CA326

Project Name:
Phaze

Prepared by:
Kevin Bortas
Kevin Cogan

Software Requirements Specification

Version: (1.0) **Date: (21/12/2020)**

Table of Contents

6. Supporting Information

1. Introduction

1.1 Purpose

This document outlines the Phaze Application's system requirements and its analysis. This will provide the purpose of offering a reference point for the system design. This will be used by system designers, project coordinator, project supervisor and the CA326 module demonstration panel.

1.2 Scope

Phaze is a dieting and fitness application that makes fitness easy. We have identified that leading dieting and fitness apps have been losing users due to lack of ease of use with their calorie-counting feature where the user has to physically type in the food into their mobile device. We plan to leverage the affordances of AI to create a mobile application that automatically identifies the food by simply taking a photo of your food. This will calculate the calorie intake of the user for the day and display this information to the user so they can make more informed decisions on their food intake. We also monitor the user steps per day so we can subtract any burned calories from their total food intake ensuring the user can achieve their desired goal of weight loss, maintain their weight, weight gain, which is determined on setting up their account and automatic bi-weekly reminders that check-in on the users to see how they are doing with their fitness plan and if it needs to be adjusted. Phaze helps the users achieve their fitness goals in a fun and interactive way through the use of gamification. We will reward users for achieving their fitness goals by offering them virtual rewards that give the users an incentive to stay motivated during these tough times.

1.3 Definitions, Acronyms, and Abbreviations

AI: Artificial Intelligence, this is the use of technology to try to simulate human intelligence in machines. This enables the machine to take in input and process it, giving a response that a human would give or a more accurate response than what a human would give.

1.4 References

[1] My Fitness Pal. [Online] Available: https://www.myfitnesspal.com/welcome/learn_more [Accessed on 20/12/2020]

[2] Kaggle. [Online] Available: https://www.kaggle.com/kmader/food41?select=images [Accessed on 19/12/2020]

[3] Smartsheet. [Online] Available: https://www.smartsheet.com/understanding-agile-software-development-lifecycle-and-process-workflow#:~:text=The%20overall%20goal%20of%20each,software%20as%20quickly%20as%20possible.&text=As%20an%20example%2C%20the%20full,%2C%20production%2C%20and%20retirement%20phases [Accessed on 20/12/2020]

[4] AWS. [Online] Available:https://aws.amazon.com/products/databases/[Accessed on 21/12/2020]

1.5 Overview

The following document details:

Section 2: This section contains an overall description of Phaze. It contains all of the different interfaces and what their functions are. This details more towards the user/customer side of the product and how they will be interacting with it.

Section 3: This section contains an overview of Phaze and caters towards the developer side of the product and how we plan to design it.

Section 4: This section describes how we hope to manage the change process. Here it states what development methods will be used and how we plan to implement them.

Section 5: This section is used for the approval of this document, which is to be signed by the project supervisor and the project coordinator.

Section 6: Section 6 contains any additional information which might be useful when reading this SRS.

2. The Overall Description

2.1 Product Perspective

The project is an independent and totally self-contained system with three main purposes.

1. To be able to recognise different food items in a picture taken by the user accurately.
2. To be able to retrieve the nutritional information from the picture taken by the user.
3. To store the total number of calories and macronutrients that a user has eaten throughout the day.

2.1.1 System Interfaces

Our project will use a number of system interfaces.

System	Function	Position
Firebase Database	This database will store all of the user log-in information such as email and passwords.	Back-end
Android Application	This will be the UI for Phaze. This is where the user can take pictures of their meals and view their total number of calories and macronutrients eaten.	Front-end
Apache Web-Server	This will act as a messenger for the front-end of the app to the back-end.	Back-end
AI Model	This model will analyse the picture taken by users and determine what food item is within the picture.	Back-end
AWS Database	This is where the nutritional information of each food item will be stored.	Back-end

2.1.2 Interfaces

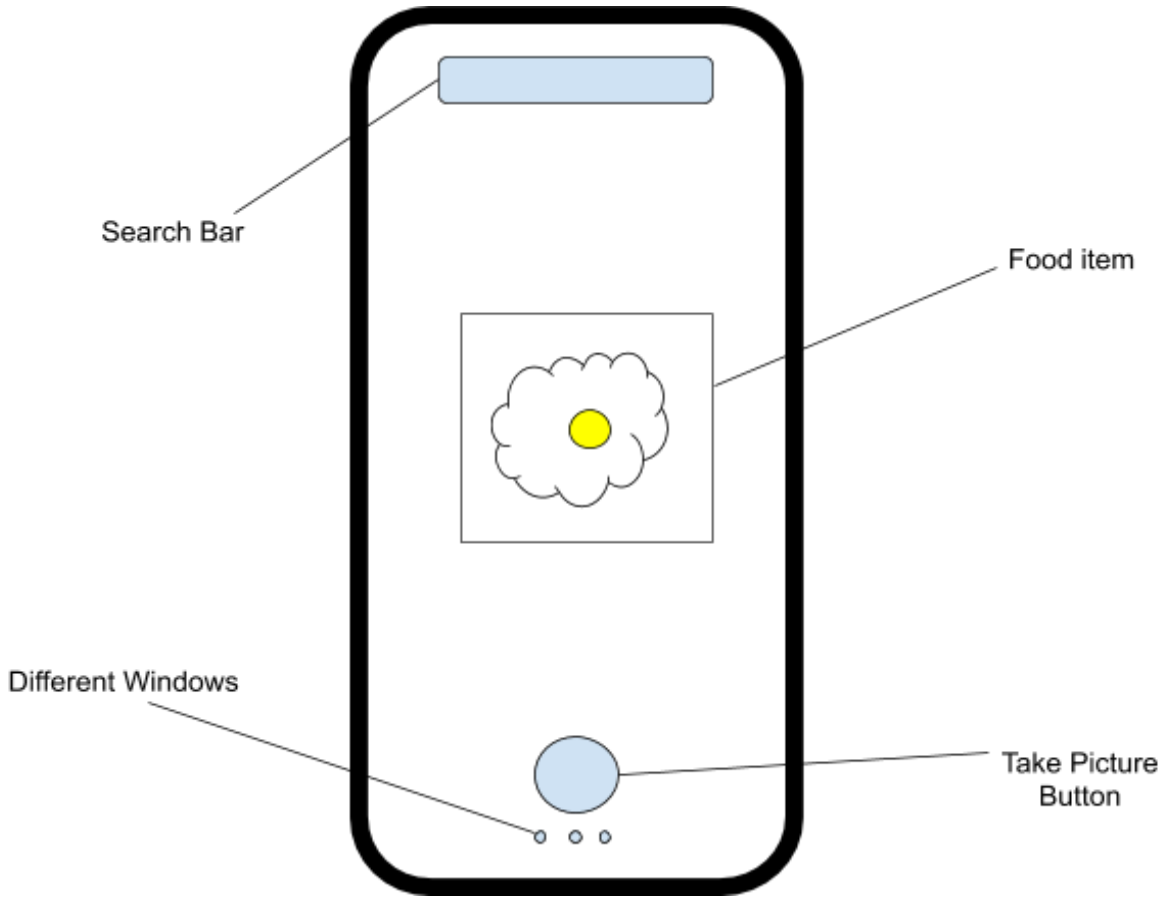

Figure 1. The main interface of our Android application

Phaze will use a graphical user interface on any Android device as shown in figure 1. The user will be prompted to the main screen after logging in. This interface will feature a constantly running video image in the background captured by the device's camera. The interface will also feature a button that the user can use to take a picture of a food item which will then send it to our server where it will be analysed by our deep learning model. The user also will have an option to manually search for a food item using the search bar.

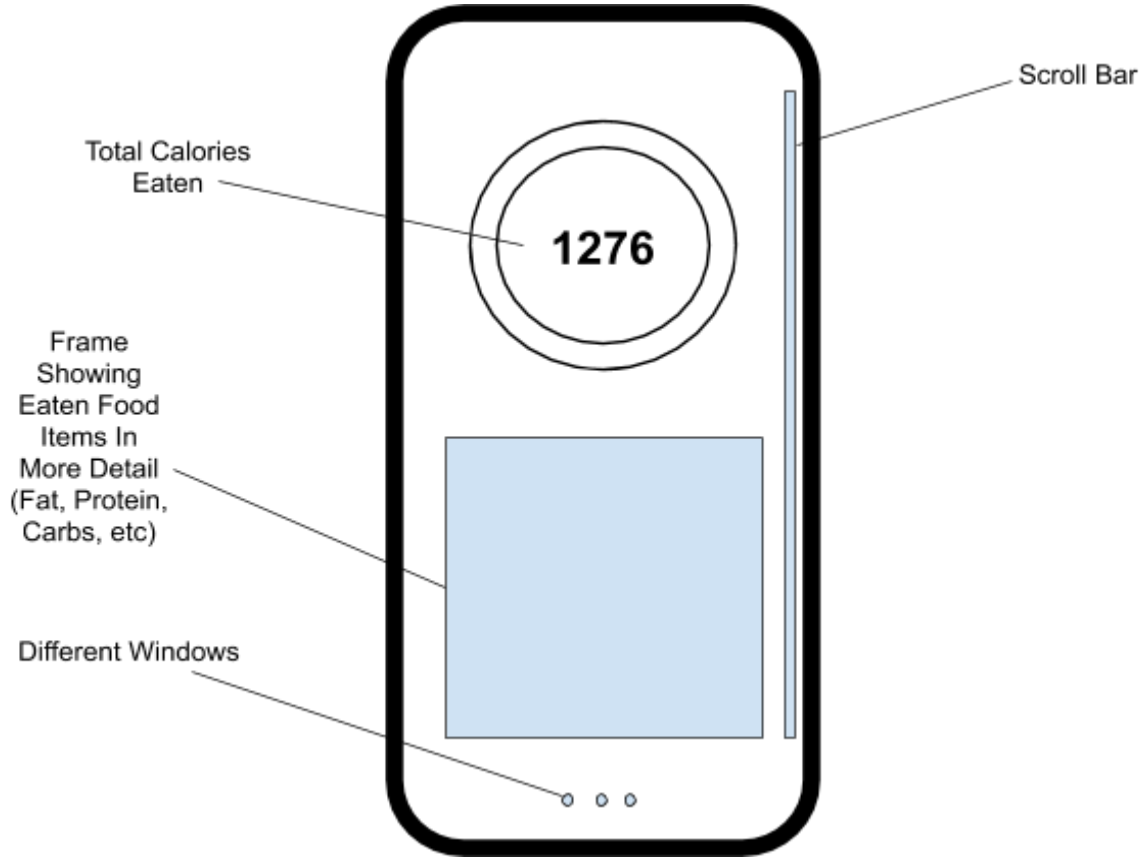

Figure 2. User information window

The user information will be visible on a separate fragment as seen in figure 2. Information such as total calories and the macronutrients of food items eaten will be stored here and they will be updated continuously throughout the day as the user adds more food items. The fragment data will reset every day and the user will start from 0 calories.

2.1.3 Hardware Interfaces

This product will interact with the camera of all Android mobile devices.

2.1.4 Software Interfaces

A SQL database must be used to store and retrieve the nutritional information of the different food items.

2.1.5 Communications Interfaces

The system will use TCP to send the images taken by the users and receive their corresponding nutrition information.

2.1.6 Memory Constraints

12GB of RAM should be enough to train our model to be able to recognise some food types but 128GB would allow it to train using more examples.

2.1.7 Operations

The system supports data saving operations so that the user can upload and update their nutritional status, close the application, and log back in at a later time to check on their status or to update again.

2.1.8 Site Adaptation Requirements

In order to use the product, the latest version of the Android OS must be installed on the user's device.

2.2 Product Functions

The main goal of this product is to accurately and reliably control the diet plans of users. The system will have to communicate with the apache server and our AWS database in order to fetch the nutritional information of each food item and display it on the Android application. The Firebase Database will also contain all of the user information such as email address and password but it will also be used to store their diet progress and keep track of their nutritional status so that it can update the application.

2.3 User Characteristics

It is foreseen that the main users of Phaze will be a wide variety of people, from everyday users looking to achieve a healthier lifestyle to already training athletes looking to keep track of their progress. The simple UI makes this product appealing and attractive to everyday people who might not have the time to learn how to use complex applications or they might be reluctant or intimidated to start a diet plan.

There is also a potential for using the system as an educational tool. With only a requirement of basic nutritional knowledge, users could also get the opportunity to expand their prior knowledge on nutrition and lifestyle.

2.4 Constraints

Hardware Limitations:
The AI model is trained using a 12GB RAM storage. Because of this we can only train our model to recognise a small number of food items as in order to cover all food types, more RAM storage is required.

Higher-order Language Requirements:
The system has to be implemented using the programming languages Python and Java.

Safety and Security Considerations:
The system uses Firebase to store all sensitive user information like email addresses and passwords. For the sake of our CA326 project, this application will only use synthetic information.

2.5 Assumptions and Dependencies

The following had been assumed when writing the product requirements.

1. All users have access to a working camera on their mobile device.
2. The users have the most up to date operating system.
3. The users have access to a working wireless internet connection.

2.6 Apportioning of Requirements.

The ability to recognise all food types will be delayed until future versions of the system are released. This is due to our hardware limitations and not having access to enough RAM storage. This is also impacted by the limited availability of training images, meaning that we must work with what we have we have available to us.

3. Specific Requirements

3.1 External Interfaces

3.1.1

Phase is an independent and totally self-contained system with three main purposes.

1. To be able to recognise different food items in a picture taken by the user accurately using a deep learning algorithm.
2. To be capable of querying a database containing the nutritional information of the foods which the deep learning algorithm had recognised.
3. To store user information such as calories eaten in regards to the type of food queried from the database.

3.1.2

Our mobile application will be centred around the users. All input will be received via an Android mobile application. This input includes:

1. Username and password - at the login page to log into their account.
2. Fitness regime and goals - if they would like to gain, maintain, or lose weight.
3. Photos of their food - to identify the calorie and nutritional information about the food.
4. Bi-weekly check-in - to get feedback on the user if they would like to change their fitness regime as it could be too easy or too difficult for the individual.

3.1.3

Accuracy is essential to the development of the Food Classification Neural Network as research and testing show that an accuracy greater than 85% will ensure the food will be accurately identified by the Neural Network making it worthy for customer use.

3.1.4

The measurement of the accuracy of the food classification neural network is determined by comparison of the inputted imagery with the training dataset and the validation dataset, which gives us the accuracy and the loss (aspects of the image that are different). From this, we generate a percentage accuracy after each epoch then an overall accuracy once all epochs are completed.

3.1.5

The timing to send the image to be classified by the food classification neural network and retrieval of nutritional information need to be less than one seconds to ensure the user conveniently and quickly gets the information they need.

3.1.6

The inputted image of food taken from the user's Android mobile phone is then sent to the Apache web server the image is then sent to the image classification neural network model to be identified. The image label is then returned to the server where it is sent to the AWS Relational database to retrieve the nutritional information for the inputted food label. The food is returned to the server then finally sent to the mobile phone.

3.1.7

The Phaze mobile application is using these data formats:

1. Images will be formatted as JPEG
2. Retrieving information from the food classification model will be in JSON format.
3. Retrieving information from the database will be in CSV format and converted to JSON in the server.

3.2 Functions

3.2.1 Registration

The user shall input their username and password on the registration page on the android application. The user shall be prompted for fitness metrics such as weight, height and fitness goals. This user information will be securely added to the Firebase database with the new user details. The user will be redirected to the login page.

USE CASE 1	Phaze Registration
Goal in Context	The player registers their detail on the Phaze Mobile App
Scope	Functional requirements of a system
Level	Medium Detail
Preconditions	Users must have an Android mobile operating system and have the Phaze application downloaded on their phone.
Success End Condition	User successfully registers their details on the Phaze app

Failed End Condition	User are unable to register due to an error	
Primary, **Secondary Actors**	User. Online server system.	
Trigger	The user selects register now button	
DESCRIPTION	**Step**	**Action**
	1	System presents the user with Registration Page
	2	User inputs the fields of username, email and password
	3	System validates information and checks if username is available
	4	System shows a registration confirmation
	5	User inputs weight, height and fitness plan.
	6	System shows registration complete
EXTENSIONS	**Step**	**Branching Action**
	2a	User updates one of the fields incorrectly

	2b	User leaves a field blank
	3a	System tells player the username is taken
	3b	System tells player the password is invalid
	3c	System tells player the email account already has an associated account
	4b	System shows registration error
VARIATIONS		**Branching Action**
	1	The User selects the option on a version of android mobile phones with different android operating systems.

3.2.2 Login

The user shall enter their username and password into the login page on the android application where the username and password will be checked in the Firebase database containing all the users details. If the details are correct the user shall proceed to their account. Otherwise, the user shall be prompted to input the correct username or password.

USE CASE 2	Phaze- Login
Goal in Context	User logs into the Phaze mobile application.
Scope	Functional requirements of a system
Scope	Functional requirements of a system
Level	Medium Detail

Success End Condition	The user logs into their Phaze account via mobile application.	
Failed End Condition	User struggles to login and leaves the login page.	
Primary, **Secondary Actors**	User. Online server system.	
Trigger	The user taps on the mobile application to open it.	
DESCRIPTION	**Step**	**Action**
	1	The login page is displayed
	2	User types in details on the login screen
	3	The server checks user login details
	4	The home page is displayed to the user
EXTENSIONS	**Step**	**Branching Action**
	3a	Invalid username or password: Display invalid login details
VARIATIONS		**Branching Action**

	1	Users log in on different android operating systems

3.2.3 Image Classification

The user shall use our nutrients from an image feature by taking a picture of their food. This image will then be sent to our Apache web server where it will be classified what food it is by our pre-trained food classification model. The label of the classified food will be queried from the SQL Relational database where the food nutrients will be retrieved to the server and passed back to the mobile application to be displayed to the user. The user shall add these nutritional values to their daily intake or decide not to add the values to their daily nutritional intake count.

USE CASE 3	Phaze- Image Classification Feature	
Goal in Context	User logs into the Phaze mobile application.	
Scope	Functional requirements of a system	
Level	Medium Detail	
Success End Condition	The user gets the food nutrition for the picture that the user inputs.	
Failed End Condition	The user does not get the desired nutritional information for the inputted image of food.	
Primary, **Secondary Actors**	User. Online server system.	
Trigger	The user takes a picture of food.	
DESCRIPTION	**Step**	**Action**
	1	Food image is sent to the server

	2	Food image is classified by the food classification model.
	3	The classified name of the food is sent to the database.
	4	The food nutrition for the classified food image is sent back to the server
	5	The server processes the information
	6	Information is sent to the user's mobile device.
	7	The user accepts the nutritional values to be consumed and values are added to the overall daily food intake.
EXTENSIONS	**Step**	**Branching Action**
	2a	Food not identifiable : User has to input food name in input box
	2b	Food proportion not identifiable: User selects food proportions
	4	The food is not in the database: User has to input food name and nutritional value.
	7a	User decides not to consume the food and rejects it to add to the daily food intake count.

VARIATIONS		**Branching Action**
	1	Users login on different android operating systems

3.2.4 Bi-Weekly Check-In

This feature allows assessment of the user on how they are getting on with their suggested food regime with a simple survey. Based on the survey and record of their food intake we can make suggestions to edit the user's fitness regime according to their needs.

USE CASE 4	Phaze - User Bi-Weekly Check-In	
Goal in Context	User completes the bi-weekly automatic check-in	
Scope	Functional requirements of a system	
Level	Medium Detail	
Success End Condition	User completes the bi-weekly automatic check-in	
Failed End Condition	User cannot complete the a bi-weekly automatic check-in due to errors	
Primary, **Secondary Actors**	User. Mobile system.	
Trigger	The Phaze application has been activated and used for two weeks.	
DESCRIPTION	**Step**	**Action**

	1	Survey of questions is displayed to the user.
	2	Based on the responses of the user the Phase food regime will be adjusted accordingly
VARIATIONS		**Branching Action**
	1	Users login on different android operating systems

3.3 Performance Requirements

Phase application requires the application to be downloaded onto a supported android operating system. There will have to be upto 200mb of data available for the internal memory of the mobile application for storage of personal data and offline use. Inorder to make full use of the software features the mobile device must have a camera the highest resolution cameras possible for the food classification software to work accurately we would recommend any camera of 5MP or greater. All information processed by our software such as the image classification software will be completed in less than one second to reduce wait time so the customer can have a seamless experience. The server and database will be programmed to scale or reduce size depending on customer usage. This ensures that when there are surges in usage the system will not crash.

3.4 Logical Database Requirements

The use of an AWS Relational database will be used to ensure scalability when surges occur. Administrator login detail will be created to ensure we can control all aspects of the database. This includes updating the database with new food values, accepting food calories values to update the food calorie table as well as storing customer's username and passwords. We want to make our application ready for scaling so we

have created a feature that allows administrators to add and remove members from accessing the database . We also plan to give each user different permission that limits the control they have over the database. This ensures that if the application and team begin to employ new members the application will be able to facilitate this as well as removing departed older members.

We expect that our users will use this application at least three times a day for breakfast, lunch and dinner. This means that usage will surge at peak hour lunch times.

Any food that is not in our database will then request the user to input the food name and the calorie information. We envisage that the input of calories and food name may be wrong so that could be a constraint however with smarter algorithms to process the information such as a screening process will alleviate this problem.

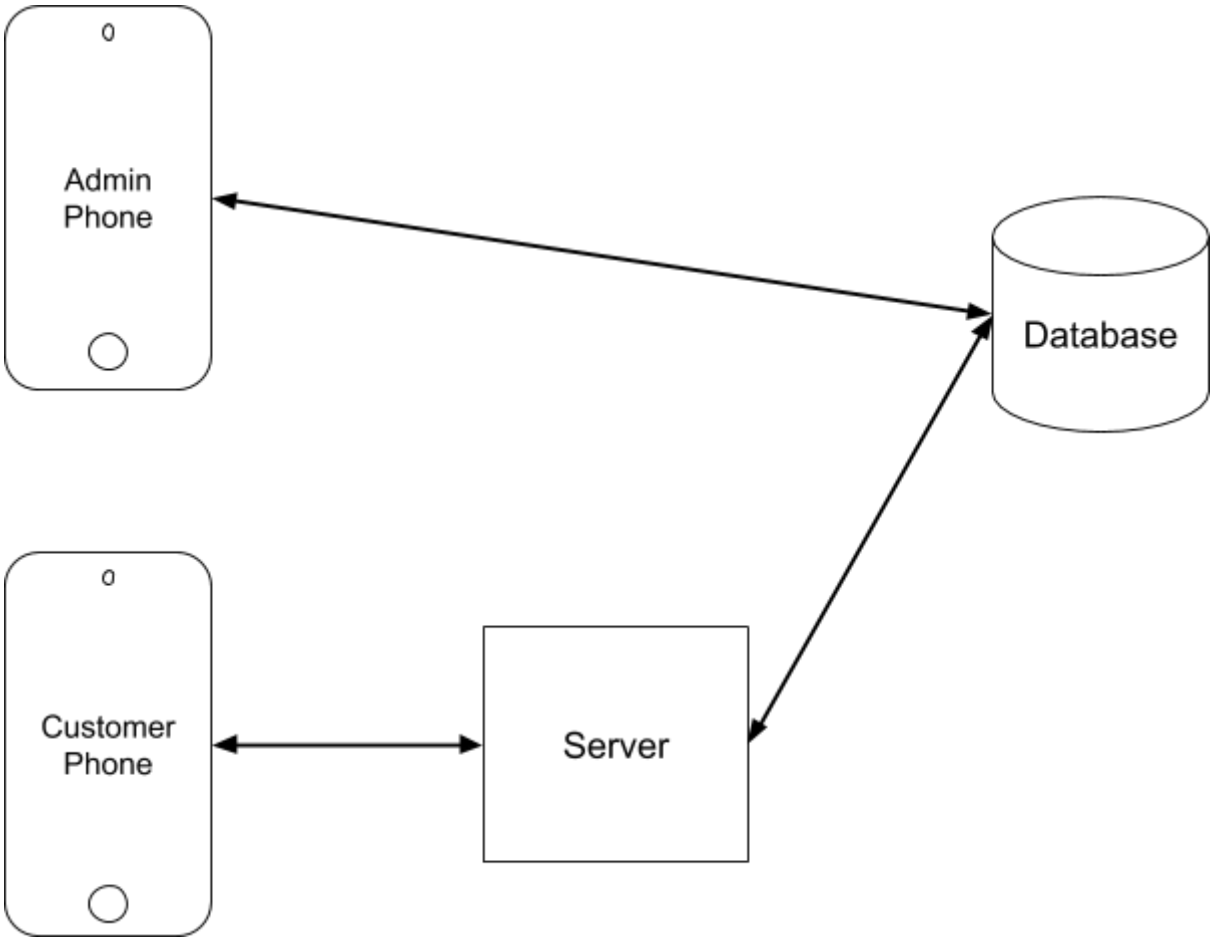

Figure 3. System's communication diagram

3.5 Design Constraints

Phaze contains several design constraints:

1. From initial planning we envisage that we will not be able to identify groups of foods together such as a plate with chicken and peas. This is because the accuracy significantly decreases causing inaccurate food classifications labels.

2. We have a limited training dataset of food, meaning that the AI model will not be able to cover all food types. This will be updated as we get access to more food image datasets.
3. Due to the lack of access to hardware resources, our AI model cannot be trained to incorporate all food types as it requires way more resources than what we currently have access to (12GB of RAM).

3.6 Software System Attributes

3.6.1 Reliability

Our AI model must be able to recognise food items in pictures with an accuracy of 85% or higher to be deemed successful. If it cannot recognise a food item, the user will have the option to search for it manually.

3.6.2 Availability

The application will allow its users to restart the software after failure with a loss of at most their picture taken and its corresponding nutritional information.

3.6.3 Security

All data being sent between the different systems will be encrypted.

3.6.4 Maintainability

Our AI model will be continuously trained using new data acquired from the users.

3.6.5 Portability

Our device will run on 100% of Android devices.

3.7 Organizing the Specific Requirements

3.7.1 System Mode

We plan to design our mobile application so that only users that have an account can access all features of the application. This offers security to our system as if an attempt to flood our database occurs we will be able to locate the user and suspend or ban the user from using the application based.

3.7.2 User Class

Our software tailors the functionality according to the user by an initial onboarding process such as an initial survey that will determine the best fitness regime for the user. The main grouping are a fitness regime to:

1. Reduce the calorie of the user.
2. Maintain the calorie intake of the user.
3. Increase the calorie intake of the user.

3.7.3 Feature

When a user completes a milestone or goal for their fitness regime they will be rewarded. We would like to contact companies to offer discounts and vouchers to users as rewards such as fitness companies such as MyProtein that could off reduce their highly demanded products. However, since we are only in the product design phase we do not have a business plan to pitch to sponsors to give us discounts and vouchers.

3.7.4 Stimulus

Our reward system will gamify their fitness regime. Each daunting fitness milestone will be a challenge and every time the challenge is overcome the user will get a virtual badge. This incentivises the user to continue using the app as it will be fun and enjoyable to use.

3. 7.5 Response

This feature will encourage the user to push themselves to their limits without injuring themselves. As humans we all love games so if our application can play to this human characteristic we will be able to make a challenging undesired exercise task into something that is fun to do.

4. Change Management Process

In order to manage changes made to the requirements of Phaze, we will implement an agile development cycle as it is both incremental and iterative.

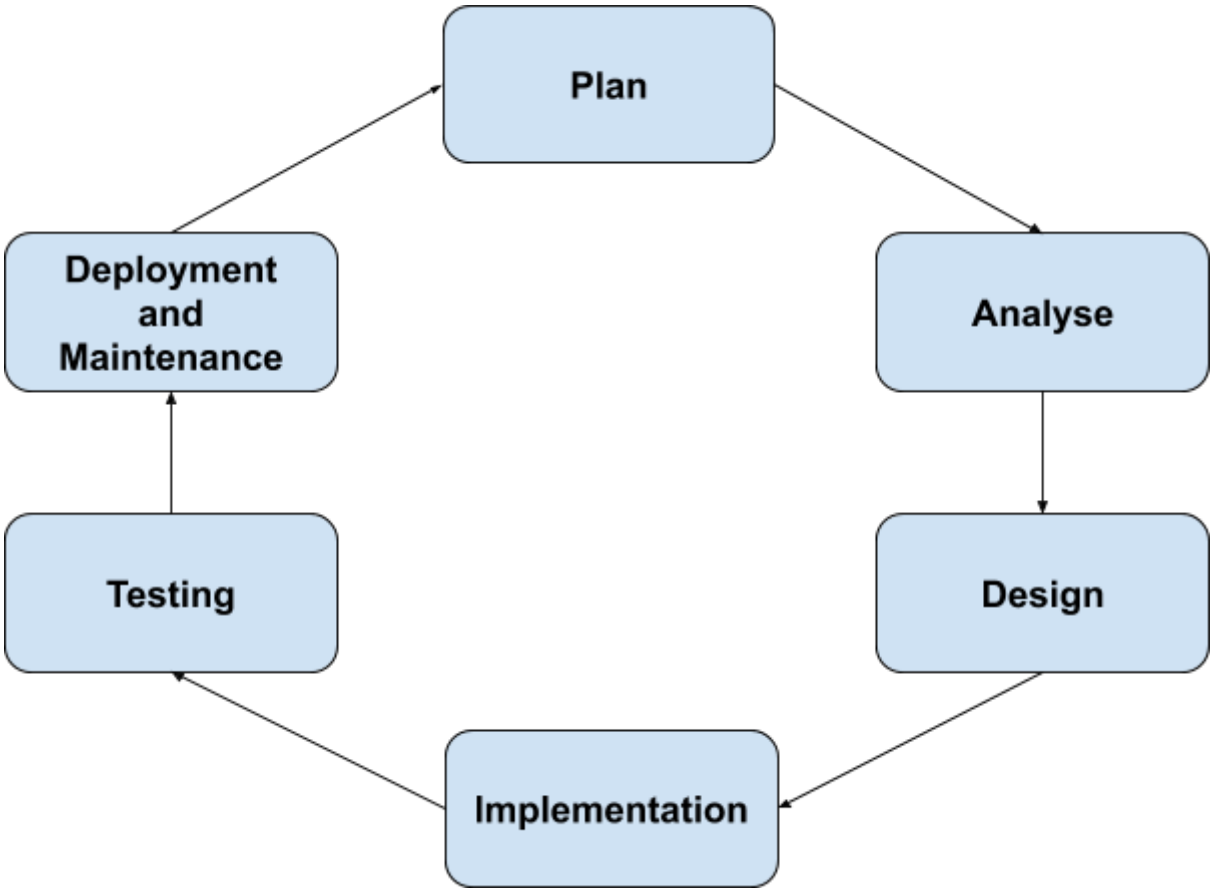

Figure 4. Agile method diagram

This model has been chosen due to the size of our product and team. Agile is best suited for smaller teams and after the completion of each sprint, we will be able to re-adjust our SRS. Changes in requirements can be submitted using the team email address.

5. Document Approvals

Identify the approvers of the SRS document. Approver name, signature, and date should be used.

6. Supporting Information

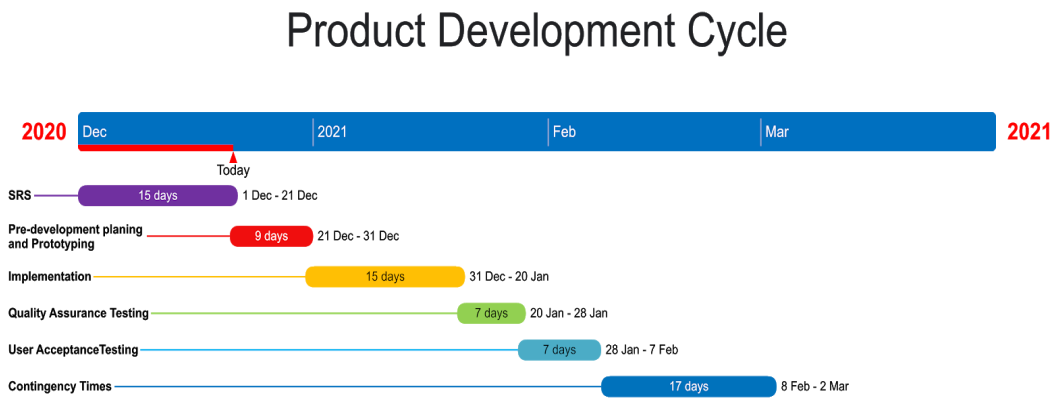

Figure 5. Product Development Cycle diagram