**Senior Design Project**

**Healthscope**

*High Level Design Report*

Cüneyt Erem

Doğukan Ömer Gür

Kaan Kale

Melih Sancak

Mert Gürcan

**Advisor:** A. Ercüment Çiçek

**Jury Members:** Selim Aksoy, Özcan Öztürk

May 14, 2017

This report is submitted to the Department of Computer Engineering of Bilkent

University in partial fulfilment of the requirements of the Senior Design Project

course CS491/2

Table of Contents

1. Introduction............................................................................................................ 3

1.1. Purpose of the System………………………………………………………. 3

1.2. Design Goals............................................................................................ 3

1.2.1. Reliability..................................................................................... 3

1.2.2. Availability................................................................................... 3

1.2.3. Usability....................................................................................... 4

1.2.4. Portability.................................................................................... 4

1.2.5. Performance................................................................................ 4

1.2.6. Extensibility................................................................................. 4

1.2.7. Maintainability............................................................................. 4

1.3. Definitions................................................................................................. 4

1.4. Overview................................................................................................... 5

2.Current Software Architecture …............................................................................ 5

3. Proposed Software Architecture............................................................................. 6

3.1. Overview .................................................................................................. 6

3.2. Subsystem Decomposition....................................................................... 7

3.3. Hardware Software Mapping ................................................................... 9

3.4. Persistent Data Management .................................................................. 9

3.5. Access Control and Security .................................................................. 10

3.6. Global Software Control ……………………………………………………. 10

3.7. Boundary Conditions ……………………………………………………….. 11

4. Subsystem Services ………………………………………………………………….. 12

4.1. GUI Layer.......................................................................................................... 12

4.2. Implemantation Layer ....................................................................................... 12

4.2.1. Data Evaluation Service …………………………………………………. 12

4.2.2. Data Management Service ………………………………………………. 12

4.2.3. User Management Service ………………………………………………. 12

4.2.4. Barcode Scanner Service ..…………………………………………….... 13

4.3. Service Layer …................................................................................................ 14

4.3.1. Authentication Services .….……………………………………………… 14

4.3.2. Connection Services ……...……………………………………………… 14

4.4. Database Layer ................................................................................................ 14

4.4.1. Database Service ……………………………………………………….... 14

5. References .......................................................................................................... 14

1. Introduction

In this final report, high-level design will be given by explaining the design goals, software architecture and subsystem services.

1.1. Purpose of the System

Healthscope is java based project, running on the Android platform, aims to make people more aware of impact of the products on their health. When customer do shopping, Healthscope identifies the product that customer like and tell him/her whether it is a healthy choice or not according to the customers’ health status which is reflected as codes in our app. As a result, this project would help customers to fill their shopping carts with more healthy foods that improve their medical conditions. Our “Healthscope” application helps user to shop on supermarkets with aware of their medical conditions. Users create their own profile on this application. There are fields for their age, physical appearances (height, weight), diseases and allergies. And our “Healthscope” doctor evaluate every food and their ingredients in the supermarket for our users. There are five levels for every ingredients agreeableness (white, yellow, orange brown, red). It is very easy to use because users just read the barcodes of each food they want to buy with their androids, ingredients and agreeableness will appear on their screen and they can see appropriateness respect to their medical conditions.

1.2. Design Goals

1.2.1. Reliability

If the system give any type of failure, HealthScope should give error message to user and developer to fix the failure quickly as possible.

1.2.2. Availability

The application will be free on Google Play Store. Any user, who has android devices, can easily access and use the application. However, our application is not available for iOS user for now; in the future it will also be available for iOS users.

1.2.3. Usability

Since the application is designed to easily use from users, it has to be also easy to learn for them. To make the application easier, UI and design are kept simple.

1.2.4. Portability

HealthScope is a mobile application on Android. So portability is not an issue for our application. Since every mobile Android devices will be able to use the application, and devices are portable; we do not have to consider portability issue.

1.2.5. Performance

The system should be efficient and give fast response for user request. When user scans any food product, waiting time for server answer should not be more than 2 seconds.

1.2.6. Extensibility

HealthScope application needs to be compatible with the future releases of the application. The future releases or extensions are generally likely to adding new features or improving the existing features of the application.

1.2.7. Maintainability

Since our application is extensible, we need to deal with the changes and meet the new requirements.

1.3. Definitions, Acronyms And Abbrevations

API:Application Programming Interface

GUI: Graphical User Interface

Android:Operating system that is easy to use

Google Play Store:Android application market

1.4. Overview

Healtscope aims to help users to be aware of what will they eat recording to their medical conditions when they are shopping in markets. Our main purpose to have more conscious human beings and healthier bodies. To use this app user must just download it and create their own profiles, while creating they must enter their gender, age, height, weight, diseases and allergies for proper usage.Users will see each food ingredient and levels of recommendation on their screen by reading the barcodes of goods. For our users we agreed with doctors. There are five levels for every ingredients agreeableness (white, yellow, orange brown, red) and doctors scale every food ingredient for each disease and allergies.

2. Current software architecture (if any)

There are some barcode applications in google play store and ios appstore. But this application is different from them because almost all of the apps are designed to scan the barcode and show the barcode information by connecting to database. This app connects to the database, after receive the data, it uses user profile information and database product information. After the calculation by using these data, app introduce the suggestion information that user should buy the item or not. Therefore, there is no similar app like this app.



3. Proposed software architecture

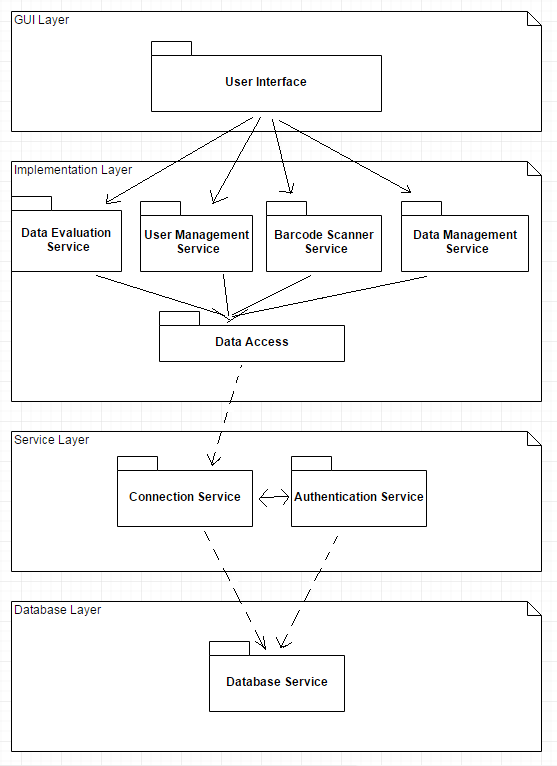
3.1. Overview

Healthscope is Android based app that aims to make people more aware of impact of the products on their health by scanning product's’ barcode and giving a brief information about products according to their health status. Project involves a separate client and server system, and a connection to Database in order to get product information.Because of these reasons, Server/Client architecture style is the best for our application.

Also, we can easily say that, because of these communication flow between user interface and data, it is best to use MVC pattern so we can tie these pieces together and reduce the amount of coding and improve the performance

3.2. Subsystem Decomposition

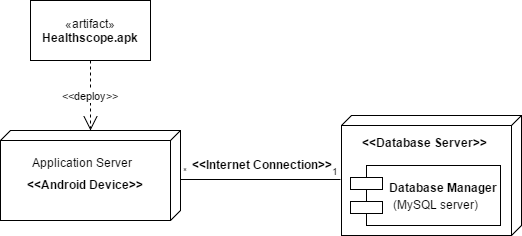
Our program consists four layer, which are GUI Layer, Implementation Layer Service Layer and Database Layer.GUI layer consists User Interface which is depend on implementation layer services which are Data evaluation service, User management service,barcode scanner service and Data Management service.Implementation layer also consists data access which calls on service layer services. Service Layer consist of connection service and authentication service. Also these services are depend on eachother and calls on database service which is a part of database layer.Detailed explanation of these services provided on Seciton 4. Figure is provided below.



*Subsystem Decomposition*

3.3. Hardware-Software Mapping

Our application will be developed for the Android operating system therefore our users should use the Google Play Store to get Healthscope. For using Healthscope the Android device our client uses should have a camera since our application will be used by scanning the barcodes of products. In addition to that our application requires internet connection for communication with our database. Healthscope will uses a database server to keep user and product information. We will run a MySQL server to serve our services to users. Since it’s an Android program we will use java to write our program since Android supports it. The final product will be an APK (Android Application File) and this file can be used for distribution and installation for our application.



*Hardware/Software Mapping*

3.4. Persistent Data Management

Healthscope an application that rely on database and a persistent database is crucial for our product. The database will keep the all information about users like name, email and password etc. and all the product information will be kept in database and we use this information when we match scanned product with products that in database. Our database will have a backup in case of loss of information also Database should work as fast as possible to decrease waiting time of our users. We try to implement fastest queries and algorithms and organize our database the most efficient way .In other words the database directly affect the performance of our application to maximize our efficiency we try to make our MySQL serves as optimized as possible.

3.5. Access Control and Security

To use Healthscope all users must be register first without registration and login no user can use the properties of our application. For registering a user should determine a password for themselves and they will log in with their unique email and password. Since each user will login using their password their password should be hashed maybe salted and stored in the protected server. No user can be able to see other users’ personal information and no two users can have the same email address. After we match their password and emails with our database we will let them use our application and they only can see the information that are public.

3.6. Global Software Control

To use the user program, you must first create an account. It is necessary to connect to the database server in order to read the codes that the user has scanned. Therefore, server-client architecture is used. Once the application reads the barcode code, it connects to the server and requests information from the database. GUI Layer is connected to implementation layer. Barcode Scanner service connected to database service and it connects to connection service. It also connects to database service in database layer. These exchanges between the layers compose the global software control.

3.7. Boundary conditions

Initialization:

· When user stars our application for the first time they will welcomed with a login page which contain a “login” and “register” button.

· If user already have an account they will enter their email and password and hit the “login” button.

· If they have not an account they can hit the “register” button and can create an account via filling the required spaces and press the “create” button.

· After they login successfully our home page will be shown and user can use our application freely.

Termination:

· Users can terminate the application by simply using Androids phones terminate features like pressing home button then swipe right the application.

· User can use exit button to terminate the application they are asked if they are sure or not to leave.

· They can also log out and close the application but they are asked to log in again when they use it again.

Failure:

· Internet connection is required to use Healthscope if connection is not available failure may occur.

· Our applications will be written for Android 4.4 and later versions. If users’ device has an outdated version failure may occur.

· The application needs a camera to scan barcodes if there is not one on the device a failure may occur.

In addition to these conditions failures may occur on database and our users cannot use the Healthscope properly in these conditions we will notify our users via email or other notifications. Also while maintaining the servers they may face failures but they will notify for these situations also.

4. Subsystem Services

4.1. GUI Layer

GUI layer services focuses on visuazilitain of Healtscope app.User Interface consist one main GUI classes which controls other subclasses.These subclasses implements different app layouts, such as login screen, User Info screen.

4.2. Implementation Layer

4.2.1. Data Evaluation Service

User Management Service; This service used by user interface and data access in the implementation layer. This service receive the food production data to be used and also user data to use both of them to conclude the result of the suggestion information.

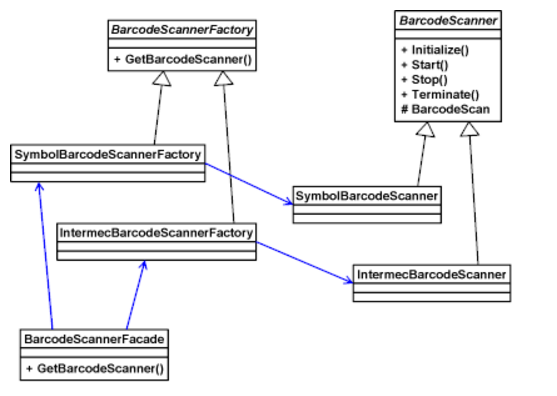
4.2.2. Data Management Service

It manages the data that taken from database and it regulates the data and keeps them to be used for the other works if food production details are necessary. This service does not keep the all of the food production data, it only keeps the data when user scan the barcode and barcode data is needed for a while.

4.2.3. User Management Service

It regulates the user profiles and if new user profiles are need to add, it manages these features as well. Also, if user profile information is need to be used such as Data evaluation services needs them, it is conneced to that service and share the user information.

4.2.4 Barcode Scanner Service



Barcode Scanner Service consists of some features. There are two generic abstract barcode scanner class Factory base class, BarcodeScannerFactory and bar code scanner base class, BarcodeScanner. Certain barcode scanner classes inherit these classes. BarcodeScannerFacade decides which device works. The facade class is responsible for bringing back the right type of bar code scanner object. The customer code calls the static method GetBarcodeScanner in the front class BarcodeScannerFacade. At the same time, the code acquires a factory instance depending on the device on which the device works. The facade class calls the GetBarcodeScanner method on that factory to get an example of the correct barcode scanner.

4.3. Service Layer

4.3.1. Authentication Services

Authentication service is using Authorization Manger and checks that if the email and password that users give match with the login information that we store in the database.

4.3.2. Connection Services

Connections services working on connections problems of the operation. For communication between database and users. We use connection services for client-server side also our barcode scanner use connection to match barcodes that scanned with products that in the database.

4.4. Database Layer

4.4.1.Database Service

Database Services is used to perform all database operations. All the data (name, age, gender etc.) of the user are working with Database Production Services and also all of food products and their properties database services. The app connects to web to access database provided by API.

5. References

https://developer.android.com/studio/intro/index.html/, Online, Accessed: 13.05.2017

https://dev.mysql.com/doc/, Online, Accessed: 13.05.2017

https://github.com/zxing/zxing/, Online, Accessed: 13.05.2017

<https://ndb.nal.usda.gov/ndb/doc/index>/, Online, Accessed: 12.05.2017

https://github.com/, Online, Accessed: 13.05.2017

<https://msdn.microsoft.com/en-us/library/ee658117.aspx#ClientServerStyle>, Online Accessed : 13.05.2017