Team Three: Project Design

Constance Sturm, Michele Cook, Chelsea Nieves, Valerie Rudich, and Hoi Lam Wong

University of Maryland Global Campus

CMSC 495-7382: Capstone in Computer Science

Professor David Castillo

November 14, 2023

**Team Three: Project Design**

**Table of Contents**

1. Introduction
   1. Purpose of Project Design
2. Requirements Analysis
   1. Functional
   2. Non-Functional
3. Use Case Models
   1. Use Case 1.0: View Main Menu
   2. Use Case 2.0: View Rules
   3. Use Case 3.0: View Fortune
   4. Use Case 3.1: Choose Another Fortune
   5. Use Case 3.2: Go Back to the Main Menu
   6. Use Case 4.0: Exit Program
4. Use Case Sequence Diagram
   1. Use Case Diagram
   2. Sequence Diagram
5. Activity Diagram
6. Text Files Description
   1. Example
   2. Purpose
   3. Constraints
7. Entity- Relationship Diagram
8. Conclusion

## **1.0 Introduction**

The purpose of the design plan is to demonstrate how the application will be implemented and developed. This plan design will show all aspects of the application including the format and tools that will be used to implement different functions. Each part of the application will be fully described including the different prompts, the user interaction, how the application processes commands, and what the desired result should be. The project design should fully show what each part of the application will include, as well as, an overview of how the code will be structured. The project will be written in python and will use tools such as a tkinter for GUI , random generator, and buffered reader to read .txt files.

## **2.0 Requirements Analysis**

### **2.1 Functional**

The application requires that the user makes selections from a GUI menu. The menu will be fully described with notes about how the application should function. There will be a menubar applied to the rules and regulations aspect of the program which will be fully described with coding references. The random generator that is used for the “Surprise Me” option and other categories to display a random fortune which will be described with code references and tested for randomness. The different fortune .txt documents will be described with examples of what the fortunes will look like for each category and how the buffered reader will work in the program. Any messages for the user will be demonstrated in the project design.

### **2.2 Non-functional**

Non-functional aspects include the welcome message which will be displayed using labels in the GUI. The testing will also be a non-functional but important aspect of this application.

## **3.0 Use Cases**

**Actor Profile:** User who desires to gain a deeper insight into their future using the program. A user can be new to the system or have prior experience with the program. This program is suitable for all ages with the use of a computer.

### **Use Case 1.0: View Main Menu**

**Actor:** User

**Description:** Displays the welcome message and asks the user if they would like to login. After the user chooses to login or play as a guest they will then be shown the main menu.

**Preconditions:** The user is welcomed to the program and prompted to choose if they’d like to login or play as a guest.

**Sequence of events (scenario):**

1. The user is welcomed to the program
2. The user is prompted to choose from the following menu.
   1. If they’d like to login
      1. yes/no buttons
3. The program then takes the user to the menu or prompted to login/register.

**Postconditions:** The user is then transferred to another menu or prompt based on their selection.

### **Use Case 2.0: View Rules**

**Actor:** User

**Description:** The user wants to view the rules for using the fortune teller program.

**Preconditions:** The user selected the choice to view the rules from the menu bar at the top of the window

**Sequence of events (scenario) :**

1. The user selects the ‘Rules’ option from the menu bar at the top of the window.
2. The program displays the rules for using the fortune teller.
3. The program uses a button to close the rules window.

**Postconditions:** The user is navigated back to the main menu.

### **Use Case 3.0: View Fortune**

**Actor:** User

**Description:** The user wishes to receive a fortune based on a specific category of their choice.

**Preconditions:** The program has started and the user is presented with the fortune menu

**Sequence of events (scenario) :**

1. The user is transferred to the fortune menu after selecting the login/guest option.
2. The program displays a menu with options to choose a fortune category, including Love, Career, General, Health, and Surprise Me (random).
3. The user is prompted to click a button that corresponds to the fortune that they’d like to view.
4. The program displays a fortune from the selected category.
5. The user is prompts to choose between:  
   - “Choose Another Fortune”

- “Back to Main Menu”

- “Save their fortune”

**Postconditions:** The user has received a fortune from their selected category and can either choose another fortune, save their fortune, or return to the main menu.

### **Use Case 3.1: Choose Another Fortune**

**Actor:** User

**Description:** The user, after viewing a fortune, decides to choose another fortune from a different category of their choice.

**Preconditions:** The user has already selected and viewed a fortune from a specific category using the Fortune Teller program.

**Sequence of events (scenario) :**

1. After viewing a fortune, the user is prompted with the option to“Choose Another Fortune”, “Save their Fortune”, or “Go Back to Main Menu”.
2. The user selects “Choose Another Fortune”
3. The program displays a menu with options to choose a fortune category, including Love, Career, General, Health, and Surprise Me (random).

**Postconditions:** The user is presented with the menu of fortune categories again and can select a different category to view another fortune.

### **Use Case 3.2: Go Back to the Main Menu**

**Actor:** User

**Description:** The user, after viewing a fortune, decides to return to the main menu without making any further selections.

**Preconditions:** The user has viewed a fortune from a specific category using the Fortune Teller program and is prompted to view another fortune, save their fortune, or return to the main menu.

**Sequence of events (scenario) :**

1. After viewing a fortune, the user is prompted with the option to“Choose Another Fortune” or “Go Back to Main Menu”.
2. The user selects “Go Back to Main Menu”
3. The program displays the main menu of the Fortune Teller program.

**Postconditions:** The user is taken back to the main menu where they can make new selections.

### **Use Case 4: Exit Program**

**Actor:** User

**Description:** The user intends to exit the fortune teller program

**Preconditions:** The program has been launched, and the user is presented with the main menu.

**Sequence of events (scenario) :**

1. The user hover over the ‘Exit’ option from the menu bar.
2. The ‘Exit’ option display a dropdown menu with option “Exit Program”
3. The user clicks “Exit Program”
4. The program ceases to run.

**Alternate Flow**

4a. The user chooses to stay in the program and is presented with

the main menu where the user can make another selection.

**Postconditions:** The program exits or returns to the main menu based on the user's choice.

## **4.0 Diagrams**

### **4.1 Use Case Diagram**

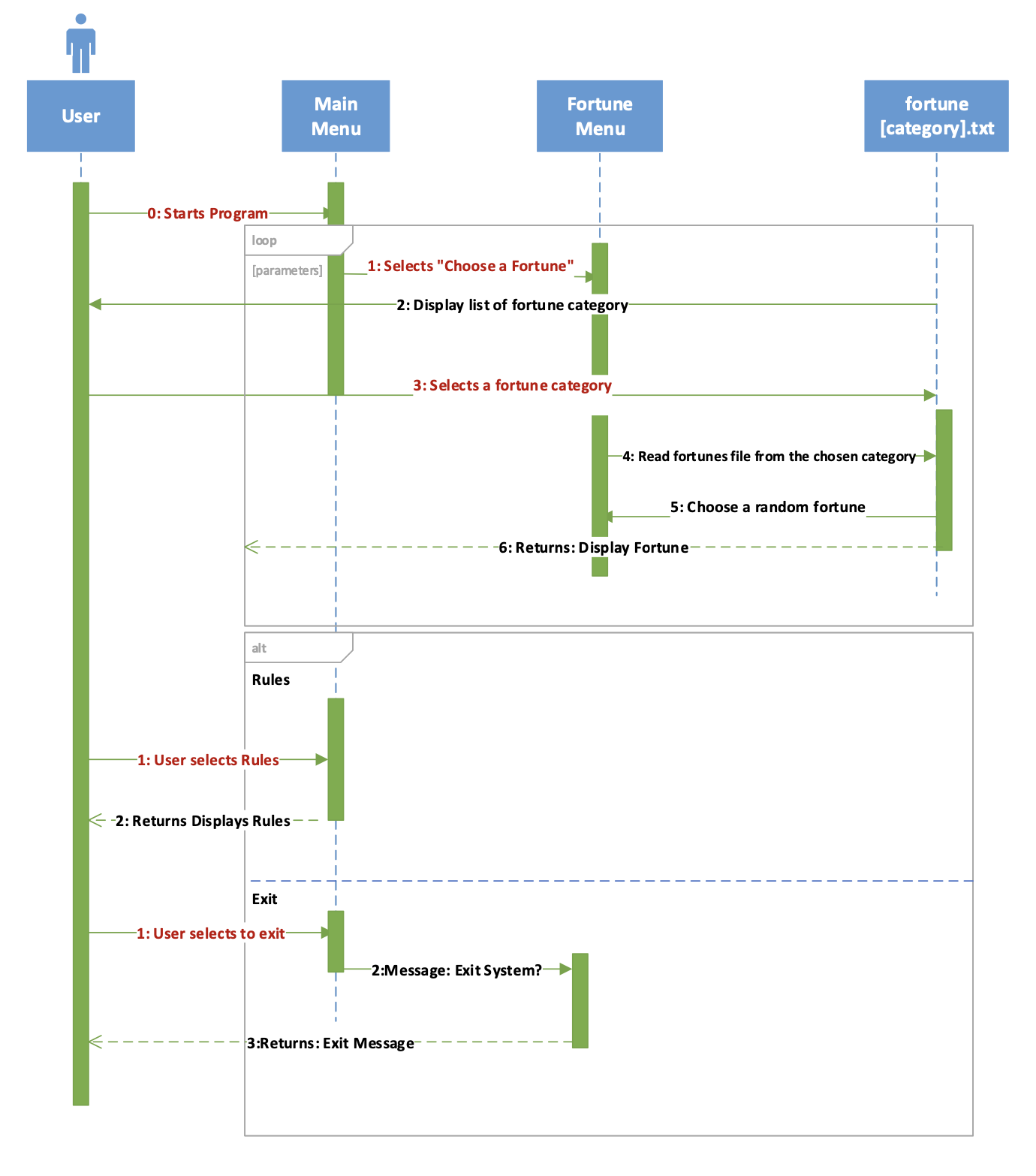
### 

**Description:**

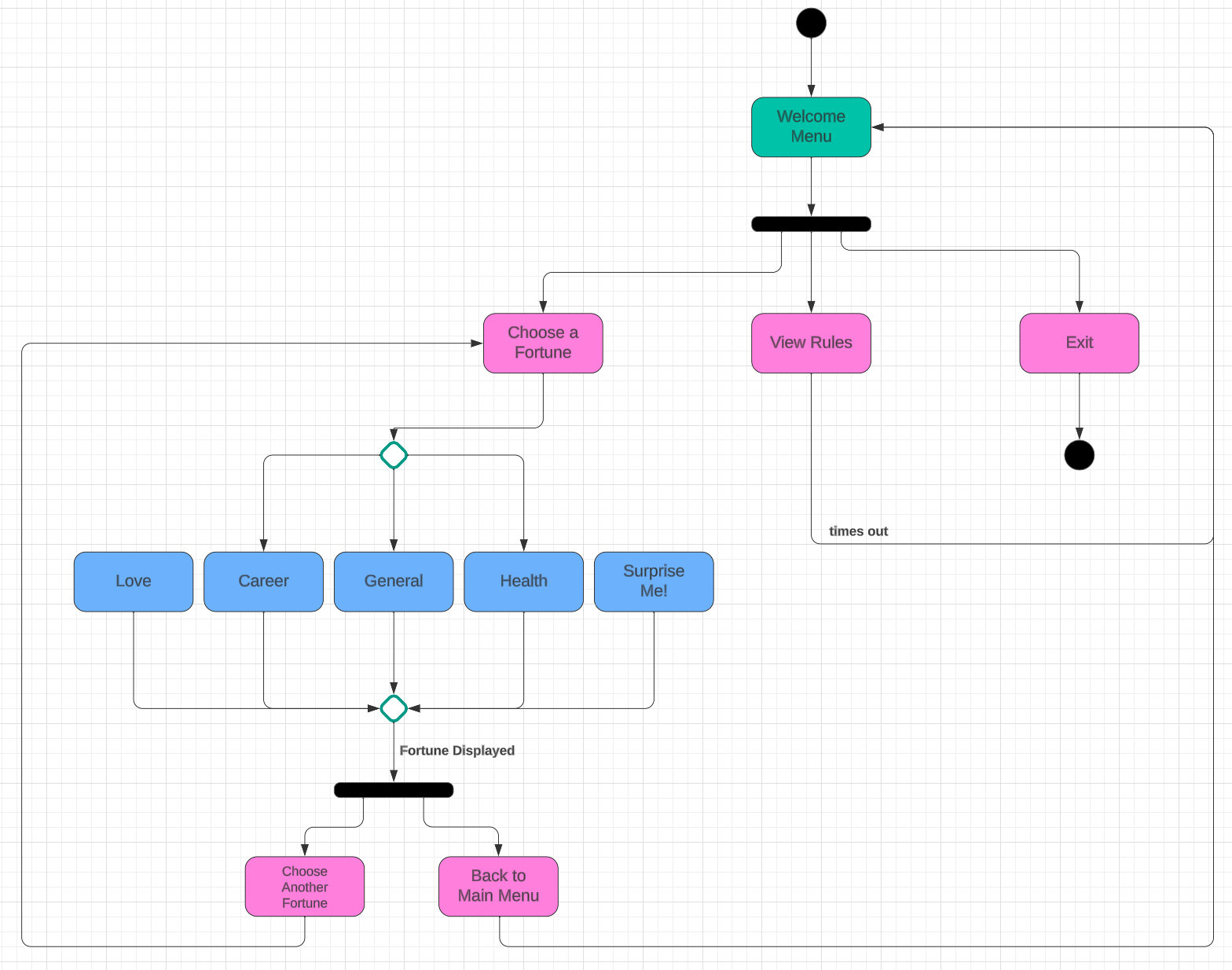
The figure represents the actor/user that interacts with the main system. When the user first starts the Fortune Teller program, he/she is performing the “View Main Menu'' action. These menus are represented by the ellipses. After the user is welcomed, they are prompted to login, register, or play as a guest. After that they can proceed with the “View Fortune ” menu options. The rules and exit options are displayed in the upper menu bar for ease of use. These are represented with arrows to show the relationship or action the user can choose. The cases are extended from the main or base use case. The use cases that extend from “View Main Menu” and are optional. If the user decides to view a fortune, they will be prompted to either “Go Back to Main Menu”, “Save their Fortune”, or “Choose Another Fortune” which are mandatory and they must choose one to perform.

### 

### **4.2 Sequence Diagram**



## **5.0 Activity Diagram**

****

## 

## **6.0 Text file**

The program will be reading from .txt files for the different fortune categories to display a fortune to the user based on their selection.

### **6.1 Example**

fortune\_love.txt

:This is a love fortune:This is a 2nd love fortune:This is a 3rd love fortune:...

fortune\_general.txt

:This is a general fortune:This is a 2nd general fortune:..

fortune\_health.txt

:This is a health fortune:This is a 2nd general fortune:..

fortune\_career.txt

:This is a career fortune:This is a 2nd career fortune:..

The random option will read a fortune from all the .txt files and chooses one randomly.

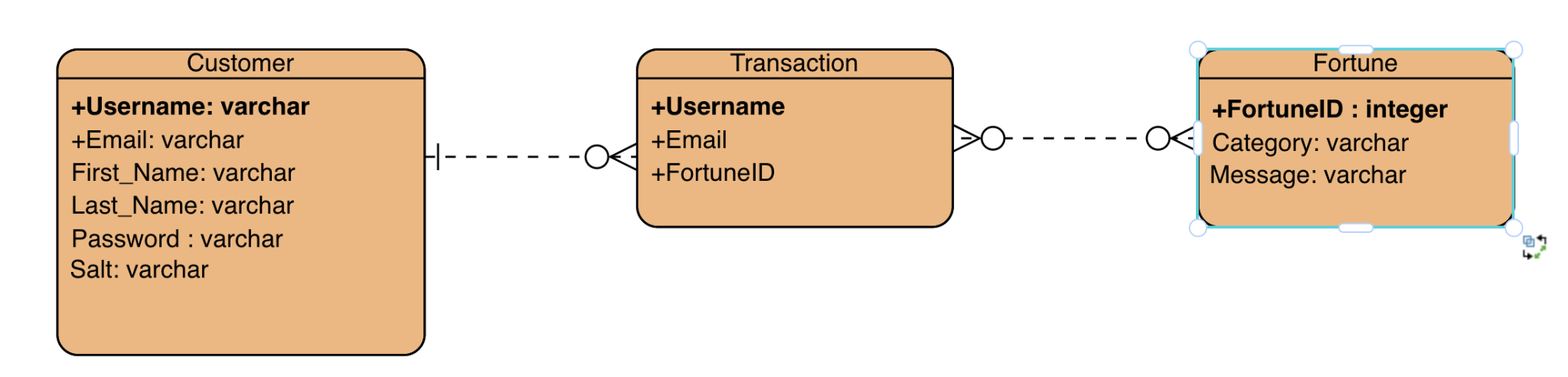
### **6.2 Purpose**

The multiple text files *fortune\_[category].txt* will represent a collection of fortunes within a specific category. For example, *fortune\_love.txt*  will hold the fortune messages from the category, Love. Each category has their own message getter function which accesses different text files according to the user’s input. A separate function would then pick a random fortune message within the category. For random category selection, multiple text files will be read and a fortune is then picked.

### **6.3 Constraints**

The file name must follow the naming convention “ fortune\_[category].txt” where [category] is the name of the fortune category the user has chosen. For example, love fortunes are stored in the text file “fortune\_love.txt”. Each fortune message is separated with a delimiter, serving as a distinct marker that separates one fortune from another within the text file. This allows efficient parsing and retrieval of individual fortunes. With that said, the character that serves as the delimiter cannot appear within the fortune message itself. As such, a “:” would be used to separate messages to enhance the reliability of parsing.

## **7.0 Entity- Relationship Diagram**



## **8.0 Conclusion**

The project will have user interaction using a GUI menu to play as a guest or login/register as a returning user. The rules selection will be displayed with the menu bar selection. The fortune menu will be displayed with the incorporation of buttons to choose the fortune that they would like to display. The returning users will have the option to save their fortunes to a database to be able to view them again at another time. The categories for the fortunes involve love, career, health, general, and random. The fortunes will have a random generator that will choose from the different .txt files to choose a fortune for the user. With the exit the program option the user will be prompted if they really want to exit the program. If they want to exit they will then have a goodbye message displayed and the program will end. We are using loops for the menu options and the user input to generate a desired result.