Software Requirements

Specification

for

Healthy Food App

Version 1.0 approved

Prepared by

19CE097 Dhruvesh Patel

19CE101 Jeet Patel

CSPIT-CE

25th February 2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this SRS is to outline both the functional and non-functional requirements of the Subject Healthy Food App System. In addition to said requirements, the document also provides a detailed profile of the external interfaces, performance considerations and design constraints imposed on the subsequent implementation. The document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future.

1.2 Document Conventions

HFAS – Healthy Food App System
SRS – Software Requirements Specification

1.3 Intended Audience and Reading Suggestions

The primary audience of this SRS document will be the development team employed to implement the specified Healthy Food App System. It will not only provide an extensive capacity for project planning and progress assessment but it will further assist with stakeholder interactions. The secondary document audience comprises the stakeholders of the project, that is, restaurateurs and associated staff. To this audience group, this SRS should convey and confirm the required functionality and represent a contractual agreement between the involved parties.

1.4 Project Scope

In current formal dining environments, some form of physical static menu is utilized to convey the available food and beverage choices to customers. Said menus are generally paper based and hence impose restrictions on the textual real estate available and the ability a restaurateur has to update them. The related concepts are encompassed by the general scope of the Restaurant food ordering System. It is to the replacement of paper-based menus using an electronic format.

1.5 References

- *You Tube:*

    1. *Figma - https://youtube.com/playlist?list=PLuRPummNMvINdAbI_WT7R5vdjcyRPeRiq*
    2. *Adobe XD - https://youtube.com/playlist?list=PLuRPummNMvIN43IBo1EltezV2nqqYz5T6*
    3. *Wireframe - https://youtu.be/rLdpn0vHV-M*
    4. *Prototype - https://youtu.be/GEnSPZkcKKA , https://youtu.be/4aBcCmYGxxk*

- *Figma Documentation: https://www.figma.com/best-practices/guide-to-developer-handoff/components-styles-and-documentation/*

- *Adobe XD: https://helpx.adobe.com/in/xd/user-guide.html*

2. Overall Description

2.1 Product Perspective

The software described in this SRS is the software for a complete Healthy Food App System. The system merges various hardware and software elements and further interfaces with external systems. it relies on a number of external interfaces for persistence and unhandled tasks, as well as physically interfacing with humans.

2.2 Product Features

The Healthy food app system interfaces with an existing system, including Body data with your goal details give and than give suggestion which food healthy or not healthy, in order to quickly and easily give details about your progress. And also give food details in depth like food calories, ingredients and some other instructions. The set schedule system should be operable such that it can return information to the HFAS as to whether schedule is today or not.

2.3 User Classes and Characteristics

There are three separate user interfaces used by the HFAS software, each related to an interfaced physical hardware device. These three user interfaces are the Surface Computer UI, Tablet UI and Display UI.

2.4 Operating Environment

The Surface Computer UI, Tablet UI, Display UI is the interface used by user. This interface uses the surface paradigm - users interact with the system by dragging 'objects' around on the flat screen touch-sensitive display.

HFAS is also an Android Application which is curated to run on Android Device.

Minimum Software/Hardware Requirements:

- *Android Version 5.0+*
- *100MB Free Storage Space*
- *1GB Ram*

2.5 Design and Implementation Constraints

The HFAS UI should be design in figma. And also design site map in figma. The design should be more creative, neat, clean and professional.

2.6 User Documentation

The end-users of the HFAS fall into three primary categories, unskilled, partly skilled and highly skilled.

2.7 Assumptions and Dependencies

The SRS assumes that none of the constituent system components will be implemented as embedded applications. It is further assumed that tablet PCs of sufficient processing capability and battery life will be utilized. The user should have the basic knowledge of computing and web-surfing, with mainly the knowledge of mobile usage in order to access the application efficiently. The user should, obviously, have the basic knowledge about the core factor of the application's aim and how to interact with it. There are no explicit requirements for the usage of the application. Another basic requirement would be having the knowledge of the reading a map. As the application aims to target no specific audience but all the common folk, it does not required any special or specific skill sets to operate it. The interface has been made as simple and minimalistic as possible.

3. System Features

The application HFAS comes in with a variety of features which prove its worth in the society.

3.1 System Feature 1

Account Feature:

3.1.1 Description and Priority

The application provides the feature of making accounts so as to save personal data and enable better interaction and efficiency with the application. The user can make personal accounts providing their contact details such as mobile number, email id and address for a better interaction with other interested 'people. There would be options provided for public display of some of the details whether to hide or show them. Therefore, if a buyer is interested in buying some item, they can check out the seller's details such as their name and email address and contact them on their personal mobile number if needed. This feature is highly important and prioritized because without it there would be no existence and transparency of the application. This feature also helps in saving their personal interaction with the application, saving their search data, the products they last saw and other things required for a smoother experience. Most importantly, providing the location fillers out the searches and makes neighborhood interaction easier.

3.1.2 Stimulus/Response Sequences

Upon entering the application, the user will be prompted to login to an existing account or sign in to
create a new account, If the user already has an account they can login and continue their activities on the application. If the user is new, they can tap on the sign in button and provide the required information so as to successfully create their account, Upon completion of the account creation process, The user will have to confirm their account by accessing the mail sent on their email account and verify their identity from there. After this, they can easily work around their way in the application.

3.1.3 Functional Requirements

The user should have proper contact number and a working and active email address for the successful completion of the account creation process.

If any of the information provided is ambiguous or wrong, the user will be prompted time and again to provide a valid email address or mobile number.

Also, unless the data provided is not been verified, the user cannot use their account and access the application's full potential and functionality.

3.2 System Feature 2

Food details:

3.2.1 Description and Priority

Upon uploading the food, the user will have to see food and check food details and qualities of food and judge food and check food id more efficiently and dawn upon the decision as whether they should to take the food or not or is the food their needs. All information can be easily extracted from the interface.

3.2.2 Stimulus/Response Sequences

During to take food, the user will check the food and add in schedule and save it. And also see categories of food and categories of schedule like breakfast, lunch, dinner, snack, juices. And also user can all information see in the app.

3.2.3 Functional Requirements

The feature, obviously, requires complete registration of the user and also user need all information about food to add in schedule.

3.3 System Feature 3

Add to Favorites:

3.3.1 Description and Priority

This feature provides the luxury of storing all the food that a particular user liked at one place, so that they do not need to search it again anytime and find them under their favorites section. The favorites section can contain any number of foods that the user might have liked while surfing on the application. If the user dislikes the product and wants it to get removed from the favorites section, they can simply un like the product and it will automatically be removed from their favorites list.

3.3.2 Stimulus/Response Sequences

Every food will contain a heart shaped transparent icon near them wherever they appear. Upon tapping on the heart icon, the food will be marked as liked and will be added to their favorites section. The food upon searching, will now show an opaque color filled heart icon wherever they will be seen.

The product's opaque heart can be tapped again to make it transparent. Upon doing that, the food will be removed from the favorites section of the user's account.

3.3.3 Functional Requirements

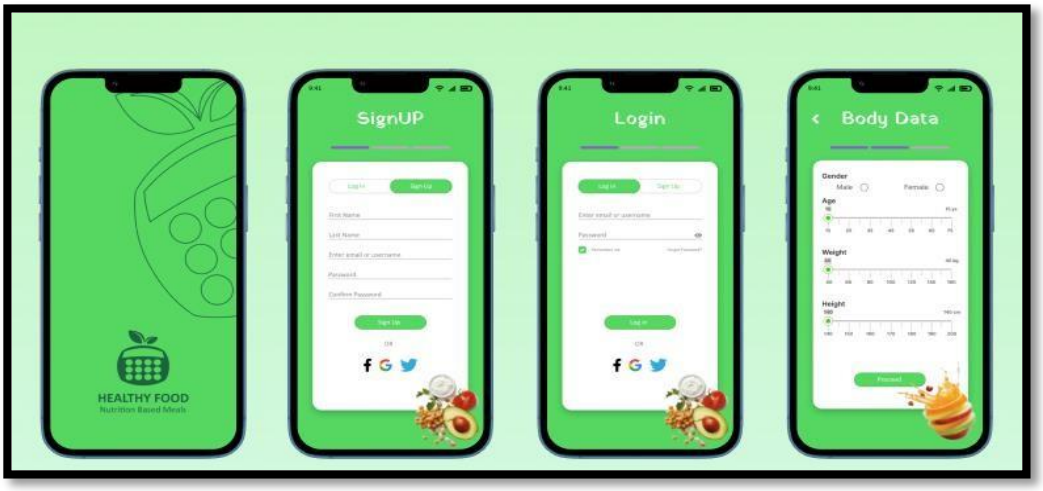
The feature of adding food to the favorites section requires the user to be logged in to their account. If the user is not logged in, they would not be able to use this specific feature as there would not be a database of the user's name to store the favorite food data.

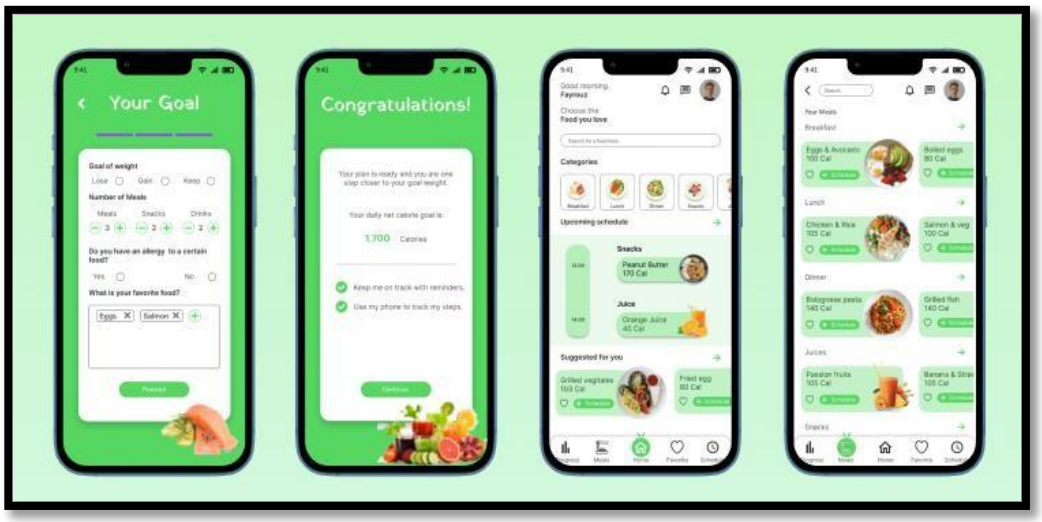
4. External Interface Requirements

4.1 User Interfaces

This interface uses the surface mobile paradigm - users interact with the system by dragging 'objects' around on the flat screen touch-sensitive display. For the HFAS, users can manipulate objects such as items of food, Juices, tips and menus on the surface of their categories. Such objects can be moved into static objects such as meals to perform various functions. In addition to this object manipulation paradigm, a limited system menu is necessary. Users will summon their food, which is combined with a system/command food, using an easy touch gesture, a double-tap on the touch surface, and dismiss it with a similar gesture or by tapping a close button GUI element.

UI Designs:






Chicken and Rice Bowl
0
Keep going, you are doing well!
510
Keep going, you are doing well!
Schedule Today
Set Schedule

4.2 Hardware Interfaces

These devices are the surface mobile, the wireless tablets and the touch displays. All the devices must be physically robust and immune to liquid damage and stains. The devices (with the possible exception of displays) must also have good industrial design aesthetics.

4.3 Software Interfaces

The HFAS will interface with a Database Management System (DBMS) that stores the information necessary for the HFAS to operate. The DBMS must be able to provide, on request and with low latency, data concerning the food menu, user (and their passwords) and available dietary requirements.

4.4 Communications Interfaces

The HFAS will interface with a Local Area Network (LAN) to maintain communication with all its devices. It should use a reliable-type IP protocol such as TCP/IP or reliable-UDP/IP for maximum compatibility and stability. All devices it will interface with should contain standard Ethernet compatible, software accessible LAN cards to maintain communication between the server and the surface mobile, computers, tablets, displays and the external system.

5. Other Nonfunctional Requirements

This subsection presents the identified non-functional requirements for the subject HFAS. The subcategories of non-functional requirements given are performance, safety, security, requirements.

5.1 Performance Requirements

The server shall be capable of supporting an arbitrary number of surface mobile, computers, tablets and displays, that is, it shall provide no limit on how many devices are in the system. The server shall be capable of supporting an arbitrary number of active user, that is, no data shall be lost under any circumstances.

5.2 Safety Requirements

The system shall log every state and state change of every surface mobile, computer, tablet and display to provision recovery from system failure.

The system shall be capable of restoring itself to its previous state in the event of failure (e.g. a system crash or power loss).

5.3 Security Requirements

A user password used for mobile login must have a bit-strength of at least 64 bits. A user password used for mobile login must be changed every three months. A user shall only be able to log into one mobile at any given instance of time. A user that attempts to log into a second mobile while already logged into.

5.4 Software Quality Attributes

The software shall be capable of supporting an arbitrary number of surface mobile, computers, tablets and displays, that is, it shall provide no limit on how many devices are in the system.

6. Other Requirements

No other extra requirements defined as of now. Subject to change.

Appendix A: Glossary

*To Be Decided**

Appendix B: Analysis Models

*To Be Decided**

Appendix C: Issues List

*To Be Decided**