A PROJECT REPORT BY

TEAM NO. 1

RITIKA SINGH (E23CSEU0751)

JAI KUMAR SRIVASTAVA (E23CSEU0753)

VANSHIKA AGARWAL (E23CSEU755)

PRANJAL SINGH(E23CSEU0763)



SUBMITTED TO

SCHOOL OF COMPUTER SCIENCE ENGINEERING AND TECHNOLOGY, BENNETT UNIVERSITY

GREATER NOIDA, 201310, UTTAR PRADESH, INDIA

April 2025

# DECLARATION

I/We hereby declare that the work which is being presented in the report entitled “FOODSAVER”, is an authentic record of my/our own work carried out during the period from JAN, 2023 to April, 2023 at School of Computer Science and Engineering and Technology, Bennett University Greater Noida.

The matters and the results presented in this report has not been submitted by me/us for the award of any other degree elsewhere.

Signature of Candidate

Ritika Singh

(Enroll. No. E23CSEU0751)

Jai Kumar Srivastava

(Enroll. No. E23CSEU0753)

Vanshika Agarwal

(Enroll. No. E23CSEU0755)

Pranjal Singh

(Enroll. No. E23CSEU0763)

# ACKNOWLEDGEMENT

We would like to take this opportunity to express my/our deepest gratitude to our mentor, **Ms. Sanchali Das** for guiding, supporting, and helping me/us in every possible way. We were extremely fortunate to have him as our mentor as she provided insightful solutions to problems faced by us thus contributing immensely towards the completion of this capstone project. We would also like to express our deepest gratitude to VC, DEAN, HOD, faculty members and friends who helped us in successful completion of this capstone project.

Signature of Candidate

Ritika Singh

(Enroll. No. E23CSEU0751)

Jai Kumar Srivastava

(Enroll. No. E23CSEU0753)

Vanshika Agarwal

(Enroll. No. E23CSEU0755)

Pranjal Singh

(Enroll. No. E23CSEU0763)

## TABLE OF CONTENTS

<<Right click on the heading and click update flied. Your heading will be pulled here.

Same goes with all the table of content in the subsequent pages>>

LIST OF TABLES ......................................................................................................................... vi

LIST OF FIGURES ...................................................................................................................... vii

LIST OF ABBREVIATIONS ...................................................................................................... viii ABSTRACT ................................................................................................................................... ix

1. INTRODUCTION ...................................................................................................................... 1
   1. Problem Statement ............................................................................................................... 1
2. Background Research ................................................................................................................. 2
   1. Proposed System .................................................................................................................. 4
   2. Goals and Objectives ............................................................................................................ 4
3. Project Planning .......................................................................................................................... 5
   1. Project Lifecycle .................................................................................................................. 6
   2. Project Setup ........................................................................................................................ 6
   3. Stakeholders ......................................................................................................................... 7
   4. Project Resources ................................................................................................................. 7
   5. Assumptions ......................................................................................................................... 7
4. Project Tracking .......................................................................................................................... 9
   1. Tracking ............................................................................................................................... 9
   2. Communication Plan ............................................................................................................ 9
   3. Deliverables ........................................................................................................................ 10
5. SYSTEM ANALYSIS AND DESIGN ..................................................................................... 12
   1. Overall Description ............................................................................................................ 12
   2. Users and Roles .................................................................................................................. 13
   3. Design diagrams/ UML diagrams/ Flow Charts/ E-R diagrams ........................................ 14
      1. Use Case Diagrams ...................................................................................................... 15
      2. Class Diagram ............................................................................................................. 16
      3. Activity Diagrams ....................................................................................................... 17
      4. Sequence Diagram ....................................................................................................... 18
      5. Data Architecture ......................................................................................................... 19
6. User Interface ............................................................................................................................ 20
   1. UI Description .................................................................................................................... 20
   2. UI Mockup ......................................................................................................................... 20
7. Algorithms/Pseudo Code .......................................................................................................... 21
8. Project Closure .......................................................................................................................... 22
   1. Goals / Vision ..................................................................................................................... 22
   2. Delivered Solution .............................................................................................................. 22
   3. Remaining Work ................................................................................................................ 23

REFERENCES ............................................................................................................................. 24

**LIST OF TABLES**

Table Page

Table 1: Goal and Objectives .......................................................................................................... 5 Table 2: Sample 2 .......................................................................... **Error! Bookmark not defined.** Table 3: Sample 3 ........................................................................................................................... 7 Table 4: Sample 4 .......................................................................... **Error! Bookmark not defined.**

Table 5: Sample 4 .......................................................................... **Error! Bookmark not defined.**

Table 6: Sample 6 ........................................................................................................................... 9

Table 7: Regularly Scheduled Meetings ......................................................................................... 9

Table 8: Information To Be Shared Within Our Group ................................................................ 10

Table 9: Information To Be Provided To Other Groups ............................................................... 10

Table 10: Information Needed From Other Groups ..................................................................... 10 Table 11: Deliverables .................................................................................................................. 10 Table 12: Sample 12 ...................................................................... **Error! Bookmark not defined.**

## LIST OF FIGURES

Figure Page

Figure 1: Sample use-case diagram .............................................................................................. 16

Figure 2: sample 2 ......................................................................................................................... 17

Figure 3: sample 3 ......................................................................................................................... 18

Figure 4: Sample 4 ........................................................................................................................ 19

Figure 5: Sample 5 ........................................................................................................................ 19 Figure 6: Sample 6 ........................................................................................................................ 20

## LIST OF ABBREVIATIONS

Abbreviation ..................................................Explanation of the Abbreviation

AAA Authentication Authorization and Access Control

CSP Cloud Service Provider

DNS Domain Name System

IAM Identity and Access Management

## ABSTRACT

Food wastage is a critical global challenge, with a significant portion stemming from households discarding items that go unnoticed until expiration. This issue not only contributes to environmental degradation but also affects household budgets and food security. To address this pressing concern, **FoodSaver** has been developed as a comprehensive mobile application aimed at minimizing food waste through smart expiry tracking and enhanced user engagement.

**FoodSaver** enables users to efficiently manage their pantry by allowing multiple modes of input— manual entry, receipt scanning, and image-based detection—to log perishable food items. Once stored, the app automatically tracks expiration dates and sends timely alerts and notifications to the user, ensuring that food is consumed before it spoils. This proactive approach not only saves money but also fosters a more sustainable lifestyle by reducing unnecessary food disposal.

Beyond expiry tracking, **FoodSaver** differentiates itself by offering a powerful feature: **Reward points system**. The reward points system gamifies the experience—users accumulate points for actions such as consuming items before expiry, regularly updating their inventory, and using recipes provided by the app. These points can later be redeemed for incentives, making the app both practical and enjoyable.

The application leverages cloud storage for data backup and synchronization, ensuring that users have seamless access to their inventory across devices. It also features a user-friendly interface designed for minimal input and maximum efficiency. With intuitive navigation, smart notifications, and meaningful visual insights, **FoodSaver** aims to empower users to take control of their food consumption habits.

This report details the entire development lifecycle of **FoodSaver**—from initial problem identification and background research to system design, UI/UX development, and final implementation. It highlights how the integration of modern mobile technologies with sustainability-driven features can not only address real-world issues but also enhance user experience and behavioral change. Through **FoodSaver**, we hope to inspire smarter food management practices and contribute to a reduction in household food wastage at scale.

# 1. INTRODUCTION

Food wastage is a growing concern globally, with a large portion occurring in households due to poor tracking of expiration dates. This not only leads to economic loss but also contributes significantly to environmental issues. In response to this problem, we developed **Foodly**, a mobile application designed to help users manage their kitchen inventory more effectively.

**FoodSaver** allows users to add food items through manual entry, receipt scanning, or photos, and then tracks their expiry dates, sending timely reminders to prevent spoilage. What makes **Foodly** unique is its integration **reward points system** that encourages regular use and sustainable habits.

The app combines a user-friendly interface, smart notifications, and cloud-based storage to provide a seamless experience aimed at reducing food waste, improving household budgeting, and promoting responsible consumption. This report explores the background, design, and development of **FoodSaver** , highlighting its potential impact on everyday food management.

## 1.1. Problem Statement

In many households, a significant amount of food is wasted due to a lack of awareness or tracking of expiration dates. Perishable items often get forgotten in refrigerators or shelves, leading to spoilage and unnecessary disposal. This not only results in financial loss for families but also contributes to larger environmental issues, such as increased carbon emissions from food waste.

Despite the availability of various grocery and cooking apps, there is a noticeable gap when it comes to dedicated solutions that help users monitor food expiry, suggest ways to use ingredients efficiently, and encourage responsible consumption habits. There is a need for a smart, user friendly tool that not only tracks food expiry but also helps users reduce waste through timely alerts, and motivation through gamification.

# 2. BACKGROUND RESEARCH

## Understanding the Problem of Food Waste

Food waste is a widespread issue affecting both developed and developing nations. According to the Food and Agriculture Organization (FAO) of the United Nations, nearly **one-third of all food produced globally is wasted**, amounting to around 1.3 billion tonnes per year (FAO, 2013). In households, much of this waste occurs because of poor storage management and failure to consume items before their expiration dates. Research indicates that lack of awareness and inadequate tools for tracking expiry dates are key contributors to this problem (Koivupuro et al., 2012).

The environmental impact of food waste is substantial. Rotting food in landfills emits methane, a potent greenhouse gas. Moreover, the resources used in producing, packaging, and transporting wasted food—including water, energy, and labor—are also lost in the process (Gustavsson et al., 2011).

## Existing Technological Solutions

Several mobile applications and digital solutions have emerged in recent years to tackle household food management. Apps like **Too Good To Go**, **OLIO**, and **Fridge Pal** provide various services, such as rescuing surplus food from stores or helping users track items in their fridge. However, most of these apps focus either on surplus food redistribution or inventory tracking, without combining **personal expiry reminders** and **reward systems** to drive user engagement and habit change.

A study by De Hooge et al. (2018) on consumer behavior around food waste emphasized the importance of timely interventions, such as notifications, that remind users before items expire. However, the study also noted that simple reminders are often not enough to influence long-term habits unless coupled with **motivational features** like gamification.

## Gamification and Behavioral Change

By implementing a **reward points system**, **FoodSaver** seeks to leverage these behavioral insights. Users earn points by consuming food before expiry, updating their inventory, and using recipe suggestions. This gamified approach is designed to encourage consistent engagement and reduce food waste over time.

## Motivation for the Project

After reviewing various sources, it became clear that existing solutions often address food waste in isolated ways—either by helping businesses reduce waste, offering food-sharing platforms, or enabling basic home inventory tracking. However, few tools take a **comprehensive, usercentric approach** that includes **expiry tracking**, and **behavioral incentives**.

The motivation to develop **FoodSaver** stemmed from this gap. By combining practical features with a smooth, intuitive interface, the project aims to empower everyday users to manage their food smarter, save money, and contribute to sustainability. The literature strongly supports the need for such a tool, and the current lack of integrated solutions creates a valuable opportunity for innovation.

## References

* FAO. (2013). *Food wastage footprint: Impacts on natural resources.* Food and Agriculture Organization of the United Nations.
* Gustavsson, J., et al. (2011). *Global food losses and food waste – Extent, causes and prevention.* FAO.
* Koivupuro, H.-K., et al. (2012). *Influence of socio-demographic, behavioural and attitudinal factors on the amount of avoidable food waste.* Waste Management, 32(11), 2241-2249.
* De Hooge, I. E., et al. (2018). *Reducing household food waste: A meta-analytic review of intervention studies.* Journal of Cleaner Production, 210, 1725-1738.
* Johnson, D., et al. (2017). *Gamification for sustainable behavior change: Leveraging motivational affordances in mobile apps.* Environment Systems and Decisions, 37(3), 306–320.
* Papargyropoulou, E., et al. (2014). *The food waste hierarchy as a framework for the management of food surplus and food waste.* Journal of Cleaner Production, 76, 106–115.

## 2.1. Proposed System

This project aims to reduce household food waste by helping users track and manage the expiration dates of food items through a mobile application. By combining expiry tracking with a reward-based gamification system, we intend to encourage users to consume food more efficiently and mindfully.

The vision of **FoodSaver** is to transform how people manage their food at home—reducing waste, saving money, and promoting sustainable consumption habits. By providing smart reminders and actionable suggestions, the app empowers users to make better decisions around food usage. In doing so, **FoodSaver** contributes to the broader goals of environmental sustainability and responsible living.

## 2.2. Goals and Objectives

The primary goal of the Foodly project is to develop a user-friendly mobile application that helps individuals and households reduce food waste by tracking expiration dates and encouraging efficient food usage.

**Project Goals:**

* **Product Goal:** Develop and launch a fully functional mobile app (Android) that enables users to:
  + Add food items via manual entry, barcode scan, or image upload.
  + Receive expiry alerts at customizable intervals.
  + Earn and track reward points for reducing food waste.
* **Process Goal:** Complete the design, development, and testing phases of the app within 8 weeks using Agile methodology.
* **Quality Goal:** Achieve a minimum user satisfaction rating of 85% in beta testing based on usability, performance, and usefulness of features.
* **Teamwork Goal:** Ensure consistent collaboration and contribution from all team members, with weekly sprint reviews and division of responsibilities for design, development, and testing.

**Project Objectives:**

* Reduce food waste in participating households by at least **20%** (as self-reported in user surveys during beta testing).
* Integrate **at least 3 core features**: expiry tracking, and a points-based reward system.
* Maintain app crash rate below **1%** during test and release phases.
* Onboard at least **50 users** during the initial testing phase to collect meaningful feedback.

### Table 1: Goal and Objectives

### Make the system extensible – future updates like new recipe modules or additional tracking features can be implemented easily.

### Make the system easy to support – include clear documentation, setup guides, and a user/admin manual.

### Make the system very easy to use – ensure users can navigate the app with little to no training. Build a prototype that demonstrates the user interface by [insert target date], to collect early feedback from users.

### Have fun working on the project while fostering innovation and collaboration among team members.

# 3. PROJECT PLANNING

The development of *FoodSaver* follows the **Agile development lifecycle**, chosen for its iterative approach and flexibility, allowing continuous improvement and user feedback integration throughout the project.

**Stakeholders** include the project owner, developers, a UI/UX designer, QA tester, and end users who will benefit from food expiry tracking and recipe suggestions.

**Resources required**:

* *Software:* Android Studio, Firebase, Figma, GitHub
* *Human:* 1 project manager, 2 developers, 1 designer, 1 tester
* *Hardware:* Laptops/desktops, Android devices/emulators

**Assumptions made**:

* Users have internet access and are willing to input food data
* Firebase will reliably handle backend tasks
* Recipes will be sourced from a reliable database or API

## 3.1. Project Lifecycle

The **FoodSaver** project will follow an **Agile methodology**, specifically a **SCRUM-like approach**. At the beginning of the project, the team will gather initial requirements and outline a high-level development plan. The implementation will take place over **three main iterations (sprints)**, each focusing on developing and refining key features such as food expiry tracking, and user authentication.

Each sprint will last approximately **2 weeks**, with regular team meetings to discuss progress, challenges, and priorities. This iterative process allows for continuous improvement, flexibility in responding to feedback, and ensures the final product meets user needs effectively

## 3.2. Project Setup

### # Decision Description

1. Android OS, Python, Firebase for backend, Streamlit for frontend Git for version control
2. Google Material Design guidelines for UI, Firebase security rules, clean code principles
3. No special access or NDA required; app may be released as open source in the future
4. Firebase hosting and real-time database used for testing and deployment environments

## 3.3. Stakeholders

<< Identify all stakeholders for this project (groups or individuals that are affected by or are in some way accountable for the outcome of the project – business managers, end users, developers, testers, support people, instructors, etc.) >> **Example:**

### Table 2: Sample 3

|  |  |  |  |
| --- | --- | --- | --- |
| **Stakeholder** |  |  | **Role** |
| Jai Kr. Srivastava | Backend |  |  |
| Ritika Singh | Backend and Frontend |  |  |
| Vanshika Agarwal | Frontend |  |  |
| Pranjal Singh | Research and Development |  |  |
|  |  |  |  |
|  |  |  |  |

## 3.4. Project Resources

|  |  |
| --- | --- |
| **Resource** | **Resource Description Quantity** |
| Development Team | Team members responsible for coding, testing, and deploying the  4 application. |
| Project Mentor | A mentor who provides technical and project guidance. 1 |
| Android Studio | IDE used for developing the Android application. 1 |
| Firebase | Backend service used for authentication, data storage, and hosting. 1 |
| Figma | Tool used for designing UI/UX wireframes and prototypes. 1 |
| GitHub | Platform for version control and collaboration. 1 |
| Android Phones | Devices used for testing the app on physical hardware. 2 |
|  | Development machines used by the team members. 4  **3.5. Assumptions**     |  |  | | --- | --- | | # | Assumptions | | A1 | Firebase and third-party APIs (e.g., for recipes) will be available and stable throughout the project | | A2 | Team members will be able to quickly familiarize themselves with Android Studio and Firebase | | A3 | All planned features (expiry tracking, notifications, recipe suggestions) can be implemented within the project timeline. | | A4 | The app will be tested effectively using emulators and Android devices available to the team. | | A5 | Users will have internet access to utilize Firebase features and access recipe suggestions. | | A6 | No major changes will be required to the project scope after the initial planning phase | | A7 |  |   The following assumptions were made while planning and executing the **FoodSaver** project. These assumptions guided the scheduling, resource allocation, and development methodology throughout the semester.  Table 5: Project Assumptions |

# 4. PROJECT TRACKING

## 4.1. Tracking

<< Provide information about how the project was tracked and where information was kept. This should include information such as what type of source control was being used and how it can be accessed, any bug-tracking system that was used for the project and where it can be accessed, what type of regressing testing suite was used and where it can be accessed, and any similar information that provides details on the project’s status, etc. >>

**Example:**

### Table 3: Sample 6

|  |  |  |
| --- | --- | --- |
| **Information** | **Description** | **Link** |
| Code Storage | Project code will be stored in SVN repository. | Link |
| Bug Tracking | Bugs were manually tracked and resolved during team meetings. | Link |
| Project  Documents and  Assignments | Weekly reports, design diagrams, and sprint goals were stored. | Link |
| Continuous Integration | | |  | | --- | | Manual integration done after each sprint. | | N/A |
| Regression Testing | Regression and functional tests were performed manually. | N/A |

## 4.2. Communication Plan

Communication throughout the *FOODLY* project was carefully structured to ensure all team members and stakeholders were consistently aligned. This included scheduled meetings, defined communication tools, and deliverables handled by the four core team members.

### Table 4: Regularly Scheduled Meetings

|  |  |  |
| --- | --- | --- |
| Meeting Type | Frequency/Schedule | Who Attends |
| Team Meeting | Weekly | All Team Members |
| |  | | --- | | Class Check-ins | | Weekly in class | All Team Members, Instructor |
| Sprint Planning | Start of each sprint | All Team Members |
| Sprint Review Meeting | End of each sprint | Project team,Instructor |

### Table 5: Information To Be Shared Within Our Group

|  |  |  |  |
| --- | --- | --- | --- |
| **Who?** | **What Information?** | **When?** | **How?** |
| Project team | Task assignments & General scrum information | Weekly | WhatsApp |

### Table 6: Information To Be Provided To Other Groups

|  |  |  |  |
| --- | --- | --- | --- |
| Who? | What Information? | When? | How? |
| Instructor | Weekly updates & final report | Weekly / Final week | |  | | --- | | LMS submissions & DEMO in class |  |  | | --- | |  | |
| Instructor | |  | | --- | | Presentations |  |  | | --- | |  | | |  | | --- | | As scheduled |  |  | | --- | |  | | In-class & submission |

### Table 7: Information Needed From Other Groups

|  |  |  |  |
| --- | --- | --- | --- |
| Who? | What Information? | When? | How? |
| Sponsor and mentor | Requirement changes | Start of each sprint | Conference call or meeting with sponsor and mentor. |

## 4.3. Deliverables

<< Identify the major deliverables that this project is expected to produce. Assume the deliverables apply to all features or stories listed above unless indicated otherwise. Deliverables may include prototypes, documentation, software, etc. >>

### Table 8: Deliverables

|  |  |
| --- | --- |
| **#** | **Deliverable** |
| 1 | Background research and literature study |
| 2 | Manual and functional testing reports |
| 3 | System diagrams and flowcharts |
| 4 | Final Report, PowerPoint presentation, and 3-minute video |

# 5. SYSTEM ANALYSIS AND DESIGN

The *FOODSAVER* app is designed to help users manage food expiry, discover recipes, and earn rewards for reducing waste. The system is structured around key modules such as food tracking, recipe suggestions, and a rewards engine.



## *Functional Components*

* **User Authentication**: Secure login/registration.
* **Food Expiry Tracker**: Add food items and receive expiry alerts.
* **Reward System**: Points for using food before it expires.



## *System Design*

* **Architecture**: Follows MVC (Model-View-Controller) pattern.
* **Frontend**: Developed using Streamlit
* **Backend**: Firebase or Python
* **APIs**: Integrated with a third-party recipe API (e.g., Spoonacular).



## *User Interface*

• **Simple and intuitive UI** with screens for:

* Home (expiring items),
* Add Item
* Rewards,
* Settings/Profile.

## 5.1. Overall Description

**FOODSAVER** is a mobile application designed to address the growing issue of household food waste by leveraging intuitive food tracking, and gamified user incentives. The primary goal of the app is to help users manage their food inventory by logging food items and tracking their expiry dates in real time. FOODSAVER sends timely notifications to users before items expire, encouraging their use and reducing wastage.

Technically, the project is built using a cross-platform mobile development framework like Flutter or React Native for smooth performance across Android and iOS devices. Firebase serves as the backend for real-time database operations, authentication, and cloud storage. Users’ food inventory data is stored and synced in the cloud, enabling persistent access. The app implements a reward system, where users earn points by logging food items and marking them as "used" before expiration. These points accumulate in the user’s profile and can be converted into virtual badges or other in-app achievements, reinforcing sustainable habits.

The system architecture follows the Model-View-Controller (MVC) pattern to separate concerns and ensure maintainability. Unit testing is implemented using frameworks such as JUnit, and CI/CD is handled through GitHub Actions or Jenkins for automated testing and deployment. Overall, FOODSAVER combines food management, smart suggestions, and behavioral reinforcement to build a scalable solution that encourages users to form healthier and more sustainable food habits.

## 5.2. Users and Roles

The *FOODSAVER* system supports several different types of users and roles, each with unique responsibilities and interactions with the application. At the core are **end users**, who are everyday individuals or families seeking to manage their household food more efficiently. These users interact with the app to add food items, receive expiry alerts, view recipe suggestions based on available ingredients, and collect reward points for minimizing food waste. Their experience is primarily mobile-based, with an emphasis on ease-of-use and intuitive design.

Another key role is the **developer**, represented by project team members (Pranjal, Jai, Ritika, and Vanshika), who are responsible for building, testing, and maintaining the application. Developers work on front-end interfaces, back-end database interactions, integration of external recipe APIs, and implementation of features like notifications and reward tracking. They also ensure that the app follows proper coding standards, performance optimization, and data security.

Additionally, the **mentor** serves as an advisor throughout the development process. This individual provides feedback on project milestones, design decisions, and usability concerns, helping guide the technical and functional growth of the app. Optionally, the system may also include an **administrator** role who manages backend settings, monitors usage data, handles recipe database updates, and responds to user support issues.

Lastly, **automated agents**, such as background schedulers or notification services, play a vital role in the app's functionality. These agents run expiry checks, send push notifications to users about expiring food items, and trigger reward logic—all without direct user input. These automated tasks ensure that the pp remains responsive and helpful in real-time.

**5.2 Users and Roles**

**Table 12: Users and Roles**

|  |  |
| --- | --- |
| **User** | **Description** |
| Developer | |  | | --- | | A project team member (Pranjal, Jai, Ritika, Vanshika) responsible for implementing features like expiry tracking, recipe APIs, and reward logic. They also handle integration, testing, and maintaining the database. |  |  | | --- | |  | |
| **Mentor** | A guiding faculty or industry expert who reviews progress, provides feedback on technical implementation, and helps in refining user experience and system design. |
| **FOODSAVER**  **End User** | A regular user who uses the app to log food items, receive expiry alerts, view recipes, and collect rewards. They interact primarily through the mobile UI. |

**5.3. Design diagrams/Architecture/ UML diagrams/ Flow Charts/ E-R diagrams**

<<Provide all the design diagrams that were created during the design phase of your project. Those who love coding may use a tool https://liveuml.com/ to create UML diagrams. You may refer https://plantuml.com/ for commands/syntax of various diagrams. Some sample may be checked below:

|  |  |
| --- | --- |
| • | Use case: https://liveuml.com/view/6024c067492d8164f7df07a3 |
| • | Class: https://liveuml.com/view/6024c3e8492d8164f7df07a4 |

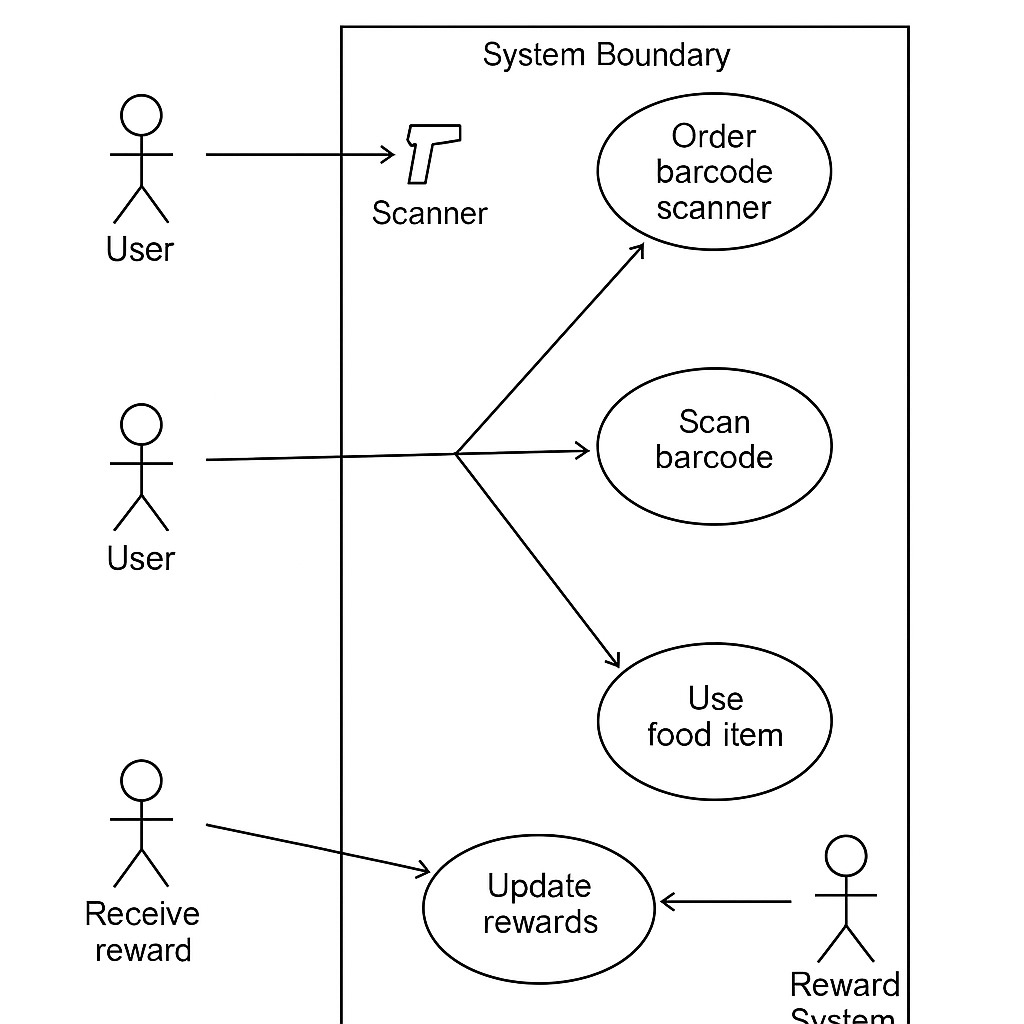
Some visual examples are in sections below:>>

**5.3.2. Architecture Diagram**

<< Provide the architecture diagram (may be a pattern selected) for your product>>.

### 5.3.3. Use Case Diagram

<< Provide any use-case diagrams that are being used as part of the project. Uniquely label each use case so that if necessary, it is easy to reference from other parts of the document. >>



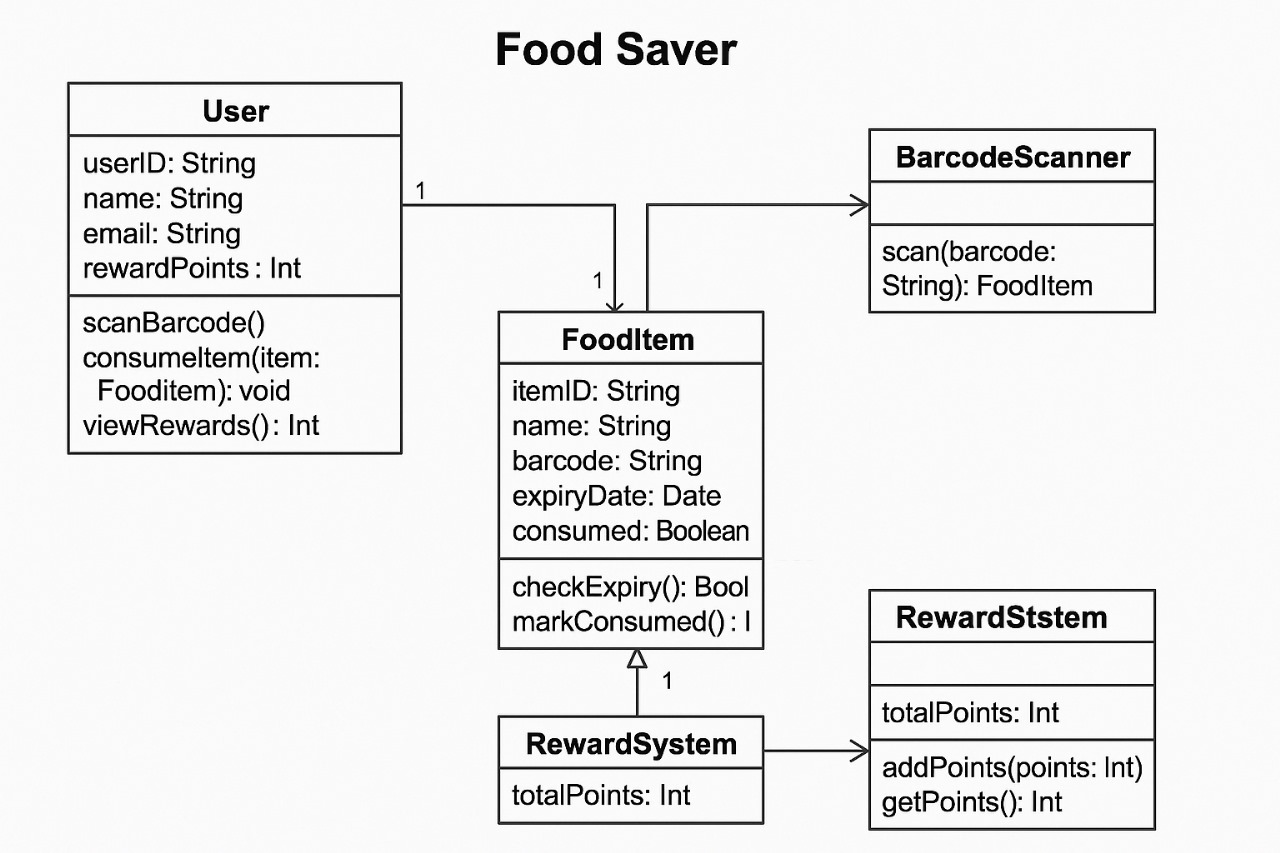
### Figure 1: Sample use-case diagram

<<to insert caption for tables and figures, click on references on the top menu, under captions section  click insert caption  choose label as table

If you use a table, figure, or non-text item that is not your original design, you must cite the original source of the item. You may use an in-text citation in the text of the title or caption of the item, or you may include the citation as a footnote. Refer to the style manual of your discipline for more information about citations of non-text items.>>

#### 5.3.4. Class Diagram

<< Include a class diagram for all classes to be designed. Optionally include major data elements of those classes and important methods and functions that will be used by other classes.>>

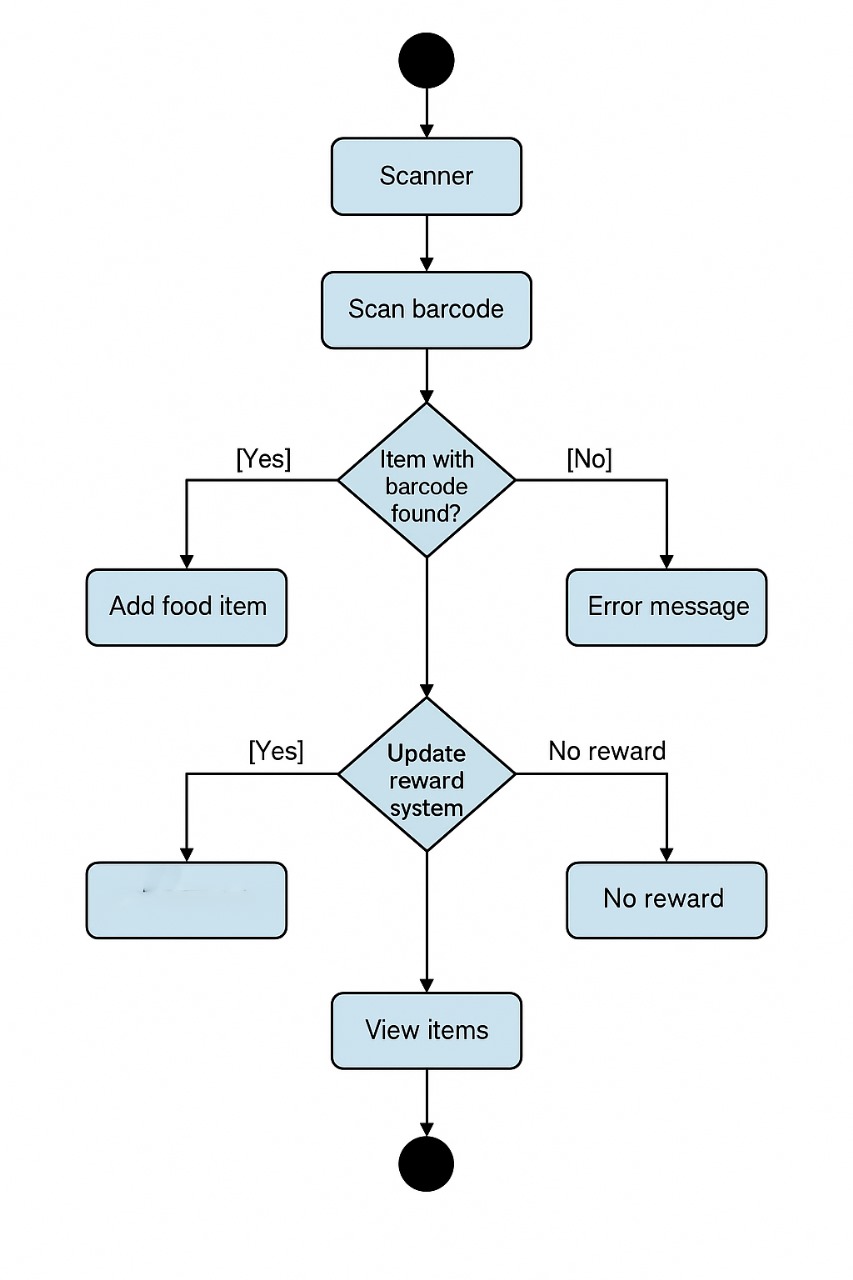


### Figure 2: sample 2

#### 5.3.5. Activity Diagrams

<< Include activity diagrams for important workflows in the program. At least one diagram should be included for the main workflow in the program. Optionally include labels that indicate which component is responsible for that part of the activity. Activity diagrams for components which perform complex tasks should also be included. >>

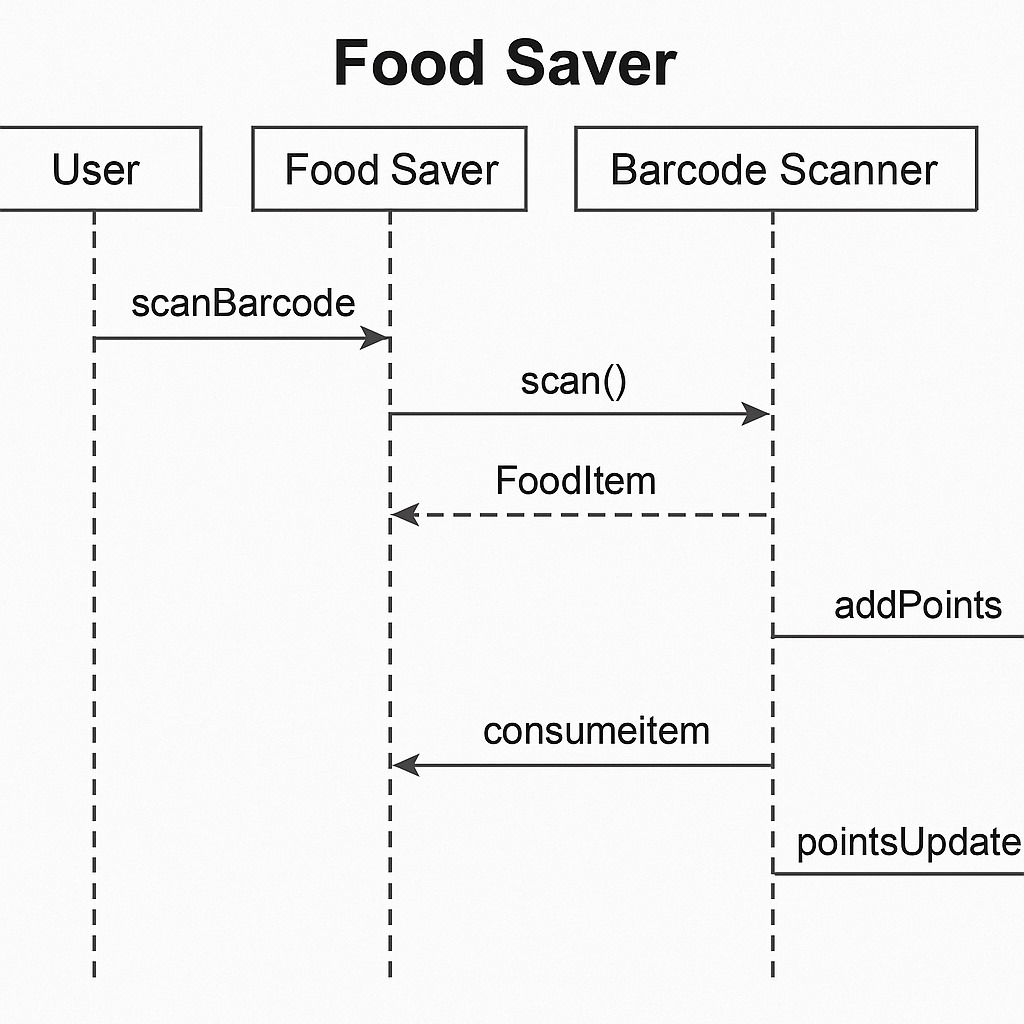
**Example**: View friend’s photo album



### Figure 3: sample 3

#### 5.3.6. Sequence Diagram

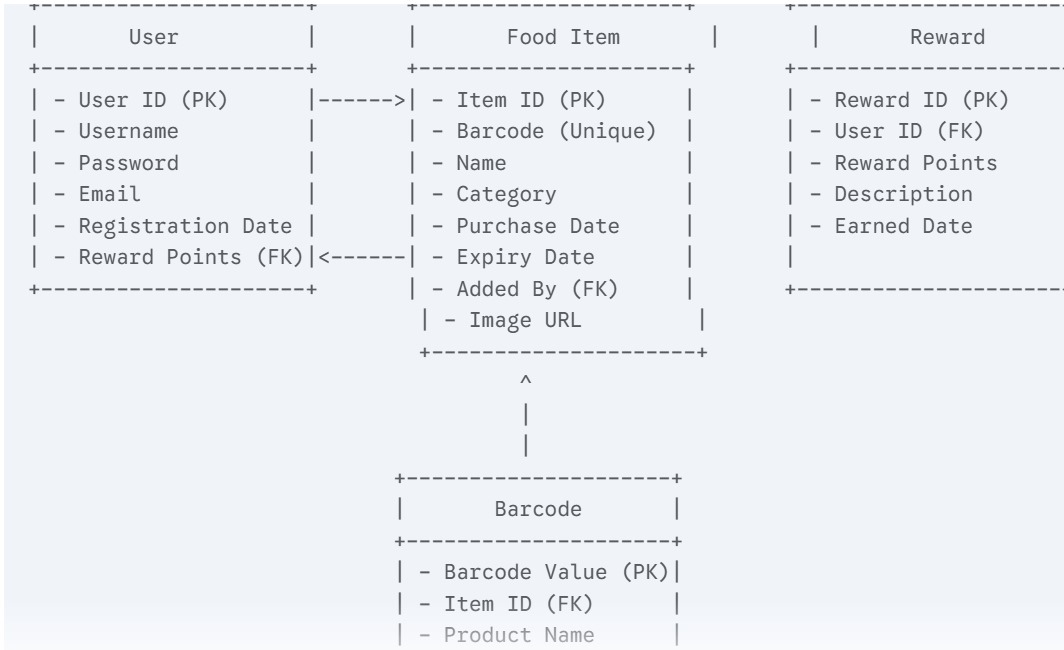
<< Include sequence diagrams for important functionality of the program to indicate control flow. These diagrams should include classes found in the class diagram and use the methods for those classes to show the interaction between them. >>



### Figure 4: Sample 4

#### 5.3.7. Data Architecture

<< Include any information or diagrams that provide details about databases, xml configuration files, or other data structures that are a part of the system. If a very specific format is required, it may be worthwhile to provide a more robust description or a detailed design such as a database schema. >>



**Figure 5: Sample 5**

# 6. USER INTERFACE

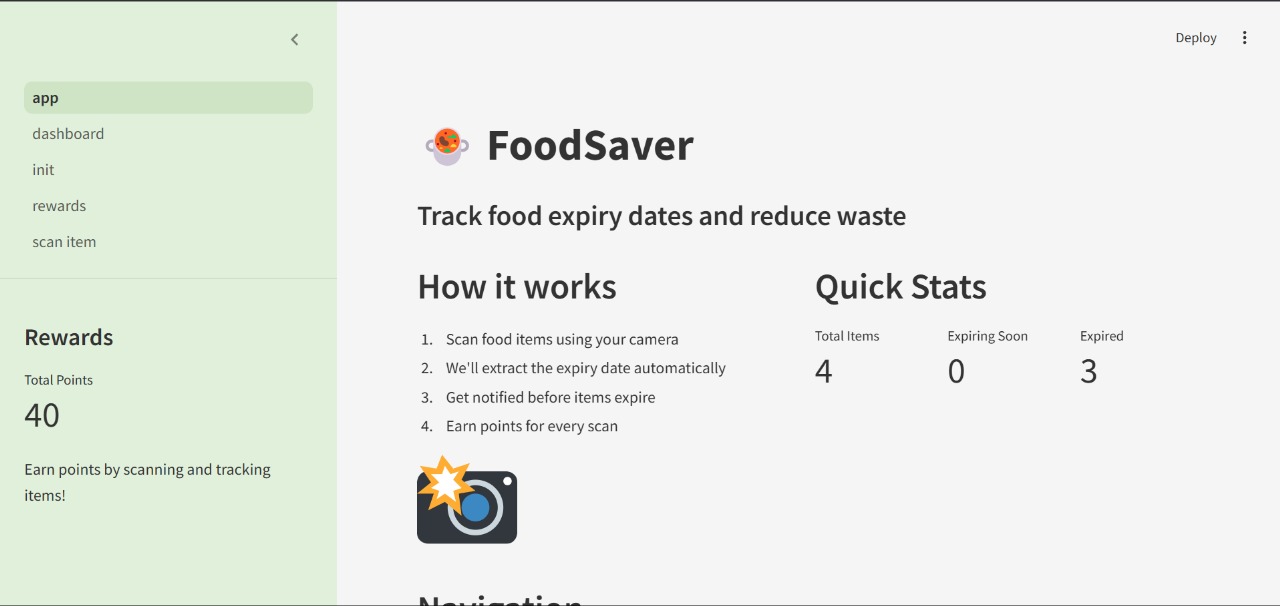
## 6.1. UI Description

<< Provide a brief description of the UI that will be used in this program and how users will interact with the program.

Example: We are creating a research project using R and Azure ML. R is a statistical programming language that uses the R console, which will be the primary means of interacting with our code. This uses standard console UI, and it is not in the scope of our project to create a UI on top of that. The Azure ML machine learning model has its own built-in drag and drop UI for the different code modules we create, so there is no need to generate any sort of AI for that portion of the project either. >>

## 6.2. UI Mockup

<< Attach the screenshot of user interface. This can be a simple drawing that demonstrates key parts of the user interface.>> Example:

**Figure 6: Sample 6** 

**7. ALGORITHMS/PSEUDO CODE OF CORE FUNCTIONALITY**

BEGIN

IMPORT necessary libraries:

- streamlit

- os, pandas, datetime

- custom modules: data\_manager, notifications

INITIALIZE session state variables:

IF 'items\_df' not in session:

LOAD food item data and assign to session 'items\_df'

IF 'points' not in session:

LOAD reward points and assign to session 'points'

SET Streamlit page configuration:

- Title: "FoodSaver - Expiry Date Tracker"

- Icon: 🍲

# 8. PROJECT CLOSURE

This section elucidates the overall lookup at the project and some of the future works that may enhance the solution.

## 8.1. Goals / Vision

Our original vision for the FoodSaver project was to develop a mobile application that tracks food expiry dates, sends timely notifications, and provides recipes to help users utilize their ingredients before they go bad. The initial goal also included integrating a reward system to encourage responsible food consumption and reduce household food waste.

As the project evolved, the focus expanded to include enhancing user experience through a visually intuitive interface, simplifying the food entry process with barcode and image-based input, and improving backend efficiency for faster alert generation. The primary goal shifted toward building a robust and scalable solution with modular architecture, allowing easy integration of future features like donation suggestions or grocery list automation. While the core mission of reducing food waste remained intact, we prioritized user engagement and ease-of-use to ensure sustained adoption and impact.

## 8.2. Delivered Solution

Initially, the plan for FoodSaver was to create a basic mobile application that allows users to manually input food items and receive expiry notifications to help reduce food waste. Additional features like simple points-based reward system were also outlined as stretch goals depending on available time and resources.

The final solution delivered goes beyond the initial plan. It consists of a fully functional Android mobile application that includes:

* A clean and intuitive user interface for adding and managing food items.
* Multiple input methods, including manual entry, barcode scanning, and image recognition (prototype stage).
* Automatic expiry alerts with customizable notification timings.
* Personalized recipe suggestions based on soon-to-expire items.
* A reward system that tracks user behavior and awards points for reducing food waste.
* Backend infrastructure to support data storage and notification logic.
* Basic analytics dashboard (admin-facing) to monitor user engagement and performance.

In addition, the project is supported by complete documentation, a user guide, and modular code architecture to allow for future scalability and feature expansion. This comprehensive delivery reflects both the original goals and the enhancements identified during the development process.

## 8.3. Remaining Work

While the FoodSaver application is fully functional and meets the core goals of food tracking, recipe suggestions, and user engagement through reward points, there are several ways the project can be further enhanced.

One important next step is to refine the image recognition and barcode scanning features, which are currently in prototype stage. Improving their accuracy and reliability will significantly enhance user convenience. Another recommended improvement is the development of a web-based dashboard for syncing user data across devices and offering additional data insights (e.g., monthly food waste reduction stats).

To further promote adoption, the app could include features like grocery list automation, donation suggestions for unused food, and community challenges that encourage collective food saving efforts. Collaborating with local NGOs, grocery stores, or environmental groups could also provide real-world incentives and visibility.

Additionally, implementing in-app feedback collection and conducting larger-scale user testing would help identify usability issues and inform future updates.

# REFERENCES

<<Would suggest you to download a tool (Mendeley: https://www.mendeley.com) for automated citation and generation of references>>

1. T. Dillon, C. Wu and E. Chang, “Cloud Computing: Issues and Challenges”, 24th IEEE International Conference on Advanced Information Networking and Applications, pp. 27-33, 2010.
2. H. Casanova, et al. Heuristic for scheduling parameters sweep applicationsin grid environments, in: Proceedings of the 9th Heterogeneous Computing Workshop, HCW, pp. 349–363. 2009.
3. Kwok, Yu-Kwong, and Ishfaq Ahmad. "Static scheduling algorithms for allocating directed task graphs to multiprocessors." ACM Computing Surveys (CSUR) 31, Vol. 4, pp. 406-471, 1999.
4. A. Mutz, R. Wolski, Efficient auction-based grid reservation using dynamic programming, in: IEEE/ACM Int’l Conf. on Super Computing, SC 2007.
5. M. Mezmaz, N. Melab, Y. Kessaci, Y. C. Lee, E. G. Talbi, A. Y. Zomaya and D. Tuyttens, “Parallel Bi-Objective Hybrid Metaheuristic for Energy-Aware Scheduling for Cloud Computing Systems”, Journal of Parallel Distributed Computing, Elsevier, Vol. 71, pp. 1497-1508, 2011.
6. R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and I. Brandic, “Cloud Computing and Emerging IT Platforms: Vision, Hype and Reality for Delivering Computing as the 5th Utility”, Future Generation Computer Systems, Elsevier, Vol. 25, pp. 599-616, 2009.
7. Basements and crawl spaces. Retrieved from http://www.hud.gov/ offices/hsg/sfh/ref/sfhp1-

25.cfm (Access in June 2020).