants. The waste generated in the food service industry is due to high consumption of food & beverage products.

The scope of food waste management services has increased over time in the food service industry as various systems are being installed & utilized by hotel chains and restaurants involving aerobic and anaerobic digestion processes for recycling food waste.

The North American region is projected to hold the largest market by 2022. The region generates a major amount of the total world's food waste. Growth in this region can be attributed to the various innovations in different processes of food waste management that help to meet the growing demand of the food manufacturers.

3.2 GENERAL DESCRIPTIONS

3.2.1 PRODUCT FUNCTION

This system is used to manage hours for particular staff's and it can be maintained through online, it consists of many functions i.e. maintain insertionmodule , searchmodule, , delete module has full authority for managing the entire process.

3.2.2 USER CHARACTERISTICS

The user has to insert into the application by using the book name and author. It displays book details interface for user authentication. This insertion page for a project is used to provide a proper genuine authentication to the application with this we can restrict the unauthorized and illegal users in to the application It act as a very great security feature to the application. If the user use book name and author name it enters in to application or else it returns back in to the authentication insertion page indicating the book name is invalid.

3.3 System Requirement

3.3.1 HARDWARE REQUIREMENTS

- System : Intel inside i3
- System Type : 64-bit Operating System
- Storage : 500GB
- RAM : 4 GB

3.3.2 SOFTWARE REQUIREMENTS

- Operating system : Windows 10
- IDE : notepad++
- Front End : Javascript, css, html
- Backend : Mysql

CHAPTER 4

3. SOFTWARE DESCRIPTION

4.1 Java Script

Overview:

JavaScript is a general purpose programming language. Its main use is for scripts and applications that run on web pages inside of a browser. The scope of JavaScript is such that useful behaviour may be achieved in a couple of lines of code, for instance, showing a tooltip, or it can be a platform for large web-applications such as Gmail.

4.1.1 Features :

Object-orientation Like most modern programming languages, JavaScript is object-oriented. Objects are data-structures—containers for state and behaviour. They facilitate many established software engineering practices including: abstraction—hiding complexity and implementation details; encapsulation—privacy and logical grouping of functionality; and modularity—separation of distinct functionality. In JavaScript, objects are unordered sets of named properties, which can be of various data types including other objects. Everything is an object, except the primitive types: undefined, null, boolean, string and number. Differing from objects, these primitives have only one value and can't have properties—however, the latter three appear to have properties and can be interacted with like objects; this is because they have object wrappers and JavaScript will coerce them

4.1.2 Objectives :

1. To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.
2. To provide a code-execution environment to minimizes software deployment and guarantees safe execution of code.
3. Eliminates the performance problems.

There are different types of application, such as Windows-based applications and Web-based applications. To make communication on distributed environment to ensure that code be accessed by the .NET Framework can integrate with any other code.

4.2 COMPONENTS

Whether you are a seasoned PHP / MySQL developer, or a beginner, it really helps to have a local development environment. This allows you to test your code as you develop your web application before publishing it "live" to the Internet.

The following power tip will guide you through setting up a PHP, Apache, and MySQL development environment on your local Windows system. We'll also take a look at the UESTudio features that can make your PHP / MySQL development easier.

Please keep in mind that PHP, Apache, and MySQL are software provided by third party entities. Therefore we (IDM) cannot provide technical support for the installation of these on your system.

4.2.1 THE COMMON LANGUAGE RUNTIME (CLR):

The common language runtime is the foundation of the .NET Framework. It manages code at execution time, providing important services such as memory management, thread management, and remoting and also ensures more security and robustness. The concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged hosts.

4.3 Dynamic typing:

Object-oriented programming languages may deal with object types in one of two different ways: statically—at compile-time or dynamically—at run-time. Compiletime checking means that type errors, for instance dividing an integer by a string, can be caught before the program is ever run, thus offering a degree of safety. As well as preventing this class of run-time error, statically typed languages generally run faster since they do not have to make the type-checks at run-time. The advantage of dynamic typing is that not all type information needs to be available at compile-time. This enables features such as dynamic objects—where objects can be composed and manipulated at run-time, and the eval function—where arbitrary data can be executed as code.

JavaScript is dynamically typed. It is also weakly typed, meaning that it employs various rules for implicit type conversion. This can be useful, but can also be problematic; for example, the + operator is overloaded: it is used for both addition and concatenation. If both operands are numbers, the + means addition; however, if any operand is a string, it means concatenation. Building strings is a case where this is useful 'There are ' + amount + ' widgets ' This implicit conversion can be an issue when the programmer makes assumptions about variables' type.

4.4.1 First class functions:

In JavaScript, functions are first-class citizens of the language, meaning that they can be passed around just like any other value. The support of first-class function also means that JavaScript has higher-order functions: functions that may receive functions as arguments, or return a function. These are powerful constructs to have; providing opportunities for abstraction that result in less code, that is more expressive, and reusable

JavaScript also has anonymous functions—functions that are not bound to an identifier. These are useful for defining small, use-once functions for passing as arguments.

JavaScript functions also behave as closures—they close over the scope in which they are created, meaning all of the variables in scope at that time are available inside the function when it is called. This is useful for implementing private state, as shown in Code Snippet 2.

4.4.2 PROJECT MOTIVATION

There are two main aspects motivating this project. Firstly, the fragmentation of JavaScript environments results in the programmer only being able to rely on the 'lowest common denominator' if they want their code to be portable. Portability is especially poignant in web development, since the developer has no control over which browser is used. Secondly, the syntax of JavaScript, while a subjective matter, could arguably be greatly improved.

4.4 FRAGMENTATION:

Though the fifth and current version of ECMAScript describes a language with significant improvements over previous iterations, many instances of JavaScript environments, such as out-dated web browsers, are stuck with old implementations.

4.5 Features of SQL-SERVER

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services

SQL-SERVER database consists of six type of objects,

They are,

1. TABLE
2. QUERY
3. FORM
4. REPORT
5. MACRO

TABLE:

A database is a collection of data about a specific topic.

VIEWS OF TABLE:

We can work with a table in two types,

1. Design View
2. Datasheet View

Design View:

To build or modify the structure of a table we work in the table design view. We can specify what kind of data will be hold.

Datasheet View:

To add, edit or analyses the data itself we work in tables datasheet view mode.

QUERY:

A query is a question that has to be asked the data. Access gathers data that answers the question from one or more table. The data that make up the answer is either dynaset (if you edit it) or a snapshot (it cannot be edited). Each time we run query, we get latest information in the dynaset. Access either displays the dynaset or snapshot for us to view or perform an action on it ,such as deleting or updating.

FORMS:

A form is used to view and edit information in the database record by record .A form displays only the information we want to see in the way we want to see it. Forms use the familiar controls such as textboxes and checkboxes. This makes viewing and entering data easy.

Views of Form:

We can work with forms in several primarily there are two views,

They are,

1. Design View
2. Form View

Design View:

To build or modify the structure of a form, we work in forms design view. We can add control to the form that are bound to fields in a table or query, includes textboxes, option buttons, graphs and pictures.

Form View:

The form view which display the whole design of the form.

REPORT:

A report is used to view and print information from the database. The report can group records into many levels and compute totals and average by checking values from many records at once. Also, the report is attractive and distinctive because we have control over the size and appearance of it.

MACRO:

A macro is a set of actions. Each action in macros does something. Such as opening a form or printing a report. We write macros to automate the common tasks the work easy and save the time.

CHAPTER 5

5. SYSTEM DESIGN

Design is multi-step process that focuses on data structure software system, procedural details, (algorithms etc.) and interface between modules. The design process also translates the requirements into the presentation of software that can be accessed for quality before coding begins.

Computer software design changes continuously as new methods; better analysis and broader understanding evolved. Software Design is at relatively early stage in its revolution.

Therefore, Software Design methodology lacks the depth, flexibility and quantitative nature that are normally associated with more classical engineering disciplines. However, techniques for software designs do exist, criteria for design qualities are available and design notation can be applied.

5.1 USE CASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

Only static behavior is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction.

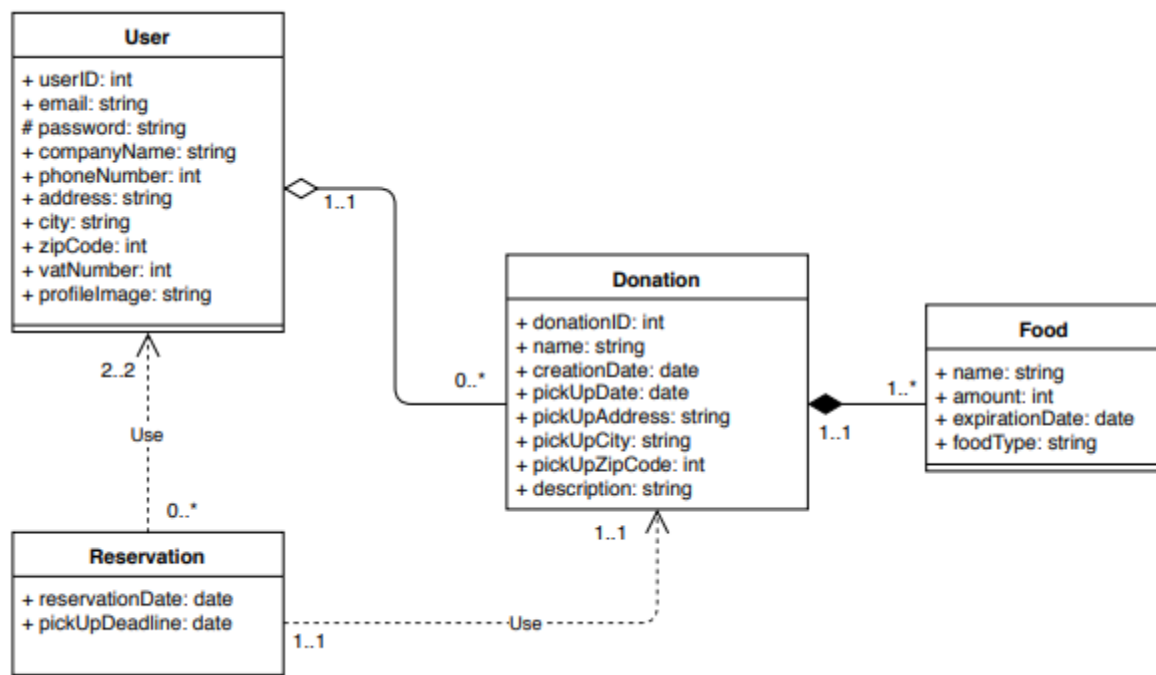
These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

5.3 ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc. Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organizational processes

5.4 CLASS DIAGRAM

In software engineering, a class diagram in the UML is a type of static structure diagram that describes the structure of a system by showing the system classes, their attributes, operations and the relationship among the classes. It explains which class contains information's and shows a collection of static model elements such as classes and types, their contents, and their relationship.



5.5 ARCHITECTURAL DIAGRAM

The design process for identifying the sub-systems making up a system and the framework for sub-system control and communication is architectural design. The output of this design process is a description of the software architecture.

- It defines an abstraction level at which the designers can specify the functional and performance behaviour of the system.
- It evaluates the architectural design



- It develops and documents top-level designs for the external and internal interface
- It develops all preliminary versions of users documentation
- It defines and documents preliminary test requirements and the schedule for software integration.

System architecture is a conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structure and behavior of the system.

Architecture Design Representation

Architecture design can be represented using the following models.

- 1. Structural model:** Illustrate architecture as an ordered collection of program components.
- 2. Dynamic model:** Specifies the behavioural aspect of the software architecture and indicates how the structure or system configuration changes as the function changes due to change in the external environment.
- 3. Process model:** Focuses on the design of the business or technical process, which must be implemented in the system.
- 4. Functional model:** Represents the functional hierarchy of a system.
- 5. Framework model:** Attempts to identify repeatable architectural design patterns encountered in similar types of application. This leads to an increase in the abstraction.

5.6 E-R DIAGRAM

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems



CHAPTER 6

6. IMPLEMENTATION

6.1 MODULES SPECIFICATION

A module is a part of a program. Programs are composed of one or more independently developed modules that are not combined until the program is linked. A single module can contain one or several routines. The project contains the following main modules.

1. Donar module
2. Agent module
3. Admin module

6.1.1 DONAR MODULE

In donator module, the donator give the wastage of food to the orphanage. The donator give the request to the admin for the purpose of to collect the wastage food. The donator view the orphanage details and agent details.

6.1.2 AGENT MODULE

In Agent module, the agent maintain the orphanage details. It can also maintain the donator details. The agent give the request to the admin for collect the food from the donator. After collect the food the agent give the alert message for the donator.

6.1.3 ADMIN MODULE

In admin module, The administrator maintain the agent details as well as the

donator

details. The administrator collect the food from the agent. The administrator give the orphanage details directly to the donator.

CHAPTER 7

5. CONCLUSION AND FUTURE ENHANCEMENTS:

7.1 CONCLUSION

The development of this product surely prompts many new areas of investigation. This product has wide scope of implementation by making it live. Moreover this product creates many benefits for the business and the community. By taking it online it will help many people throughout the city by donating food daily.

Hundreds of thousands of tons of food are either lost or wasted while millions of people suffer from malnutrition. A plausible initiative is the food donation portal in which large retail chains and potentially other organizations can donate food. This food is collected and delivered by Third Party vendor to NGO in need. Food donation portal will help thousands of people that suffer from

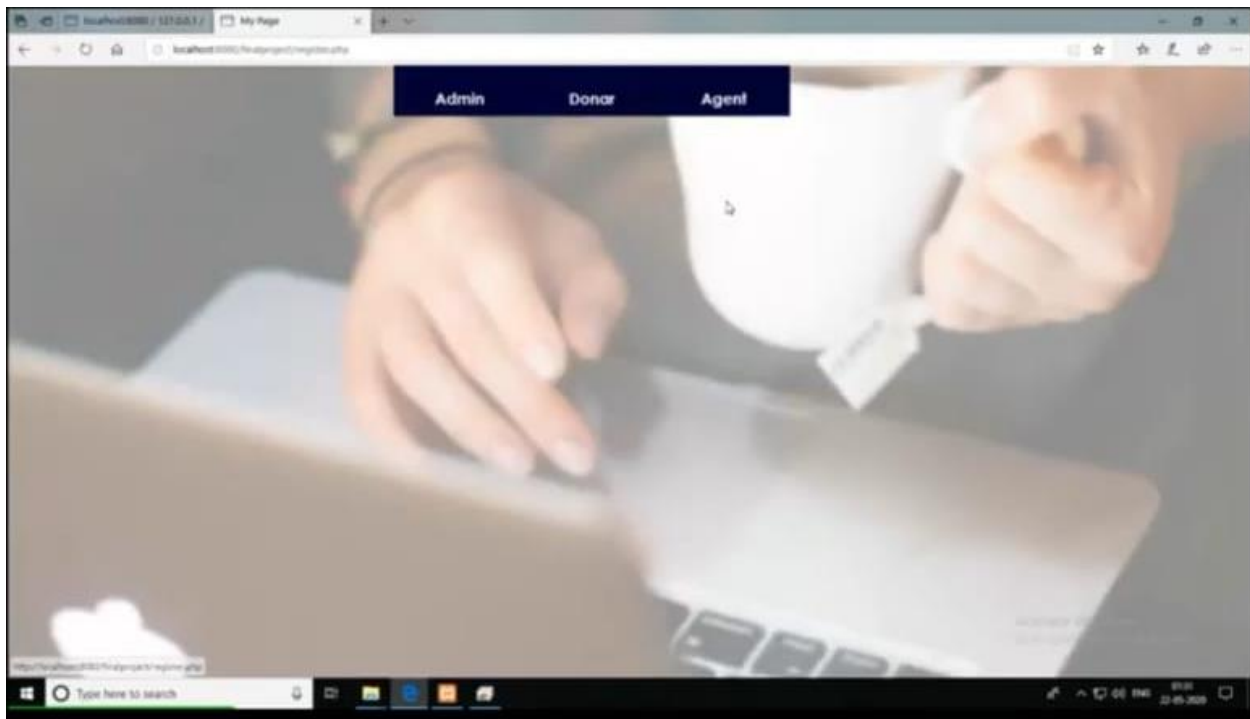
starvation and also consume food that are wasted with no reason.. As consequence, research and actions are needed to improve the efficiency of food donation portal.

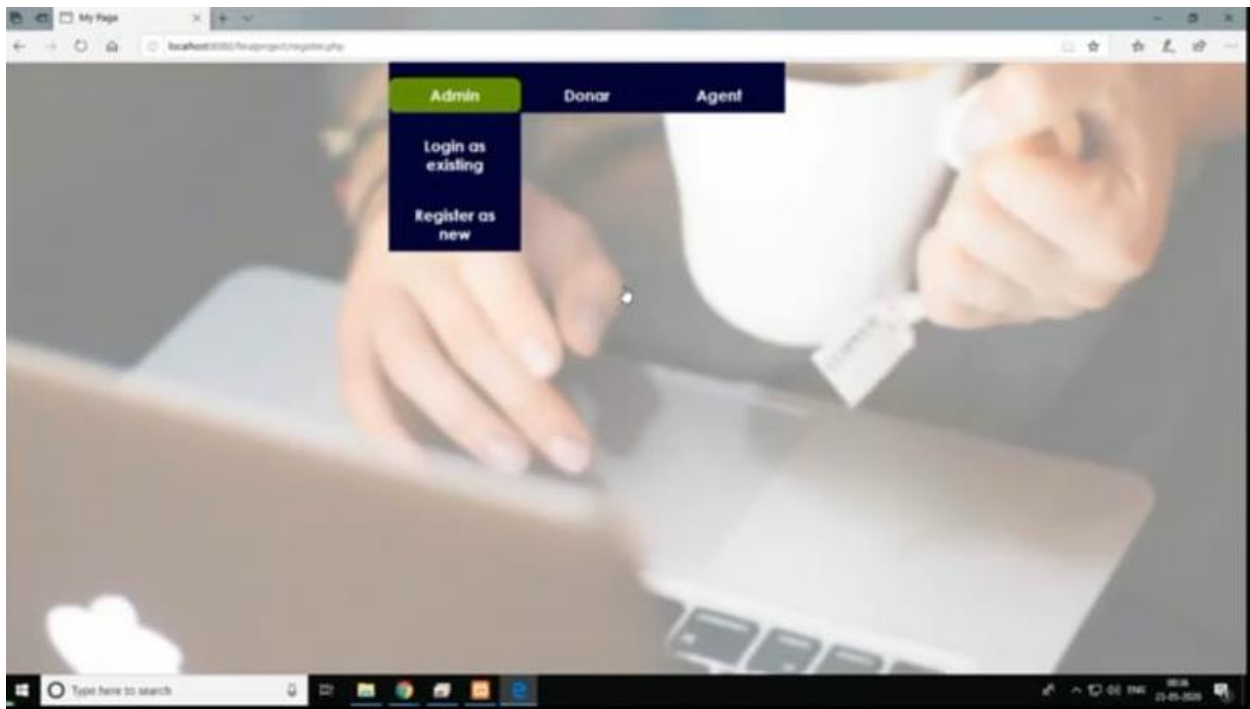
REFERENCE:-

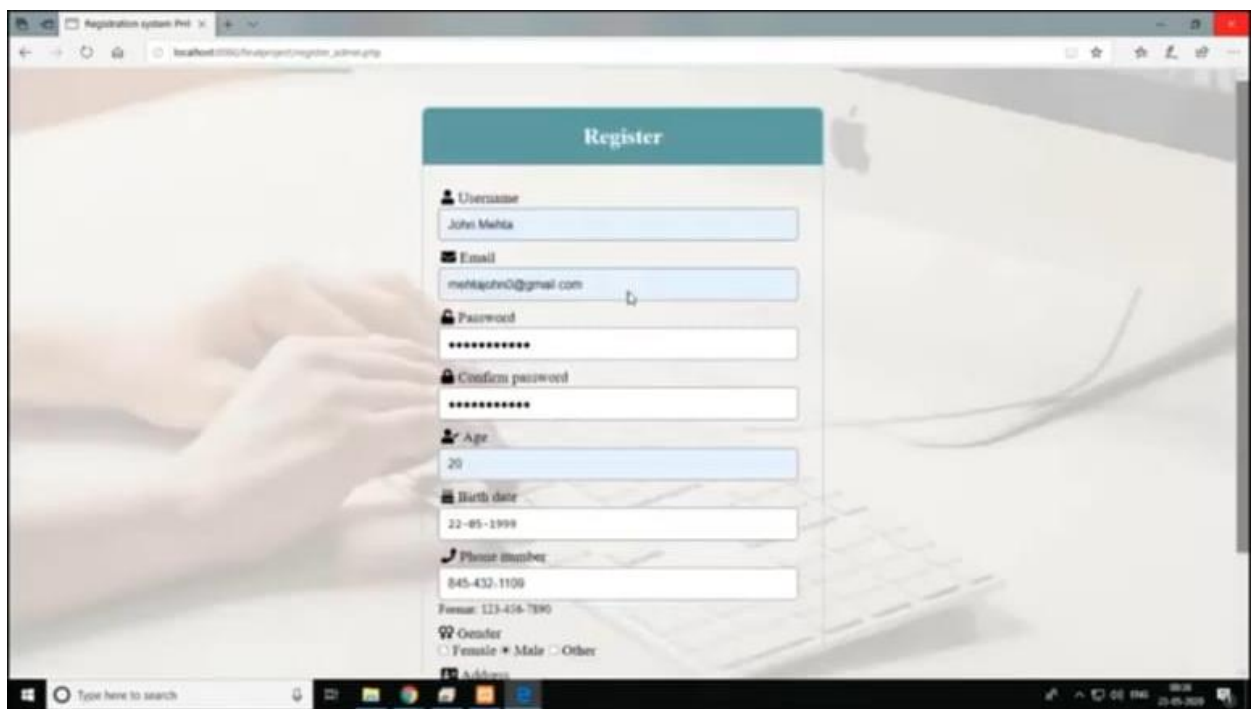
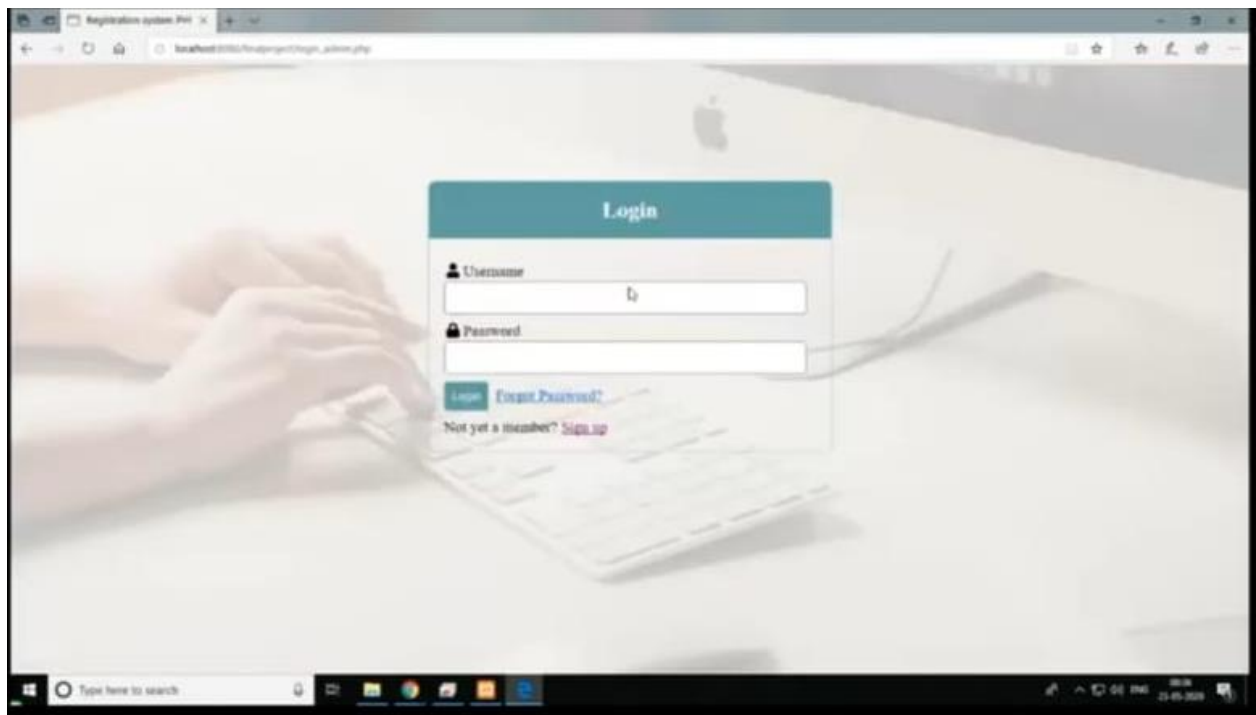
- Komal Raut, Nimesh Shah, Akash Thorat, “ Food donation portal”
<http://ijarcet.org/wpcontent/uploads/IJARCET-VOL-5-ISSUE-4-906-908.pdf>
- Dhruvi Shah, Adnan Ansari, Ruchi Sharma, “ Helping Hands”
<http://ijsrd.com/Article.php?manuscript=IJSRDV4I110485> III.
- Hitesh Raut, Swapnil Rajput, Danjhan Nalavade, “Smartphone based food supply chain for Aurangabad city using GIS location based and google web services”
<https://ieeexplore.ieee.org/document/7580874/metrics> IV.
- Issac Nuamah, Lauren Davis, Steven Jiang, “ Predicting donations of forecasting simulation model” <https://dl.acm.org/citation.cfm?id=2888832>
- Aaron Ciaght, Adolfo Villafiorita, “ Beyond food sharing: Supporting food wastage reduction using ICT” <http://esatjournals.net/ijret/2016v05/i04/IJRET20160504058.pdf>
- Amir Saxena. Khushi Verma, Aadit Patil, “ Development of a food supply chain by PHP” <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2610113/>
https://www.mecknc.gov/LUESA/SolidWaste/BusinessRecycling/Documents/WOW_FoodWasteBook_v2.pdf VIII. <http://www.eesc.europa.eu/en/agenda/our-events/events/food-donation-fighting-food-poverty-and-addressingfood-waste#download>

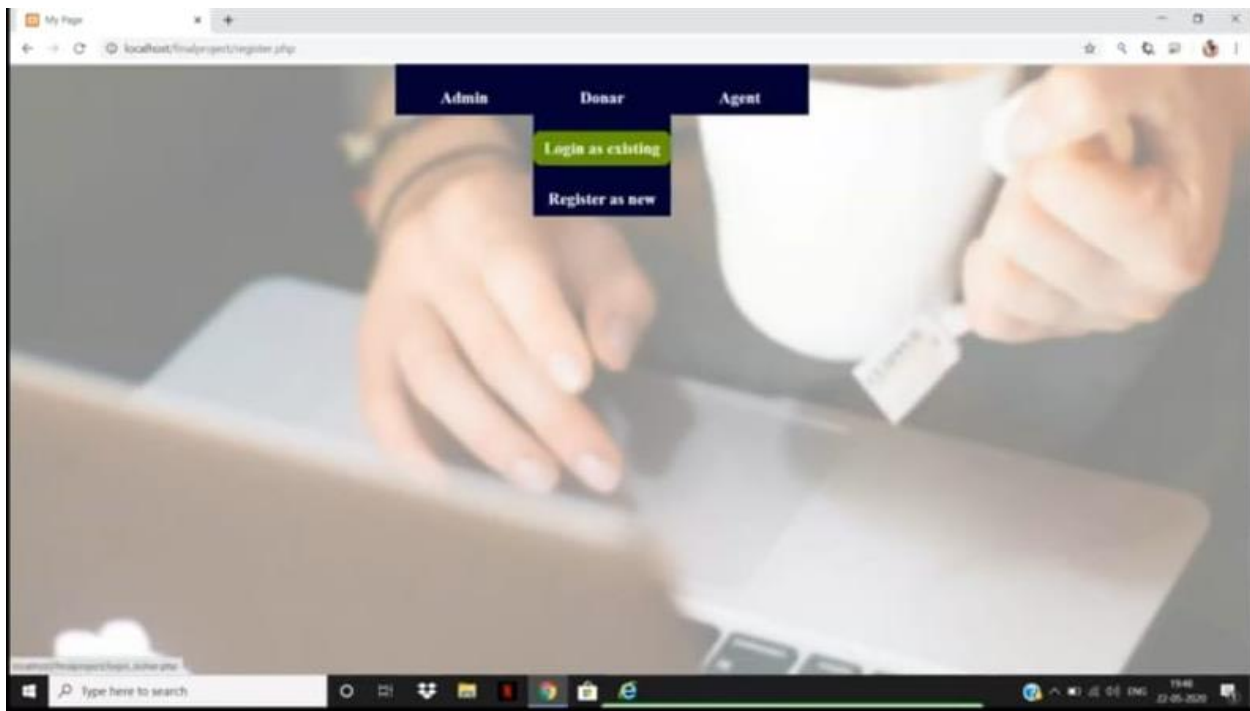
APPENDIX:

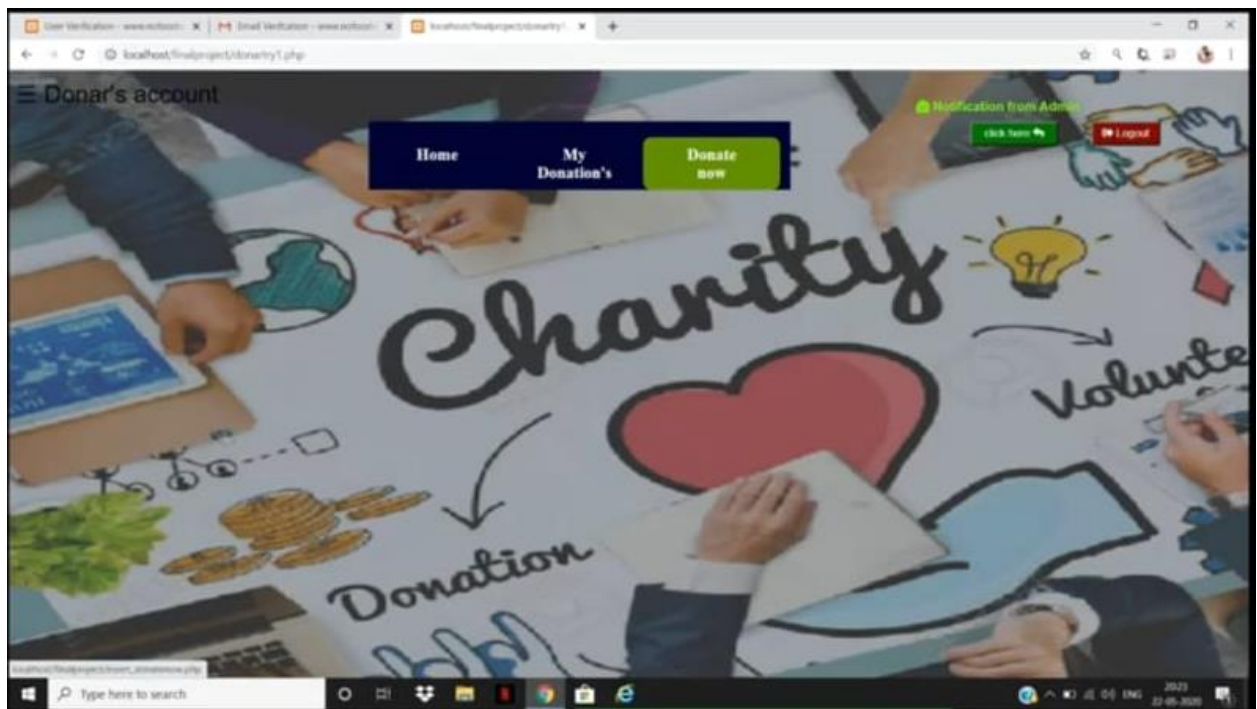
A.SCREENSHOTS

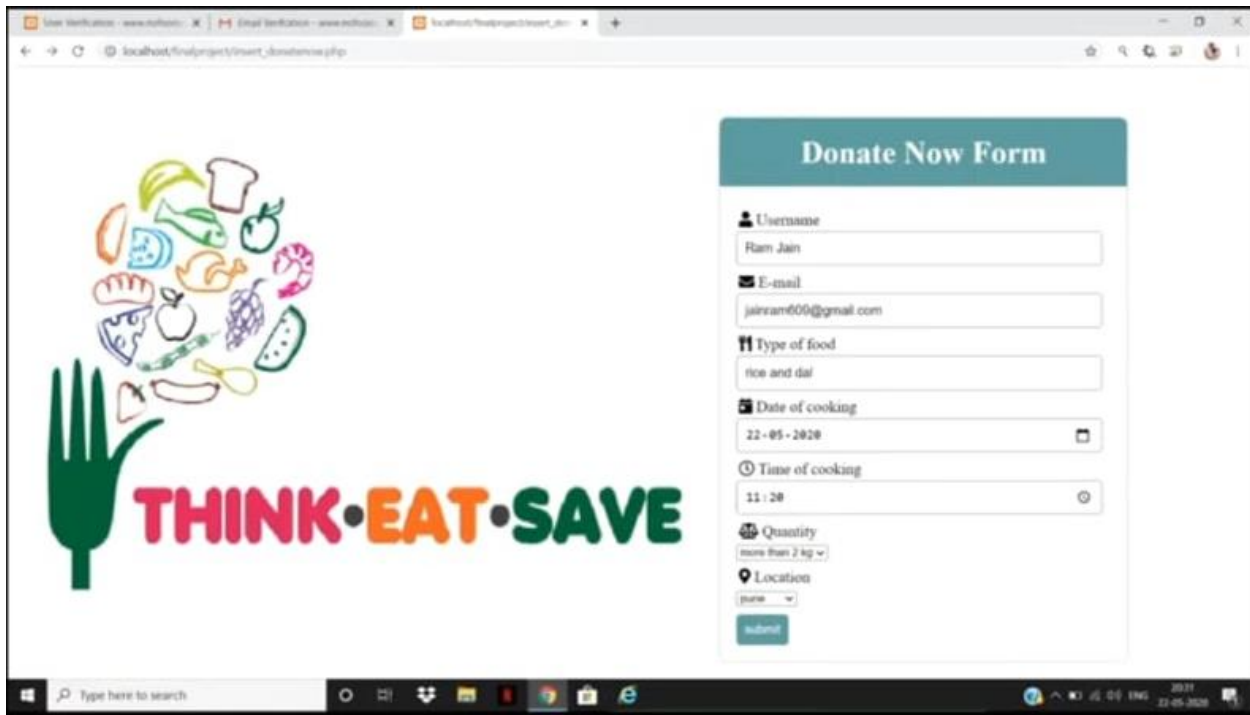


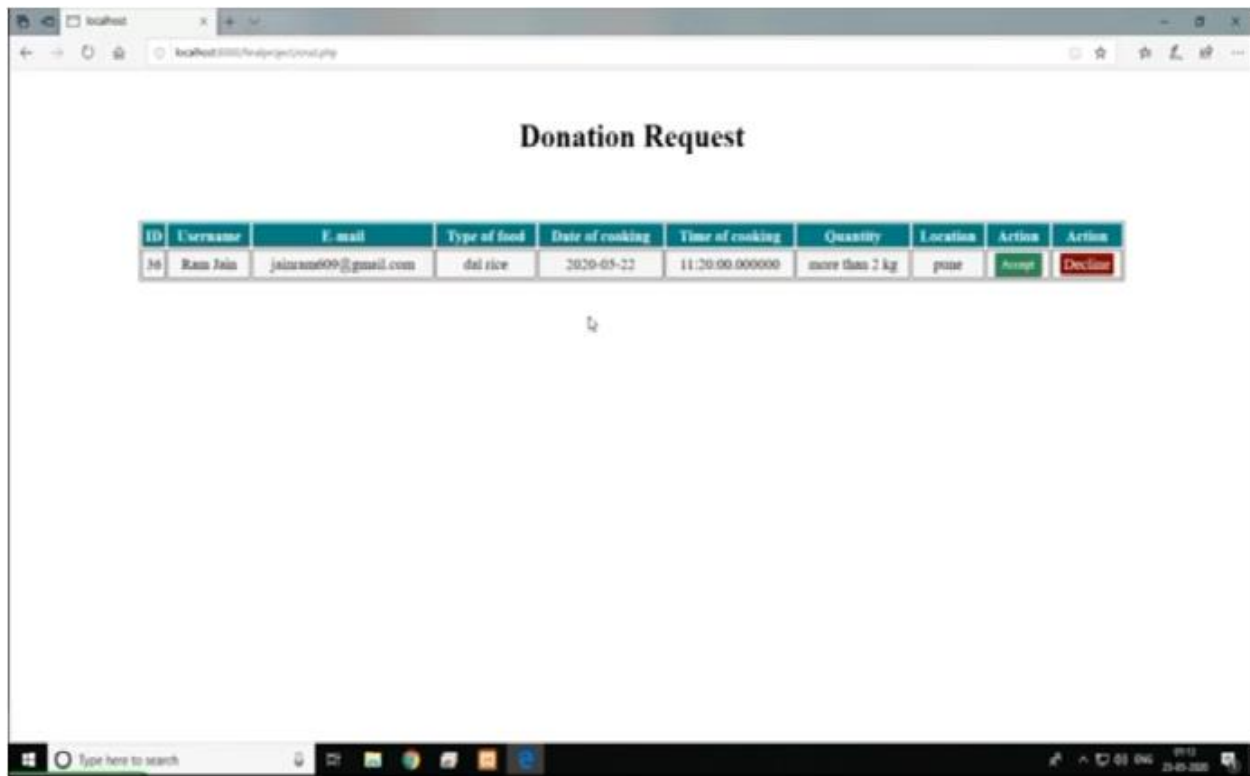


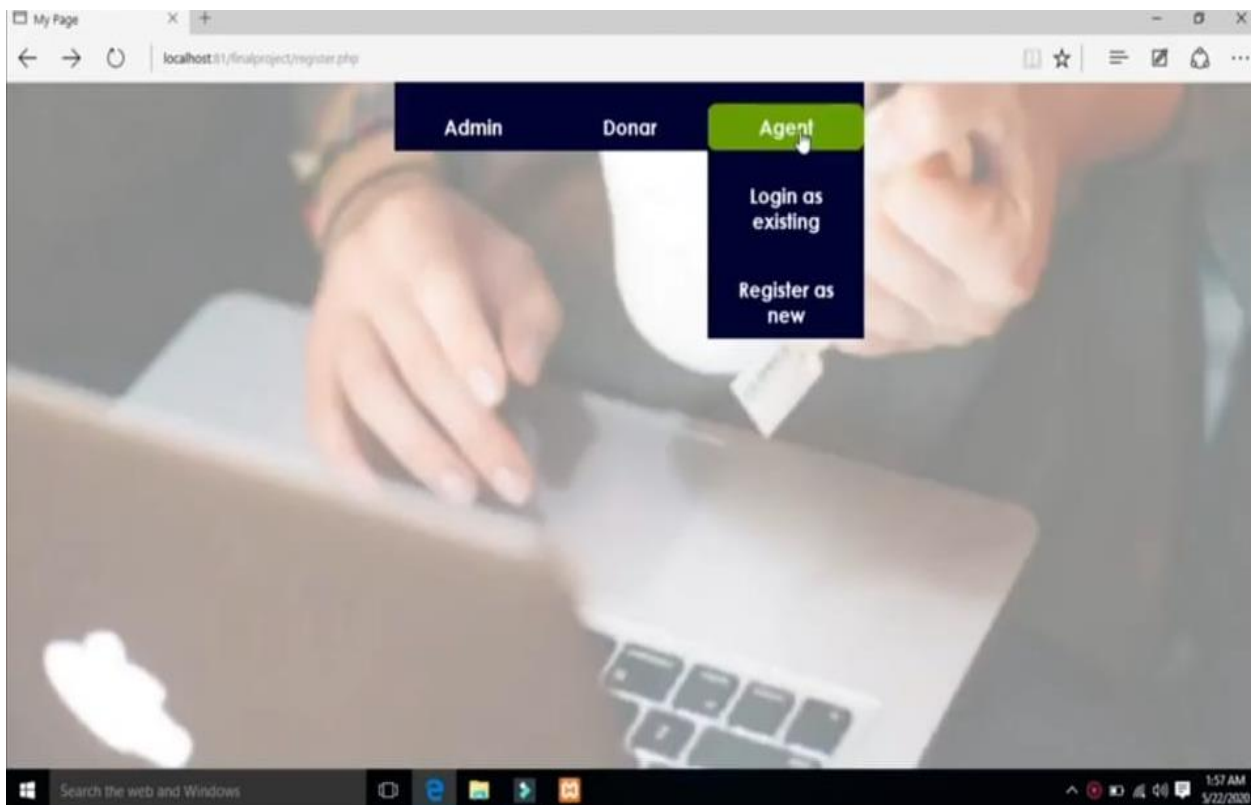
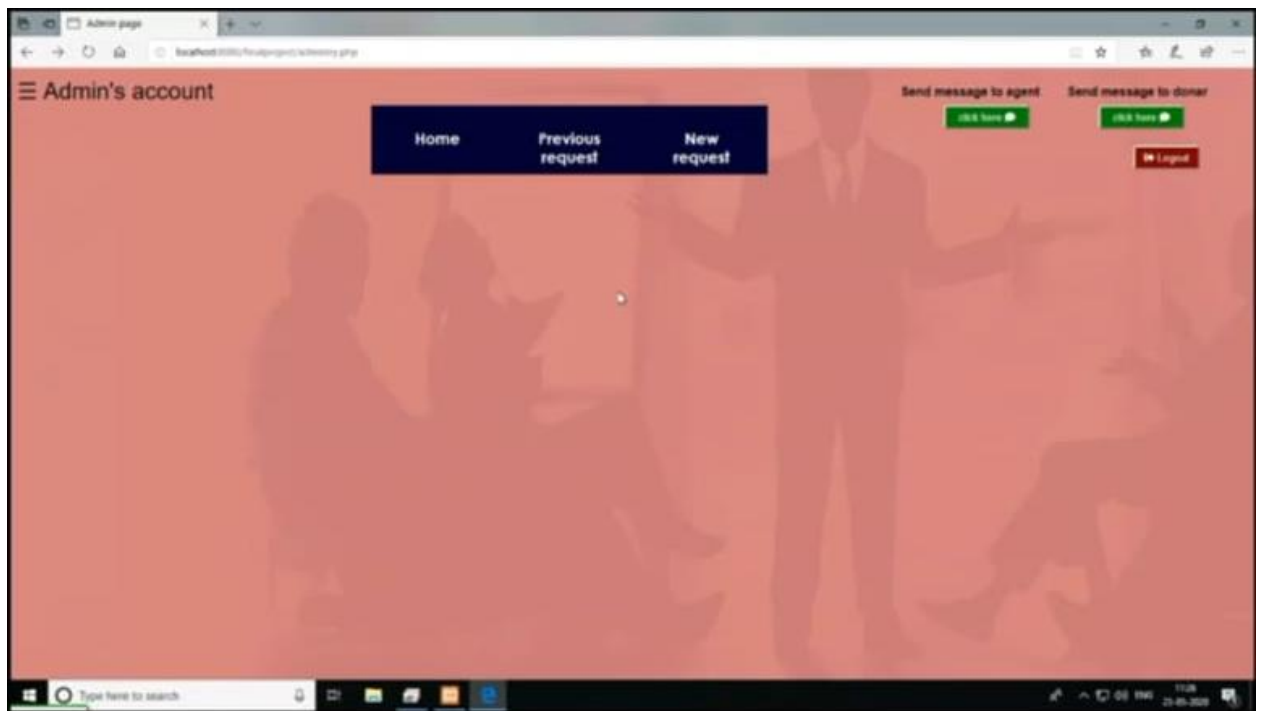












User Verification - www.nofb | Email Verification - www.nofb | Agent page

localhost:81/finalproject/agenttry.php

Agent's account

Notification from Admin

click here Logout

Home My donation record Donation form

```

graph LR
    Donar[Donar] --> Admin[Admin]
    Admin --> Agent[Agent]
    Agent --> Collect[Goes to donar and collects the food]
    Collect --> Supply[The food is supplied to Orphanage by agent]
  
```

http://localhost:81/finalproject/mydonation_agent.php

localhost

localhost:8080/finalproject/donate_nowback.php

Previously Donated Food

ID	Username	E-mail	Type of food	Date of cooking	Time of cooking	Quantity	Location
35	Ram Jain	jainram609@gmail.com	dal rice	2020-05-22	11:20:00.000000	more than 2 kg	pune

localhost

localhost:8080/foodproject/agent_donation_food.php

Food Collecting Details

id	Agent Name	E-mail	Donor's Name	Date of collecting food	Time of collecting food	Orphanage Name	Location
27	Shruti Shah	shruti.s0215@gmail.com	Ram Jain	2020-09-22	13:00:00.000000	Spash Baigram	pune

Type here to search

11:46 23-09-2020

Admin's account

Home Previous request New request

Send message to agent [click here](#)

Send message to donar [click here](#)

[Logout](#)

https://localhost:8080/foodproject/admin.php

Type here to search

11:51 23-09-2020

B.SOURCE CODE:

```
package
com.android.anoop.foodwa
stemanagement;

import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
public class MainActivity extends AppCompatActivity {
    Button
    mButtonGuestLogin,mButtonAgentLogin,mButtonCheckForFood,m
    ButtonSubmitFoodDetails,mButtonLogout;
    SharedPreferences sp;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //find button ids
        mButtonAgentLogin =
        (Button)findViewById(R.id.mainActivityButtonAgentLogin);
        mButtonGuestLogin =
        (Button)findViewById(R.id.mainActivityButtonGuestLogin);
        mButtonCheckForFood =
        (Button)findViewById(R.id.mainActivityButtonCheckForFood);
        mButtonSubmitFoodDetails =
        (Button)findViewById(R.id.mainActivityButtonSubmitFoodDetails);
        mButtonLogout =
        (Button)findViewById(R.id.mainActivityButtonLogout);
        sp = getSharedPreferences("userDetails",MODE_PRIVATE);
        String whoLogin=sp.getString("whoLogin","N");
        if(whoLogin.equals("guest")){
            mButtonGuestLogin.setVisibility(View.GONE);
            mButtonAgentLogin.setVisibility(View.GONE);
            mButtonCheckForFood.setVisibility(View.GONE);
            mButtonSubmitFoodDetails.setVisibility(View.VISIBLE);
            mButtonLogout.setVisibility(View.VISIBLE);
        }else if(whoLogin.equals("agent")){
            mButtonGuestLogin.setVisibility(View.GONE);
            mButtonAgentLogin.setVisibility(View.GONE);
```

```

        mButtonCheckForFood.setVisibility(View.VISIBLE);
        mButtonSubmitFoodDetails.setVisibility(View.GONE);
        mButtonLogout.setVisibility(View.VISIBLE);
    }else{
        mButtonGuestLogin.setVisibility(View.VISIBLE);
        mButtonAgentLogin.setVisibility(View.VISIBLE);
        mButtonCheckForFood.setVisibility(View.GONE);
        mButtonSubmitFoodDetails.setVisibility(View.GONE);
        mButtonLogout.setVisibility(View.GONE);
    }
}

//go to guest login page
public void goToGuestLoginPage(View view){
    Intent intent = new Intent(this,GuestLogin.class);
    startActivity(intent);
}

//go to agent login page
public void goToAgentLoginPage(View view){
    Intent intent = new Intent(this,AgentLogin.class);
    startActivity(intent);
}

//go to SearchFoodByAgent Activity
public void goToSearchFoodByAgent(View view){
    Intent intent = new Intent(this,SearchFoodByAgent.class);
    startActivity(intent);
}

//go to SubmitFoodDetailsByGuest Activity
public void goToSubmitFoodDetailsByGuest(View view){
    Intent intent = new
Intent(this,SubmitFoodDetailsByGuest.class);
    startActivity(intent);
}

//Guest or Agent logout method
public void logout(View view){
    SharedPreferences.Editor spe = sp.edit();
    spe.putString("whoLogin","N");
    spe.commit();
    Intent intent = new Intent(this,MainActivity.class);
    startActivity(intent);
}
}

```