Assignment One

Analysis and Design Document

Student: Maghiar Catalin Andrei

**Group: 30432**

Table of Contents

1. Requirements Analysis 3

1.1 Assignment Specification 3

1.2 Functional Requirements 3

1.3 Non-functional Requirements 3

2. Use-Case Model 3

3. System Architectural Design 3

4. UML Sequence Diagrams 3

5. Class Design 3

6. Data Model 3

7. System Testing 3

8. Bibliography 3

1. Requirements Analysis

# Assignment Specification

This is an application that helps the users minimize their food waste more efficiently by sending them notifications when the food is too much or near to the expiration date.

# Functional Requirements

The requirements for the application are to use a layered architecture, to use the Abstract Factory pattern to create weekly/monthly reports, to use a database to store the data.

# Non-functional Requirements

The validation of the data was necessary in case there are users that do not provide the correct credentials. This is necessary in order for the application to not store “bad” data into the database.

2. Use-Case Model

*Use case goal: User login and shopping cart input which will calculate the food waste levels and finally sending notifications with the danger*

*Level: <one of: summary level, user-goal level, sub-function>*

*Primary actor: User*

*Main success scenario: User logs in then provides the data for their shopping cart then the application sends notifications if food waste levels are high*

*Extensions: User provides bad login credentials (failure), User is able to see weekly/monthly reports (success)*

![A close up of text on a white background

Description automatically generated]()

3. System Architectural Design

**3.1 Architectural Pattern Description**

The architectural pattern used is the layered architecture pattern, also known as the n-tier architecture pattern. Components within the layered architecture pattern are organized into horizontal layers, each layer performing a specific role within the application. Although the layered architecture pattern does not specify the number and types of layers that must exist in the pattern, most layered architectures consist of four standard layers: presentation, business, persistence, and database.

**3.2 Diagrams**

*[Create the system’s conceptual architecture; use architectural patterns and describe how they are applied. Create package, component and deployment diagrams]*

4. UML Sequence Diagrams

*[Create a sequence diagram for a relevant scenario.]*

5. Class Design

**5.1 Design Patterns Description**

*[Describe briefly the used design patterns.]*

**5.2 UML Class Diagram**

*[Create the UML Class Diagram and highlight and motivate how the design patterns are used.]*

6. Data Model

*[Present the data models used in the system’s implementation.]*

7. System Testing

*[Present the used testing strategies (unit testing, integration testing, validation testing) and testing methods (data-flow, partitioning, boundary analysis, etc.).]*

8. Bibliography