COMSATS University, Islamabad Pakistan

Smart Diabetolog

By

Fatima Shahzad **CIIT/FA19-BSE-081/ISB**

Kashaf Fatima **CIIT/FA19-BSE-052/ISB**

Supervisor

Atique Ahmad Zafar

Bachelor of Science in Software Engineering (2019-2023)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

COMSATS University, Islamabad Pakistan

Smart Diabetolog

A project presented to
COMSATS University, Islamabad

In partial fulfillment
of the requirement for the degree of

Bachelor of Science in Software Engineering (2019-2023)

By

Fatima Shahzad **CIIT/FA19-BSE-081/ISB**
Kashaf Fatima **CIIT/FA19-BSE-052/ISB**

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other, we will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Kashaf Fatima   Fatima Shahzad

--------------------------   --------------------------

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (CS) Dukan was developed by **Kashaf Fatima (CIIT/FA19-BSE-052)** and **Fatima Shahzad (CIIT/FA19-BSE-080)** under the supervision of Atique Ahmad Zafar and that in his opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Software Engineering.

Supervisor

External Examiner

Head of Department

(Department of Computer Science)

Executive Summary

The diabetic patients have always been facing different symptoms which eventually grow to a life-threatening situation. The patients may have low or high sugar levels at times but are unaware of measures they need to take. Sometimes, the patient might consume food that eventually results in hyperglycemia without its awareness. Moreover, diabetic patients may indulge themselves in some physical activity for maintaining their glucose level. They may experience some allergy or side effects related to any oral or injectable diabetes medications they are consuming. In most cases people are unaware of the existence of such reactions which results in being fatal in some cases.

To prevent all these problems and help diabetes patients, Smart Diabetolog is developed. It is a mobile application to help diabetes patients maintain their lifestyle in a healthy way by providing them with meals and exercise plans.

The application suggests the diet chart and exercise plan to keep the calories, blood sugar and other health factors in control by keeping track of diabetic patient’s age, his BMI and health factor readings. The diet chart defines the quantity and type of food the user can have. Moreover, the diet chart and exercise plan get updated each time the health factor fluctuates from normal, or the user consumes extra calories. The system allows the user to record its medicine prescription and allergic reactions. Every time the user adds a new medication or meal intake, the system alerts the user about any possible allergic reaction it can face based on its allergic reaction history. Moreover, the system allows the user to detect retinopathy by providing retinal fundus image.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor Mr. Atique Ahmad Zafar. Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are grateful to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us with the values of honesty & hard work.

Kashaf Fatima Fatima Shahzad

-------------------------- --------------------------

Abbreviations

SRS	Software Require Specification
PC	Personal Computer
SDS	Software Design Specification
DSA	Data Structures and Analysis

Table of Contents

List of Figures

List of Table

1 Introduction

Smart Diabetolog is a mobile application for diabetes patients. It helps diabetic patients to track and record its sugar level along with blood pressure and cholesterol, their medical prescriptions, adverse medical reactions they faced in the past, and the calories intake they had in a day. The meal and exercise plan are provided according to his blood sugar and other health factors provided. Moreover, the meal plan will be made by taking in view any allergy details provided by patient. Live chat will be provided for patients to interact with each other. Discussion boards will also be provided where patients will be able to send their queries and other patient will be able to reply to these queries. Following are the 11 modules of Smart Diabetolog application:

1. **Authentication:**
   This module is responsible for authentication and verification of user accounts.
2. **User Management:**
   This module is responsible for maintaining user profiles.
3. **Tracker:**
   This module is responsible for tracking patients' blood sugar and other health factors.
4. **Medication:**
   This module is responsible for maintaining patient medication prescriptions and setting up reminders according to user medication.
5. **Diet chart:**
   This module is responsible for creating meal plans and updating it according to patient's updated heath factors.
6. **Physical Activity:**
   This module is responsible for creating exercise plans and updating it according to patients updated health factors.
7. **Allergic reactions:**
   This module is responsible for maintaining a patient's allergy reaction history regarding food ingredients and medications. It also sets medication reminders for patients.
8. **Image Processing:**
   This module is responsible for analyzing the food image taken from patient suggesting its name and calories through deep learning. It also recognizes the name of medication by picture using OCR.
9. **Diabetes Awareness:**
   This module is responsible for sharing the latest diabetes blogs and videos to keep him up to date with diabetes.
10. **Chat box:**
    This module is responsible for allowing patients to chat to other patients through live chat and to send his latest queries, where other patients will be able to reply answering his queries.

11. Retinopathy:

This module is responsible for detecting retinopathy in diabetes patients which is an eye disease and cause blindness with time in diabetic patients. It uses deep learning to analyze patient's retina fundus images and detect retinopathy.

1.1 Vision Statement

For Diabetic patient who want to maintain their lifestyle by adopting healthy and preventing measures. The Smart Diabetolog is an Internet based Smart phone Application for both IOS and Android that will help the diabetic patient to maintain his medical history, his calories intake in a day, his current medication prescriptions and adopt a routine that keeps him healthy. Unlike today's scenario where each diabetic patient has to make frequent visits to his doctor, nutritionist, and fitness trainer to maintain their lifestyle. Our product allows the diabetic patient to keep his glucose level and calories balanced to live a normal life by merely recording his sugar level, blood pressure, cholesterol and his food intake. Then on the basis of these records the application suggests diet chart and exercise plan to the patient.

1.2 Related System Analysis/Literature Review

Table 1-1 displays the comparisons of the proposed solution with the already present related system.

Table 1-1: Related System Analysis

Application Name	Weakness	Proposed Project Solution
Diabetes:M	1. The application uses CGM (Continuous Glucose Monitoring) devices connected through Bluetooth to application for glucose level detection of user. 2. For showing recoded information they use medical terms (like bolus) which are not understood by lay man. 3. The application only supports English and Spanish food nutrients data. 4. The application is not user friendly because data calculated is either shown in different tabs or not even shown.	1. The application would allow the user to even add his present glucose sugar level manually. 2. The application would use all the basic medical terms along with the term's description. 3. The application would be a learning agent supporting different foods and cuisines. 4. After each meal the data about glucose sugar, calories, carbs and proteins taken along with the proportion of insulin to use would be displayed at one particular point.

Blood Glucose Tracker	1. The application tracks only blood sugar. 2. It keeps the history of blood sugar but does not utilize it.	1. Our application keeps the track of patient blood sugar, blood pressure and cholesterol. In reality, these conditions exist independently but when a patient has blood sugar and is also affected by blood pressure and cholesterol, then these two conditions are severely affected by the blood sugar. 2. Our application utilizes the health factors history by making diet plan and exercise plans to maintain his health.
7 days Diabetes Meal plan	1. The application suggests meals recipes for diabetes patients without taking their blood glucose concentration. 2. The application only suggests some of the recipes they patients may not like or could not eat due to any allergy or tolerance.	1. Our application suggests meals for diabetes patients according to their health factors. 2. Our application allows diabetes patient to make their customized list of favorite recipes. These recipes will be accommodated in their meal plans as per their health factors conditions.
AI Nutrition	1. This is a general app for workout does not consider patients risk factors. 2. Exercises are just listed in text form.	1. Our application considers diabetes patient blood sugar concentration and update their exercise plan to maintain his health. 2. Exercises will be shown along with timer, so patient knows accurately how long they need to work out.
MySugar	1. Only tracks the blood sugar does not show any alerts due to hyperglycemia or hypoglycemia. 2. Keep records and show statistics.	1. Our application tracks blood sugar and shows alert when blood sugar concentration fluctuates. 2. Our application shows statistics, and also utilize them by making customized meal and diet plan.

mySugr	1. The application requires great amount of manual data entry. 2. The application is very cluttered. 3. No functionality available for calculating. carbs, proteins, calories in the food.	1. The functionality would be given where data entry could be reduced by using image detection. 2. Every information would be properly managed and separated under different categories with labels. 3. Functionality would be available where each taken food would be displayed along with its calories, carbs, protein and effect on out glucose sugar.

1.3 Project Deliverables

- **Project Scope Document**
  This document would contain the scope and proposal of Smart Diabetolog Application.
  Date to deliver: 14 April 2022

- **Software Requirement Specification Document (SRS)**
  This SRS document would include all the FRs and NFRs of the proposed solution.
  Date to deliver: 7 October 2022

- **Software Design Specification (SDS) with 30% project implementation**
  SDS document would include Smart Diabetolog app's design methodology and software process model, along with system overview, design models, data design and algorithm. The project 30% implementation would also be shown.
  Date to deliver: 28 November 2022

- **Half final project Report with 60% project implementation**
  Final project report would contain all the previous documents information, along with that the 60% project implementation of Smart Diabetolog application will be shown.
  Date to deliver: 20 March 2023

- **Final project Report with 100% project implementation**
  The completed final project report along with the complete software product of the proposed solution will be shown.
  Date to deliver: 15 May 2023

1.4 System Limitations/Constraints

LI-1: The system takes the record for all current medication from the user but the reaction between any diabetic medicine with other non-diabetic medicine would not be warned.
LI-2: The system could only image process the food system is trained for.
LI-3: The food must be on white plate for its calories to be estimated.
LI-3: The people may be biased towards not trusting a mobile application for their diabetes control.
LI-4: None of the medical conditions (cholesterol, blood pressure) is catered along with diabetes.

1.5 Tools and Technologies

Table 1-2 contains all the tools and technologies used to build the application.

Table 1-2: Tools and Technologies

Tools and Technologies	**Tools**	**Version**	**Rationale**
	Android Studio	2021	IDE
	Mongodb Atlas	2020	DBMS
	Visual Paradigm	16.3	Modeling
	Microsoft word	2019	Documentation
	Jupyter Notebook	5.0	Date analysis
	Google collab		Deep learning model training
	PyCharm	2021.1. 2	For creating FastAPI
	Scrapy	2.8.0	For scrapping websites
	VS code	1.75	IDE
	Technology	**Version**	**Rationale**
	ReactNative	0.70	Front-end Development
	Python	3.10.2	Deep Learning, Machine Learning, Fast API
	Express	4.18.2	Server (Backend)
	Node.js		Server (Backend)

1.6 Relevance to Course Modules

The table 0-1 contains all the subjects that are related to the modules of Smart Diabetolog Modules:

Table 0-1: Relevance to Course Modules

Courses	Relevant Modules
Introduction to software engineering	Helped in UML modeling in documents
Software requirement engineering	Helped in making SRS document
Data structures and algorithms	Helped in managing data
Communication skills	Helped in presenting our project
Database system	Helped in managing database for our system
Software design and architecture	Helped in creating architectural design according to our application
Software quality engineering	Helped in quality assessment of our project
Mobile application development	Helped in learning React Native
Topic in software engineering 1	Helped in learning Express, Node.js
Software testing	Helped in testing our project requirements

Problem Definition

The following chapter contains the information about the problem and its proposed solution.

1.7 Problem Statement

Two initial steps to maintain a glucose level in body revolves around proper diet plan and exercise. All the lay diabetic patients never have a proper knowledge about how they can maintain their diabetes by taking the proper meals with correct proportion of carbohydrates that break downs into glucose sugar. Additionally, people having low exercise awareness, when start exercising they often begin with high intensity exercise thinking that might be more effective, but the case is reversed by resulting in sugar-level increase or decrease. The patients of diabetes type-1 and few type-2 are often involved in oral or injectable medications, they are never aware of what side effects and adverse reaction their medication may cause. Moreover, the people might not realize that their body is allergic to some active ingredients present in the medication they are taking which often results being fatal.

The diabetic patient may also find it hectic to add meal details correctly every time he eats by typing them. Moreover, diabetic patients have no idea how worsen diabetes makes their blood pressure and cholesterol, if they are suffering from any of these two conditions.

1.8 Problem Solution

The application helps to maintain a proper lifestyle for any diabetic patient by reminding the user when to take which physical activity for what duration, when they are allowed to take what type of meal, in which quantity and at what time of the day, what their sleeping schedule should be. These details would be evaluated on the basis of sugar history regularly updated by the user manually. The application assists the diabetic patient by providing a proper diet plan according to current sugar level history along with blood pressure and cholesterol. The plan provides the details of starchy (carbohydrate with high calories and low fiber), non-starchy (carbohydrate with low calories and high fiber) and lean proteins (proteins with low calories and low fat) that the person can take. The application provides a guideline what necessary physical activity will result in burning calories which lowers the glucose level in body thus maintaining the sugar level. The diet and physical activity are suggested to the individual according to their BMI, age, gender, current sugar level concentration. The application aware the patient about any side-effects and adverse reaction. The application also keeps the medical record to evaluate that the patient might be allergic to which active agent then showing the warning to the patient related to the reaction in the past if he takes the medicine containing similar active agent. The system also indicates the cross sensitivity (the medication with similar formula) to patients. The application facilitates the patient to record his meal history in the system by taking pictures of the food he is consuming, whose details would be analyzed by the system itself. For past suffered reactions the user has to give medicine name and select the experienced symptoms he had. Moreover, the system also suggested the measure the patient should take to balance the blood sugar if he is suffering either from hyperglycemia or hypoglycemia.

1.9 Scope

The Smart Diabetelog helps a diabetic patient to track and record its sugar level along with blood-pressure and cholesterol, their medical prescriptions, adverse medical reactions they faced in the past, and the calories intake they had in a day.

The system suggests the diet chart and exercise plan to keep the calories, blood sugar and other health factors in control by keeping track of his age, his BMI and health factor readings. The diet chart defines the quantity and type of food the user can have. Moreover, the diet chart and exercise plan get updated each time the health factor fluctuates from normal, or the user consumes extra calories.

Additionally, the exercise plan would also be suggested with the type of exercise or activity, with duration that would help patients burn the correct proportion of their gained calories so that their glucose level remains baselined.

The system allows the user to record its medicine prescription and allergic reactions. Every time the user adds a new medication or meal intake, the system alerts the user about any possible allergic reaction it can face based on its allergic reaction history. Moreover, the system allows the user to detect retinopathy by providing retinal image.

1.10 Modules

Following are the modules for Smart Diabetolog Application.

1.10.1 Module 1: Authentication

FE-1: Register user Accounts.
FE-2: Verify user provided email.
FE-3: Login user's account by verifying them.

1.10.2 Module 2: User Management

FE-1: Maintains the types of users (diabetic type-1 and diabetic type-2).
FE-2: Register and manage account details.
FE-3: User can update their profile details.
FE-4: User can delete their account.
FE-5: User can customize their health factors tracking.

1.10.3 Module 3: Tracker

FE-1: Add or update blood-sugar level details.
FE-2: Add or update blood-pressure details.
FE-3: Add or update cholesterol details.
FE-4: View Health Factor's Statistics.
FE-5: Determines the food which causes hyperglycemia.
FE-6: Determines the exercise or activity which causes hypoglycemia.
FE-7: Updates the average, minimum and maximum blood sugar reading of the user.
FE-8: Updates the bolus insulin units for the user.
FE-9: Generates alerts along with measures whenever blood-sugar is too high or too low.

1.10.4 Module 4: Medication

FE-1: Add a new medication prescription.
FE-2: Add, Update, or Delete any oral medication instance in medication prescription.
FE-3: Add, Update, or Delete any insulin instance in medication prescription.
FE-4: View a Medication Prescription.
FE-5: Reminds the user about their oral medication consumption in correct proportion on correct time.
FE-6: Reminds the user about their long-acting insulin injection before going to bed.
FE-7: Maintains a record for medication prescriptions.
FE-8: Recognize the name of the medication through its picture by using optical character recognition.

1.10.5 Module 5: Diet Chart

FE-1: Add new meal information.
FE-2: Create Diet plan and meal history.
FE-3: Calculates the ideal number of calories allowed to patient.
FE-4: Suggests food according to allowed calories count along with its quantity using machine learning model.

FE-5: Update the diet plan of the patient according to his calories or blood sugar concentration using machine learning model.
FE-6: Generate warning for wrong calories intake which deviates the sugar level of the user.
FE-7: Analyze meal image through deep leaning model and suggest names of the dish.
FE-8: Calculate calories from the meal image of the selected dish by image processing and divide into carbohydrates, fats and proteins.

1.10.6 Module 6: Physical Activity

FE-1: Create and update exercise plan using machine learning model.
FE-2: Initiate today's exercise.
FE-3: Create and update exercise history.
FE-4: Update the exercise plan of the patient according to his calories or blood sugar concentration using machine learning model.

1.10.7 Module 7: Allergic Reactions

FE-1: Add and update any experienced oral medication allergic reaction or side effect along with their experienced symptoms.
FE-2: Add and update any food allergies along with their experienced symptoms.
FE-3: Generate a record for patient's allergic active ingredients in medications.
FE-4: Provide alerts and warnings to the patient regarding their current medication side effects and adverse reactions.
FE-5: Provide alerts to the patient if the food they are going to consume may have any ingredient they are allergic to.
FE-6: Prepares medication cross-sensitivity (alternative medication with similar formula) for patients.
.

1.10.8 Module 8: Diabetic Awareness

FE-1: Share latest blogs and articles of diabetes experts.
FE-2: Share latest videos and interviews related to diabetes.

1.10.9 Module 9: Chat Box

FE-1: Provides discussion board for diabetic patient queries.
FE-2: Provides live group chats for diabetic patients to share their experience.
FE-3: Recommends the query search using machine learning model.

1.10.10 Module 10: Retinopathy detection

FE-1: detects blood vessel damage through deep learning model.
FE-2: determine the retinopathy stage using deep learning model.

2 Requirement Analysis

This section includes the functional, non-functional, hardware and software requirements of the proposed solution along with the techniques used to identify them.

2.1 User classes and characteristics

Table 2-1 contains the information regarding the class or role of the application.

Table 2-1: User classes and characteristics

User class	Description
Diabetic patient	A diabetic patient needs to maintain their lifestyle by keeping track of their health factors by maintaining their diet and exercise routine in order to control their blood sugar.

2.2 Requirement Identifying Technique

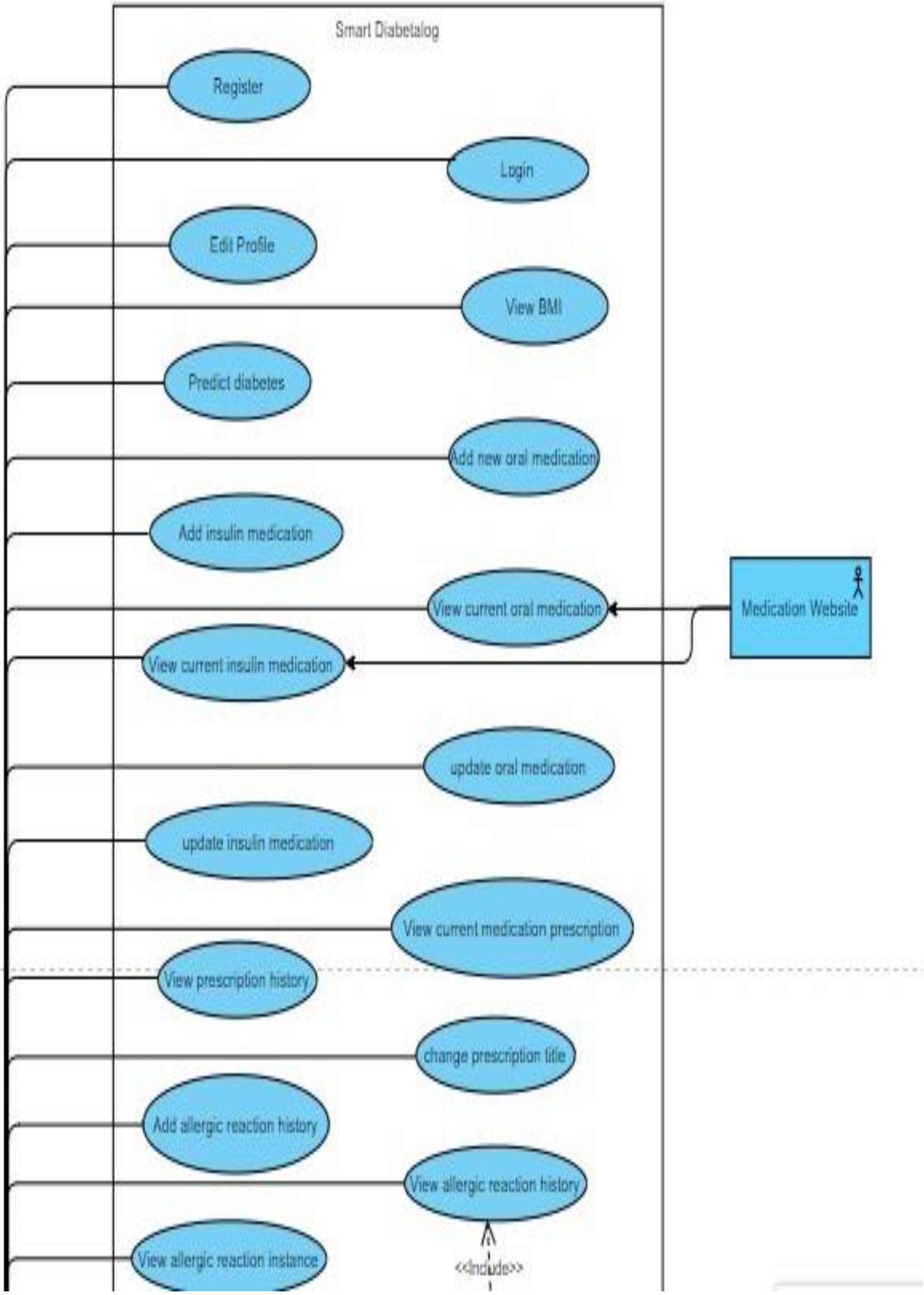
According to the requirements of our project the following two techniques are selected.

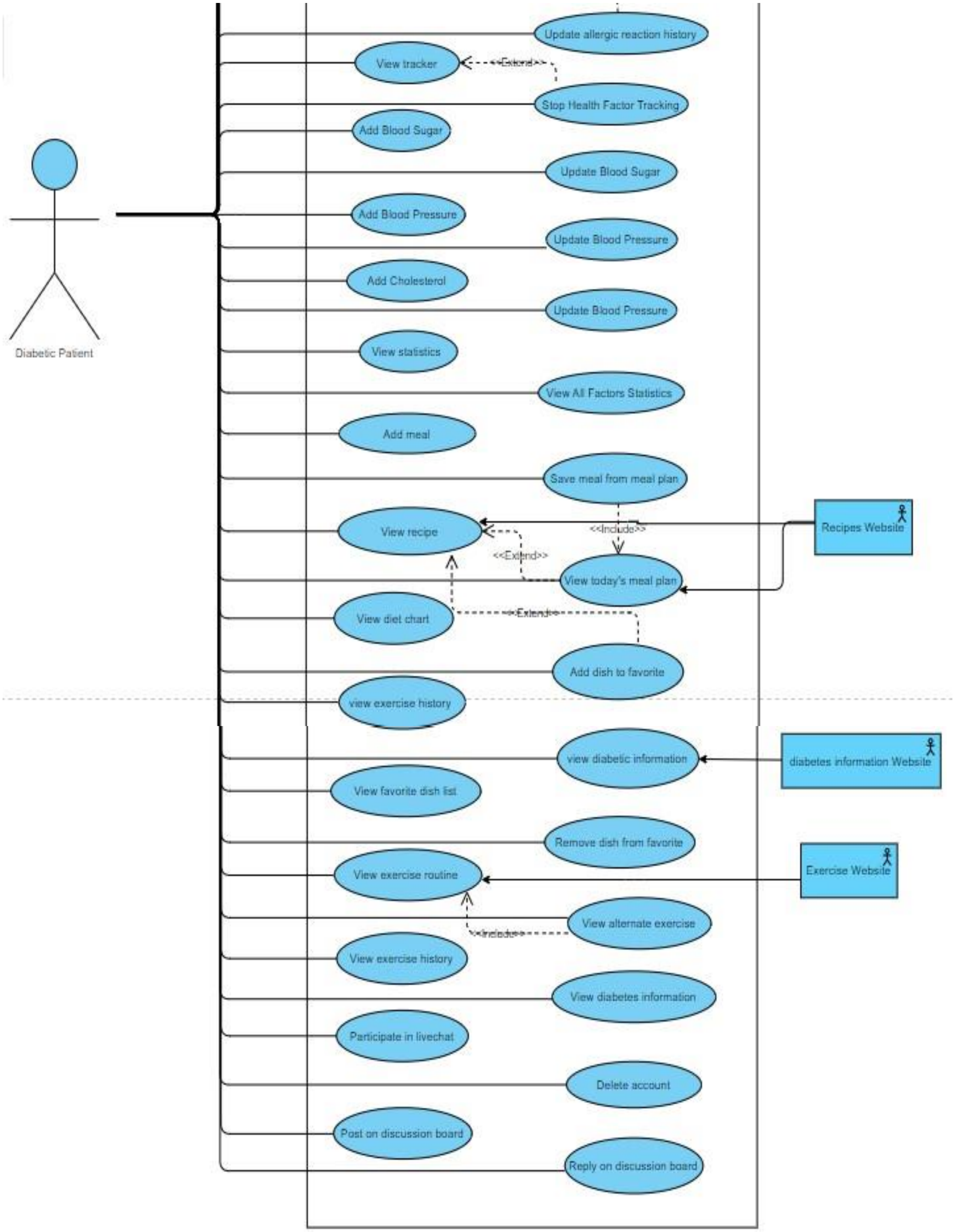
- Use case (use case diagram and detailed use case)
- Event response tables

2.2.1 Use case diagram:

The figure 2-1 shows use case diagram for Smart Diabetolog application.

Figure 2-1: Use case diagram


Smart Diabetalog
Register
Login
Edit Profile
View BMI
Predict diabetes
Add new oral medication
Add insulin medication
View current oral medication
View current insulin medication
Medication Website
update oral medication
update insulin medication
View current medication prescription
View prescription history
change prescription title
Add allergic reaction history
View allergic reaction history
View allergic reaction instance
<<Include>>


Diabetic Patient
Update allergic reaction history
View tracker
<<Extend>>
Stop Health Factor Tracking
Add Blood Sugar
Update Blood Sugar
Add Blood Pressure
Update Blood Pressure
Add Cholesterol
Update Blood Pressure
View statistics
View All Factors Statistics
Add meal
Save meal from meal plan
View recipe
<<Include>>
Recipes Website
<<Extend>>
View today's meal plan
View diet chart
<<Extend>>
Add dish to favorite
view exercise history
view diabetic information
diabetes information Website
View favorite dish list
Remove dish from favorite
View exercise routine
Exercise Website
View alternate exercise
<<Include>>
View exercise history
View diabetes information
Participate in livechat
Delete account
Post on discussion board
Reply on discussion board

2.2.2 Business Rules

Table 2-2 defines the business rules according to our project.

Table 2-2: Business Rules

ID	Rule Definition	Type of Rule	Static or Dynamic	Source
BR-1	The history of only a year would be available for the user.	Fact	Static	Fatima
BR-2	The user must register for getting an access of the system	Constraint	Static	Kashaf
BR-3	If the user has diabetes type1 then they would have no connection with diabetic oral medications	Inference	Static	Fatima
BR-4	The system must use basal-bolus regimen for insulin management.	Constraint	Dynamic	Fatima
BR-5	If the user initiates to view cholesterol, then the system must show the cholesterol instances of 6 months to the user.	Action Enabler	Static	Fatima

BR-6	If the user initiates to view blood sugar then the system must show the today’s blood sugar instances to the user.	Action Enabler	Static	Fatima
BR-7	If the user initiates to view blood pressure then the system must show the blood pressure instances of a week to the user.	Action Enabler	Static	Fatima
BR-8	The user must add only 2 current insulin medication.	Constraint	Static	Fatima
BR-9	One email can be registered only once into the system to make each patient unique.	Constraint	Static	Kashaf

2.2.3 Detailed use cases

2.2.3.1 Register

Table 2-3: UC-1 Register

Use Case ID:	UC-1
Use Case Name:	Register
Actors:	Primary Actors: diabetic patient
Description:	The diabetic will be registered into the application by providing information.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate to register into the application
Preconditions:	Pre-1: The user should have an email account. Pre-2: The user is not already registered into the system.
Postconditions:	Post-1: The diabetic patient’s information is stored into the application database. Post-2: The diabetic patients would be provided access to the application.
Normal Flow:	1. Application displays a registration form to fill. 2. Diabetic patient enters his full name. 3. Diabetic patient enters his email. (E1)
	4. Patient enters his gender. 5. Patient enters his weight. 6. Patient enters his height. 7. Diabetic patient enters a secured password for his account. (E2) 8. Re-enter password to confirm. (E3) 9. Diabetic patient indicates on registration form filled. 10. Application sends code on patient email to verify. 11. Patient receives code. 12. Application prompts the patient to enter code. 13. Application matches the code and account is created. (E4)
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	

Exceptions:	**E1) The Email is already registered** 1. Application shows a message that the email is already registered. 2. Application prompt the user to enter another email 3. Return to step 4 of normal flow **E2) The password entered is less than 6 alphabets.** 1. Application shows a message that password is too weak. 2. Application prompt the user to enter another password with at least then 6 alphabets and digits. 3. Return to step 5 of normal flow. **E3) Both passwords do not match.** 1. Application shows a message that passwords do not match. 2. Application prompts the user to re-enter passwords. 3. Return to step 4 of normal flow. **E4) If the code entered does not match with the code sent.** 1. Application returns to normal flow step
Includes:	
Business Rules:	BR-9
Special Requirements/ Other Information:	1. Application is accessible 24/7. Any user can register at any time in a day. 2. The patient would be getting the confirmation of the registered account within 1 minute of registration. 3. The user would be notified of the exception conditions within nanoseconds of its existence.

2.2.3.2 Login

Table 2-4: UC-2 Login

Use Case ID:	UC-2
Use Case Name:	Login
Actors:	Primary Actors: diabetic patient
Description:	The diabetic patient will be able to access application functionality by logging in to the application.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate to log in to the application.
Preconditions:	Pre-1: The patient is already registered into the application.
Postconditions:	Post-1: Application main page is visible to the patient. .
Normal Flow:	1. Patient enters his email. (E1) 2. Patient enters his password. (E2) 3. Patients indicates on log in completed. (E3)
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	
Exceptions:	**E1) The Username doesn't exist:** 1. The patient is notified that the email might be wrong because no such patient exists in the system. 2. The patient is asked to re-enter the email. 3. Returns to step 1 of normal flow. **E2) The password is wrong** 1. The patient is notified that he entered wrong password. 2. The patient is asked to re-enter the password. 3. Returns to step 2 of normal flow. **E3) The Internet Connection is lost** 1. As soon as the Internet is accessed the user is directed to step 1 of normal flow
Includes:	UC-1
Business Rules:	BR-2
Special Requirements/ Other Information:	1. The patient is directed into its account within 20 seconds of its request to login

2.2.3.3 Add new oral medication

Table 2-5: UC-3 Add new oral medication

Use Case ID:	UC-3
Use Case Name:	Add new oral medication
Actors:	User
Description:	The user could add all the oral medications he is currently taking to keep its record.
Trigger:	The use case would be triggered as soon as the user initiates to add a new oral medication.
Preconditions:	Pre-1: User is registered into the system.
Postconditions:	
Normal Flow:	1. User initiates to view medication. 2. System prompts the user to select prescription. 3. System prompts the user to select the type of medication from oral medication and insulin. 4. System prompts the user to select whether the oral medication is diabetic or not 5. System then prompts the user to make a choice whether to add medication name manually or by picture. 6. User writes the medication name manually. 7. User enters the total mg(milligram) unit of the medication. 8. User than selects the time of the day medication would be used 9. User than enters the prescribed dosage quantity of the medication. 10. User than initiates to save the entered medication record.

Alternative Flows: [Alternative Flow 1 – Not in Network]	**2.1 If the user wants to make a new prescription** 1. The user initiates to create a new prescription. 2. User then enters the name for the prescription. 3. Then return to step 3 of normal flow. **6.1 If the user wants to take a picture of the medication name.** 1. The user takes the picture of the medication name. 2. System than displays the name of the medicine in medication name. 3. Then return to step 7 of normal flow. **8. If the user wants to take a medication multiple time.** 1. The user initiates to add another time instance. 2. User then selects the time of the day. 3. Then return to step 9 of normal flow.
Exceptions:	
Business Rules:	BR-2, BR-3
Assumptions:	

2.2.3.4 Add insulin medication

Table 2-6: UC-4 Add insulin Medication

Use Case ID:	UC-4
Use Case Name:	Add insulin medication
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could add all the information related to its parenteral insulin medication.
Trigger:	The use case would be triggered as soon as the user initiates to add an insulin medication.
Preconditions:	Pre-1: User is registered into the system.
Postconditions:	

Normal Flow:	1. User initiates to view medication. 2. System prompts the user to select prescription. 3. System prompts the user to select the type of medication from oral medication and insulin. 4. User selects Insulin 5. User initiates to add new insulin medication. 6. System prompts the user to select a type of insulin (long acting or fast acting insulin). 7. User selects a fast-acting insulin. 8. System prompts the user to select the name of the insulin. 9. System prompts the user to enter the current insulin sensitivity factor ISF (sensitivity of person to insulin. For example: 1 unit of insulin lowers the glucose level by 50) 10. System then prompts the user to enter current carb ratio (1 unit of insulin covers how many grams of carbohydrates) 11. User than initiates to save the entered insulin medication record.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**2.1 If the user wants to make a new prescription** 1. The user initiates to create a new prescription. 2. User then enters the name for the prescription. 3. Then return to step 3 of normal flow. **5.1 If the user wants to add second Insulin** 1. User Initiates to add second insulin 2. Then return to step 6 of normal flow **6.1 If the user selects the long-acting insulin** 1. User selects a long-acting insulin 2. System prompts the user to select the name of the insulin. 3. System then prompts the user to enter the units to be used of the insulin. 4. System then prompts the user to enter the time of the insulin would be used. (see E1) 5. Then return to step 11 of normal flow.
Exceptions:	E1) If the time for the long-acting insulin is not mentioned then it is assumed to be 10:00 PM which is the time to bed at night.
Business Rules:	BR-2, BR-4, BR-8
Assumptions:	

2.2.3.5 View current oral medication

Table 2-7: UC-5 View current oral medication

Use Case ID:	UC-5
Use Case Name:	View current oral medication
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could view all the oral medications he is currently taking.
Trigger:	The use case would be triggered as soon as the user initiates to view the current oral medication.
Preconditions:	Pre-1: User is registered into the system. Pre-2: The current medication is already added to the system.
Postconditions:	Post-1: If current medication record is maintained user is provided its display. Post-2: If no record is previously added for medication, a message showing "No records are available" is shown to the user.
Normal Flow:	1. User initiates to view current oral medication. 2. Systems displays all the current oral medication along with their details to the user.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	BR-2
Assumptions:	

2.2.3.6 Update oral medication

Table 2-8: UC-6 Update oral medication

Use Case ID:	UC-6
Use Case Name:	Update oral medication
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could update or modify any of the medication details.
Trigger:	The use case would be triggered as soon as the user initiates to update the oral medication.
Preconditions:	Pre-1: User is registered into the system.
Postconditions:	Post-1: System displays a message to the user showing "The changes are successfully done"

Normal Flow:	1. User initiates to View the medication. 2. System displays the medication record to the user. 3. User then selects a particular medication. 4. System displays the stored information about the selected medication instance. 5. User than makes its desired changes. 6. User then initiates to save the changes.
Alternative Flows: [Alternative Flow 1 – Not in Network]	**5.1 If the user wants to delete the selected medication instance.** 1. The user initiates to delete the instance. 2. System prompts the user to confirm the action.
	3. User then confirms the action. 4. Then return to step 2 of normal flow. **5.2 If the user wants to cancel the modification selected medication instance.** 1. The user initiates to cancel the update action. 2. System prompts the user to confirm the cancellation action. 3. User then confirms the action. 4. Then return to step 2 of normal flow.
Exceptions:	
Business Rules:	BR-2, BR-3
Special Requirements/ Other Information:	

2.2.3.7 View current insulin medication

Table 2-9: UC-7 View current insulin medication

Use Case ID:	UC-7
Use Case Name:	View current insulin medication
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could view the details related to the insulin he is currently taking.
Trigger:	The use case would be triggered as soon as the user initiates to view the current insulin medication.
Preconditions:	Pre-1: User is registered into the system. Pre-2: The insulin medication record is already added to the system.

Postconditions:	Post-1: If insulin medication record is maintained user is provided with its display. Post-2: If no record is previously added for insulin, a message showing "No record is available" is shown to the user.
Normal Flow:	1. User initiates to view current insulin medication. 2. Systems displays the current insulin along with its details to the user.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	BR-2, BR-4, BR-8
Special Requirements/ Other Information:	

2.2.3.8 Update insulin medication

Table 2-10: UC-8 Update insulin medication

Use Case ID:	UC-8
Use Case Name:	Update insulin medication
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could update or modify any of the insulin medication record.
Trigger:	The use case would be triggered as soon as the user initiates to update the insulin medication.
Preconditions:	Pre-1: User is registered into the system.
Postconditions:	Post-1: System displays a message to the user showing "The changes are successfully done"
Normal Flow:	1. User initiates to View the insulin medication. 2. System displays insulin record to the user. 3. User selects the insulin instance. 4. System displays all the stored information about the insulin medication. 5. User than makes its desired changes. 6. User then initiates to save the changes.

Alternative Flows: [Alternative Flow 1 – Not in Network]	**If the user wants to delete the insulin medication instance.** 5a. The user initiates to delete the instance. 6.System prompts the user to confirm the action. 7.User then confirms the action. **If the user wants to cancel the modification.** 5a. The user initiates to cancel the update action. 6.System prompts the user to confirm the cancellation action. 7.User then confirms the action. 8.Then return to step 2 of normal flow.
Exceptions:	
Includes:	UC-8
Business Rules:	BR-2, BR-4, BR-8
Special Requirements/ Other Information:	

2.2.3.9 View Current Medication Prescription

Table 2-11: UC-9 View Current Medication Prescription

Use Case ID:	UC-9
Use Case Name:	View Current Medication Prescription
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could view current medication prescription in detail.
Trigger:	The use case would be triggered as soon as the user initiates to view the current medication prescription.
Preconditions:	
Postconditions:	
Normal Flow:	1. User initiates the current medication prescription. 2. System displays the current prescription to the user.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	BR-2
Special Requirements/ Other Information:	

2.2.3.10 View Prescription History

Table 2-12: UC-10 View Prescription History

Use Case ID:	UC-10
Use Case Name:	View Prescription History
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could view any of its medication prescription history in the past.
Trigger:	The use case would be triggered as soon as the user initiates to view the medication history.
Preconditions:	Pre-1: At least one medication prescription is stored.
Postconditions:	
Normal Flow:	1. User initiates to View the medication prescription history. 2. System displays all the stored medication prescriptions by title. 3. User selects a particular instance of the prescriptions. 4. The system then displays all the stored medications in the selected prescription to the user.
Alternative Flows: [Alternative Flow 1 – Not in Network]	**2.1 If the user wants to view the medication prescription by month.** 1. The user initiates to view the medication prescription by month. 2. System displays all the months along with the title of the prescription used in that month. 3. Then return to step 3 of normal flow.
Exceptions:	
Business Rules:	BR-1, BR-2
Special Requirements/ Other Information:	
Use Case ID:	UC-12
Use Case Name:	Change Prescription Title
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could change the title of any particular prescription.
Trigger:	The use case would be triggered as soon as the user initiates to change the prescription title.
Preconditions:	Pre-1: At least one medication prescription is stored.
Postconditions:	

Normal Flow:	1. User selects a particular prescription. 2. System displays the prescription details to the user. 3. User initiates to change the prescription title. 4. User changes the title. 5. User initiates to save the changes.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	BR-1, BR-2
Assumptions:	

2.2.3.11 Add Allergic Reaction History

Table 2-13: UC-11 Add Allergic Reaction History

Use Case ID:	UC-11
Use Case Name:	Add Allergic Reaction History
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to add all its allergic reaction history related to food and medicines both.
Trigger:	The use case would be triggered as soon as the user initiates to add allergic reaction history.
Preconditions:	Pre-1: The user is registered into the system
Postconditions:	
Normal Flow:	1. User initiates to create allergic reaction history 2. Then the system prompts the user to select the allergy category (diabetic Medication, Non-diabetic medication, Insulin or food) 3. Then the system prompts the user to select or write the medication or food name he had allergy in the past. 4. The System than prompts the user to select all the symptoms experienced as a reaction. 5. The system then allows the user to add a small description. 6. Then the user initiates to save the allergic reaction information.

Alternative Flows: [Alternative Flow 1 – Not in Network]	**5.1 If the user had an allergy with multiple medications or food in the past.** 1. Then step 3 to 5 would be repeated for all the allergic reaction instances 2. Then return to step 6 of normal flow.
Exceptions:	
Business Rules:	BR-2
Assumptions:	

2.2.3.12 View Allergic Reaction History

Table 2-14: UC-12 View Allergic Reaction History

Use Case ID:	UC-12
Use Case Name:	View Allergic Reaction History
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to view its allergic reaction history related to both food and medicines.
Trigger:	The use case would be triggered as soon as the user initiates to view its Allergic Reaction History.
Preconditions:	1. User is registered into the system. 2. User has already created it allergic reaction history.
Postconditions:	1. All the stored allergic reaction record's is displayed to the user.
Normal Flow:	1. The user initiates to view the Allergic Reaction records. 2. System prompts the user to select the category from diabetic medication, non-diabetic medication, insulin or food 3. System then displays the allergic reaction related to the category.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Includes:	
Business Rules:	BR-2
Assumptions:	

2.2.3.13 View Allergic Reaction

Table 2-15: UC-13 View Allergic Reaction

Use Case ID:	UC-13
Use Case Name:	View Allergic Reaction
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to view the allergic reaction instance.
Trigger:	The use case would be triggered as soon as the user initiates to view the Allergic Reaction.
Preconditions:	Pre-1: User is registered into the system. Pre-2: User has already created it allergic reaction history.
Postconditions:	
Normal Flow:	1. The user initiates to view the Allergic Reaction. 2. System prompts the user to select the category from diabetic medication, non-diabetic medication, insulin or food 3. System then displays the allergic reaction instances related to the category. 4. User then selects an instance of the allergic reactions 5. System then displays the selected instance information to the user.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Business Rules:	BR-2
Assumptions:	

2.2.3.14 Update Allergic Reaction History

Table 2-16: UC-14 Update Allergic Reaction History

Use Case ID:	UC-14
Use Case Name:	Update Allergic Reaction History
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to update its allergic reaction history related to both food and medicines.
Trigger:	The use case would be triggered as soon as the user initiates to edit its Allergic Reaction History.
Preconditions:	Pre-1: User is registered into the system. Pre-2: User has already created it allergic reaction history.

Postconditions:	Post-1: A message displaying “changes are successfully made” is shown to the user
Normal Flow:	1. User initiates to view the allergic reaction history. 2. System displays the Allergic reaction record. 3. User selects the allergic reaction instance. 4. System displays the allergic reaction information to the user. 5. User updates the allergic reaction information
	6. User then initiates to save the updated version of the allergic reaction.
Alternative Flows: [Alternative Flow 1 – Not in Network]	**7.1 If the user wants to delete the selected allergic reaction instance.** 1. The user initiates to delete the instance. 2. System displays a message showing “Are you sure?” 3. User initiates yes **7.2 If the user initiates to cancel the updating process.** 1. The user initiates to cancel the updating process 2. Then returns to step 2 of normal flow
Exceptions:	
Business Rules:	BR-2
Assumptions:	

2.2.3.15 View Tracker

Table 2-17: UC-15 View Tracker

Use Case ID:	UC-15
Use Case Name:	View Tracker
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be to track factors related to its health.
Trigger:	The use case would be triggered as soon as the user initiates to view the tracker.
Preconditions:	
Postconditions:	
Normal Flow:	1. User initiates to view a tracker. 2. System prompts the user to select the category from Blood Sugar, Blood Pressure or Cholesterol 3. User selects the category. 4. System displays the record related to the category. (see E1)
Alternative Flows:	

[Alternative Flow 1 – Not in Network]	
Exceptions:	E1) If the record is not present the system displays a message showing no record is present for the category.
Business Rules:	BR-2, BR-5, BR-6, BR-7
Assumptions:	

2.2.3.16 Stop Health Factor Tracking

Table 2-18: UC-16 Stop Health Factor Tracking

Use Case ID:	UC-16
Use Case Name:	Stop Health Factor Tracking
Created By:	Fatima Shahzad
Actors:	User
Description:	The user could stop the tracking of any health factor except blood-sugar.
Trigger:	The use case would be triggered as soon as the user initiates to stop heath factor tracking.
Preconditions:	
Postconditions:	
Normal Flow:	1. User initiates to view the tracker 2. User initiates to view the settings
	3. System displays the list of tracking health factors 4. User stops the tracking of any factor from the list.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	
Exceptions:	
Business Rules:	BR-1
Assumptions:	By default all the health factors are available for tracking.

2.2.3.17 Add Blood Sugar

Table 2-19: UC-17 Add Blood Sugar

Use Case ID:	UC-17
Use Case Name:	Add Blood Sugar
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to add its sugar level at any time of the day

Trigger:	The use case would be triggered as soon as the user initiates to add its blood sugar.
Preconditions:	
Postconditions:	Post-1: A message showing “successfully stored” is displayed.
Normal Flow:	1. User initiates to view a tracker and then selects blood sugar. 2. System displays the blood sugar record for today’s date. 3. User initiates to add an instance of blood sugar. 4. System then prompts the user to enter their blood sugar concentration. 5. System then prompts the user to select a unit for blood sugar concentration. 6. System prompts the user to select an event when blood sugar was measured. 7. System prompts the user to select the time of the day the blood sugar war measured. 8. System then prompts the user to write a note. 9. User than initiates to save the blood sugar count instance.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	E1: If the blood sugar is low (hypoglycemia) than target blood sugar range than a measure to increase blood sugar (like taking a food with high sugar concentration but without fat or proteins) is suggested. E2: If the blood sugar is high (hyperglycemia) than target blood sugar range than a measure to lower blood sugar (like may be taking an insulin or an exercise) is suggested.
Business Rules:	BR-2, BR-6
Assumptions:	

2.2.3.18 Update Blood Sugar

Table 2-20: UC-18 Update Blood Sugar

Use Case ID:	UC-18
Use Case Name:	Update Blood Sugar
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to update its blood sugar instance.
Trigger:	The use case would be triggered as soon as the user initiates to update its blood sugar instance.
Preconditions:	

Postconditions:	Post-1: System displays a message to the user showing “The changes are successfully done”
Normal Flow:	1. User initiates to view a tracker and then selects blood sugar. 2. System displays the blood sugar record for today’s date. 1. User selects an instance of blood sugar. 2. System the displays the selected instance information. 3. User than makes the desired changes. 4. User than initiates to save the changes.
Alternative Flows: [Alternative Flow 1 – Not in Network]	**6.1 If the user wants to delete the selected blood sugar instance.** 1. The user initiates to delete the instance. 2. System prompts the user to confirm the action. 3. User then confirms the action. **6.2 If the user wants to cancel the modification.** 1. The user initiates to cancel the update action. 2. System prompts the user to confirm the cancellation action. 3. User then confirms the action. 4. 8. Then return to step 2 of normal flow.
Exceptions:	
Business Rules:	BR-2, BR-6
Assumptions:	

2.2.3.19 Add Blood Pressure

Table 2-21: UC-19 Add Blood Pressure

Use Case ID:	UC-19
Use Case Name:	Add Blood Pressure
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to add its blood pressure readings into the system.
Trigger:	The use case would be triggered as soon as the user initiates to add its blood pressure.
Preconditions:	
Postconditions:	Post-1: A message showing an action done successfully is displayed.

Normal Flow:	1. User initiates to view a tracker and then selects blood pressure. 2. System displays the Blood Pressure record for today's date. 3. User initiates to add an instance of blood pressure. 4. System then prompts the user to enter systolic blood pressure. 5. System then prompts the user to enter diastolic blood pressure. 6. System prompts the user to select the time of the day the blood pressure was measured. 7. System then prompts the user to write a note. 8. User than initiates to save the blood pressure count instance.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	E1: If the high pressure continuously for few days remains above high blood pressure stage 1(130-139/ 80-89) and stage 2(140-180/ 90-120) than a warning about hypertension crisis is displayed to the user.
Business Rules:	BR-2, BR-7
Assumptions:	

2.2.3.20 Update Blood Pressure

Table 2-22: UC-20 Update Blood Pressure

Use Case ID:	UC-20
Use Case Name:	Update Blood Pressure
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to update its blood pressure instance.
Trigger:	The use case would be triggered as soon as the user initiates to update its blood pressure instance.
Preconditions:	
Postconditions:	Post-1: System displays a message to the user showing "The changes are successfully done"
Normal Flow:	1. User initiates to view a tracker and then selects blood pressure. 2. System displays the blood pressure record for a week to the user. 3. User selects an instance of blood pressure. 4. System the displays the selected instance information. 5. User than makes the desired changes. 6. User than initiates to save the changes.

Alternative Flows: [Alternative Flow 1 – Not in Network]	**6.1 f the user wants to delete the selected blood pressure instance.** 1. The user initiates to delete the instance. 2. System prompts the user to confirm the action. 3. User then confirms the action. **6.2 If the user wants to cancel the modification.** 1. The user initiates to cancel the update action. 2. System prompts the user to confirm the cancellation action. 3. User then confirms the action. 4. 8. Then return to step 2 of normal flow.
Exceptions:	
Business Rules:	BR-2, BR-7
Assumptions:	

2.2.3.21 Add Cholesterol

Table 2-23: UC 21 Add Cholesterol

Use Case ID:	UC-21
Use Case Name:	Add Cholesterol
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to add its cholesterol reading for the record.
Trigger:	The use case would be triggered as soon as the user initiates to add cholesterol.
Preconditions:	
Postconditions:	Post-1: A message showing an action done successfully is displayed.
Normal Flow:	1. User initiates to view a tracker and then selects cholesterol. 2. System displays the cholesterol record for 6 months. 3. User initiates to add an instance of cholesterol. 4. System then prompts the user to enter the cholesterol value. 5. System then prompts the user to enter the LDL value. 6. System then prompts the user to enter the HDL value. 7. System then prompts the user to enter the Triglyceride value. 8. User than initiates to save the cholesterol instance.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	BR-2, BR-5
Assumptions:	

2.2.3.22 Update Cholesterol

Table 2-24: UC-22 Update Cholesterol

Use Case ID:	UC-22
Use Case Name:	Update Cholesterol
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to update its cholesterol instance.
Trigger:	The use case would be triggered as soon as the user initiates to update cholesterol instance.
Preconditions:	
Postconditions:	Post-1 System displays a message to the user showing "The changes are successfully done"
Normal Flow:	1. User initiates to view a tracker and then selects cholesterol. 2. System displays the cholesterol record for 6 months. 3. User selects an instance of cholesterol. 4. System the displays the selected instance information. 5. User than makes the desired changes. 6. User than initiates to save the changes.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**If the user wants to delete the selected cholesterol instance.** 1. The user initiates to delete the instance. 2. 6.System prompts the user to confirm the action. 3. User then confirms the action. **If the user wants to cancel the modification.** 1. The user initiates to cancel the update action. 2. System prompts the user to confirm the cancellation action. 3. User then confirms the action. 8.Then return to step 2 of normal flow.
Exceptions:	
Business Rules:	BR-2, BR-5
Assumptions:	

2.2.3.23 View Statistics

Table 2-25: UC-23 View Statistics

Use Case ID:	UC-23
Use Case Name:	View Statistics
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to view the statistics for any factor related to its health.
Trigger:	The use case would be triggered as soon as the user initiates to view statistics.
Preconditions:	
Postconditions:	
Normal Flow:	1. User initiates to view a tracker. 2. System displays the record for the selected category. 3. User selects to view statistics. 4. User selects to view the statistics of the day. 5. System displays the statistics of the day.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**4.1 If the user wants to view the statistics of the week.** 1. User selects to view the statistics of the week. 2. System displays the statistics of the week. **4.2 If the user wants to view the statistics of the month.** 1. User selects to view the statistics of the month. 2. System displays the statistics of the month. **4.3 If the user wants to view the statistics of the 6 months.** 1. User selects to view the statistics of 6 months. 2. System displays the statistics for the 6 months. **4.4 If the user wants to view the statistics of the year.** 1. User selects to view the statistics of the year. 2. System displays the statistics of the year.
Exceptions:	
Business Rules:	BR-1, BR-2, BR-3
Assumptions:	

2.2.3.24 View all factor statistics

Table 2-26: UC-24 View all factor statistics

Use Case ID:	UC-24
Use Case Name:	View all factor statistics
Created By:	Fatima Shahzad
Actors:	User
Description:	The user would be able to view the statistics for all available health factors combined.
Trigger:	The use case would be triggered as soon as the user initiates to view statistics for all health factors.
Preconditions:	
Postconditions:	
Normal Flow:	1. User initiates to view a tracker. 2. User initiates to view the combined health factors statistics.
	3. System displays the statistics.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	BR-1, BR-2
Assumptions:	The all factor statistics would be shown month wise.

2.2.3.25 Add meal

Table 2-27: UC-25 Add meal

Use Case ID:	UC-25
Use Case Name:	Add meal
Actors:	Primary Actors: diabetic patient
Description:	Patients sometimes may not consume meal suggested by the application. That meal will be stored in the application by patient.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on adding meal.
Preconditions:	Pre-1: The patient is already logged into the application.

Postconditions:	Post-1: The meal consumption time along with its calories count will be saved in patients information. Post 2: The patient meal plan and exercise plan will be updated according to his consumed calories. .
Normal Flow:	1. Patient selects add meal. 2. Patient enters total number of calories he consumed. (1.1) 3. Patient enters the meal type i.e. (breakfast, lunch, dinner, snack). 4. Patient enters the dish names he consumed. (E2) 5. Patient enters time when meal is consumed. (1.2, E1) 6. Patients enter the number of carbohydrates, fiber, and proteins in their meal. 7. Patient indicates on save the information.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	1. **Patients add calories through meals images.** **1.** Patients select add images. **2.** Application opens the camera for patients to take images. **3.** Patient takes one image from top of meal. **4.** Patient takes another image from side of the meal. **5.** Application suggests the meal name. 6a. If meal name is not correctly suggested, the application returns ....to normal flow step 2. 6b. else the application predicts the number of calories present in ....the meal. 7. Application predicts the number of carbohydrates, fibers, and ....proteins.
	8a. If the prediction is right the application returns to normal flow .... step 7. 8b. else application prompts the user to add this information ....manually. 9. application is returned to normal flow step 7. 2. **Patient adds time using a clock.** 1. Patient selects clock. 2. Patient moves the needles of clock according to specific time. 3. Returns to step 6 of normal flow.

Exceptions:	**E1) The selected time ahead of current time.** 1. The patient is notified about the time specified is after the current time. 2a. If the patient selects ok, then the application returns to step 4 of normal flow. 2b. else the application is returned to step 3 of the normal flow. **E2) Patient wants to add more than one dish he consumed.** 1. Patient selects add dish. 2. Application shows another tab for dish name to enter. 3. Patient enters the dish name. 4a. If the patient still wants to enter more the application returns to .E2 Step 1. 4b. else the application is returned to normal flow step 5.
Business Rules:	
Assumptions:	

2.2.3.26 View diet chart

Table 2-28: UC-26 View diet chart

Use Case ID:	UC-26
Use Case Name:	View diet chart
Actors:	Primary Actors: diabetic patient
Description:	The diet information about the meals he had consumed will be stored in the diet chart which patient will be able to view.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on displaying diet chart.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: Diet chart will be displayed on screen. .
Normal Flow:	1. Patient indicates on displaying diet chart. 2. Application displays diet chart on screen. (E1)
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	
Exceptions:	**E1: Patient has not stored any meal information yet.** 1. Application will display an empty diet chart then.

Business Rules:	
Assumptions:	1.

2.2.3.27 View today's meal plan

Table 2-29: UC-27 View today's meal plan

Use Case ID:	UC-27
Use Case Name:	View today's meal plan
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to view his current day's meal plan. This meal plan will contain the number of meals he wants throughout the day.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on view current day's meal plan.
Preconditions:	Pre-1: The patient is logged into the application. Pre-2: Patient has provided his information to make his meal plan.
Postconditions:	Post-1: The numbers of meals requested by the patient according to his ideal calorie intake will be displayed. .
Normal Flow:	1. Patient selects meal plan 2. Application will display meal plan for entire day meals. (E1) 3. The Patients select breakfast from meal type. (1.1, 1.2, 1.3) 4. The application will display all the dishes included in breakfast.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**1.1. If the patient selects lunch from meal type.** **1.** The application will display all the dishes included in lunch. **1.2. If the patient selects snack from meal type.** 1. The application will display all the dishes included in lunch. **1.3. If the patient selects dinner from meal type.** 1. The application will display all the dishes included in lunch.

Exceptions:	**E1: patient has not given his information to create his meal plan.** 1. Application will prompt a message to enter related details. 2. Patient will enter his details. 3. Returns to step 2 of normal flow.
Business Rules:	
Assumptions:	

2.2.3.28 View recipe

Table 2-30: UC-28 View recipe

Use Case ID:	UC-28
Use Case Name:	View recipe
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to view recipes of dishes added in their meal plan.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on viewing recipes.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: The recipe of the selected dish will appear on the screen. .
Normal Flow:	1. Patients select the dish. 2. Patient indicates on view recipe. 3. Application will display the recipe along will all the ingredients with measurements.
Alternative Flows:	
[Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	
Assumptions:	

2.2.3.29 *Save meal from meal plan*

Table 2-31: UC -29 Save meal from meal plan

Use Case ID:	UC-29
Use Case Name:	Save meal from meal plan
Actors:	Primary Actors: diabetic patient
Description:	The patient will save the meal from the meal plan indicating he has consumed the meal.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate saving the meal.
Preconditions:	Pre-1: The patient is logged into the application. Pre-2: Patients meal plan is already set up.
Postconditions:	Post-1: The diet chart will be updated with new consumed meal. Post-2: The consumed calories count will be updated. Post-3: The meal plan will be updated. .
Normal Flow:	1. Patient selects meal plan. 2. Application will display today's meal plan. 3. The patient then selects the meal he consumed from the meal plan. 4. Then patient indicates on save. 5. Application will prompt the patient to enter consumption time 6. Patient enters the time. 7. Patient indicates on save.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	
Exceptions:	
Business Rules:	
Assumptions:	

2.2.3.30 View BMI

Table 2-32: UC-30 View BMI

Use Case ID:	UC-30
Use Case Name:	View BMI
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to check his body mass index calculated according to his given weight and height.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on displaying his BMI.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: Calculated BMI will appear on screen. .
Normal Flow:	1. Patient selects BMI. (1.1) 2. Application prompts the patient to enter his height and weight. (E1) 3. Patient enters his height. 4. Patient enters his weight. 5. Application calculates BMI. 6. Patient's BMI is displayed by application.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**1.1) if the patient has already provided his height and weight.** 1. Application will return to step 5 of normal flow.
Exceptions:	
Business Rules:	
Assumptions:	

2.2.3.31 View exercise routine

Table 2-33: UC-31 View exercise routine

Use Case ID:	UC-31
Use Case Name:	View exercise routine
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to view the exercises routine according to his provided details assigned to him and start, stop, resume and pause it.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on start exercise routine.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: The exercise routine for the respected day will start. .
Normal Flow:	1. Patients indicate on exercise plan. (E1) 2. Application displays today's exercise routine 3. Patient select start today's exercise plan. (1.1, 1.2) 4. Application starts the exercise.
Alternative Flows: [Alternative Flow 1 – Not in Network]	**1.1. if patient selects pause exercise** 1. Application will pause exercise plan. 2. Patient selects resume. (E2) 3. The exercise will resume from where it stopped. **1.2. if the patient selects Stop exercise.** 1. the use case will be terminated
Exceptions:	**E1) Patient has not given his medical details yet.**
	1. Application prompts a message to enter related details. 2. Patient will enter his details. 3. Return to normal flow step 2. **E2) Patient do not resume exercise in the day.** 1. The exercise will be marked haft attempted in the diet chart. 2. The diet chart will be updated according to exercise attempted.
Business Rules:	
Assumptions:	

2.2.3.32 View exercise history

Table 2-34: UC-32 View exercise history

Use Case ID:	UC-32
Use Case Name:	View exercise history
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to view the exercise history of when he has achieved his goals and when not.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on displaying exercise history.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: The exercise history will be displayed. .
Normal Flow:	1. Patient asks to view exercise history. 2. The exercise history of the month will be displayed. (E1)
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	
Exceptions:	**E1) If the patient has not yet attempted any exercise.** 1. Empty exercise charts will be displayed.
Business Rules:	
Assumptions:	

2.2.3.33 View diabetes information

Table 2-35: UC-33 View diabetes information

Use Case ID:	UC-33
Use Case Name:	View diabetes information
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to view all the diabetes related information uploaded by authentic sources.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on viewing diabetes related information.
Preconditions:	Pre-1: The patient is logged into the application.

Postconditions:	Post-1: diabetes related content selected by the patient will be displayed. .
Normal Flow:	1. Patient indicates on viewing diabetes related information. 2. Application will display different diabetes related content. 3. Patients select blogs. (1.1, 1.2, 1.3) 4. Application displays different blogs from authentic websites. 5. A patient selects a blog of his choice. 6. Application opens the blog on the browser of patient's device.
Alternative Flows: [Alternative Flow 1 – Not in Network]	**1.1. Patient selects videos.** **1.**Application displays video content related to diabetes from authentic websites. **2.**Patient selects the video of his choice. **3.**Application opens the video on the browser of patient's device. **1.2. Patient selects Articles.** 1. Application display Articles related to diabetes from authentic websites. 2. Patient selects the article of his choice. 3. Application opens the article on the browser of patient's device.
	1.3. Patient selects videos. 1.3.1. Application displays video content related to diabetes from authentic websites. 1.3.2. Patient selects the video of his choice. 1.3.3. Application opens the video on the browser of patient's device.
Exceptions:	
Business Rules:	
Assumptions:	

2.2.3.34 *Participate in live chats*

Table 2-36: UC-34 Participate in live chats

Use Case ID:	UC-34
Use Case Name:	Participate in live chats
Actors:	Primary Actors: diabetic patient
Description:	The patient will be able to interact with other patients through a common live chat.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on live chatting with other patients.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: Patients message will be sent in live chat to be viewed by other patients.
Normal Flow:	1. Patient indicates on live chat. 2. Patient enters the message he wants to send. (E1, E2) 3. Patient indicates on sending the message.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	**E1: patient want to attach an image with the message.** 1. Patient selects add image. 2. Application displays patient already saved images in device. 3. Patient selects image. 4. Return step 3 of normal flow. **E2: patient want to attach more than one image with the message.** 1. Patient selects add image. 2. Application displays patient already saved images in device. 3. Patients select more than one image. 4. Return step 3 of normal flow.
Business Rules:	
Assumptions:	

2.2.3.35 Delete Account

Table 2-37: UC-35 Delete Account

Use Case ID:	UC-35
Use Case Name:	Delete Account
Actors:	Primary Actors: diabetic patient
Description:	The patient should be allowed to delete his account
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on deleting account.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: The patients account will be deleted. Post-1: Register form will be displayed on screen
Normal Flow:	1. Patient selects my profile. 2. Patient profile will be displayed. 3. Patient selects delete account. 4. Application prompt the patient to ask if he really wants to delete his account. 5. Patient selects yes. (E1) 6. Application show message indication account deleted.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	**E1: If patient selects No:** 1. User case will be cancelled.
Business Rules:	
Assumptions:	

2.2.3.36 Edit Profile

Table 2-38: UC-36 Edit Profile

Use Case ID:	UC-36
Use Case Name:	Edit Profile
Actors:	Primary Actors: diabetic patient
Description:	The patient shall be able to edit his personal details on his profile.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on live chatting with other patients.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: Patients profile information will be updated. Post-2: A status message indicating changes saved will be displayed.
Normal Flow:	1. Patient selects profile. 2. Application displays patient's profile. 3. Patient then selects password to edit. (1.1, 1.2, 1.3) 4. Application prompts the user to enter previous password. 5. Patient enters previous password. 6. Application prompts the user to enter new password.
	7. Patient enters the new password. 8. Application prompts the patient to reenter the password. 9. Patient re-enters the password. (E1) 1. Patient indicates on save changes.
Alternative Flows: **[Alternative Flow 1 – Not in Network]**	**1.1 If the patient selects weight to change.** **1.** The application asks the patient to enter weight. **2.** Patient enters his updated weight. **3.** Returns on normal flow step 10. **1.2 If the patient selects height to change.** 1. Application asks the patient to enter height. 2. Patient enters his updated height. 3. Returns on normal flow step 10. **1.3 If a patient selects profile picture to change.** 1. The application asks the patient to select a picture. 2. Patient selects picture. 3. Returns on normal flow step 10.

Exceptions:	**E1: Both passwords do not match.** 1. Application shows a message that passwords do not match. 2. Application prompts the user to re-enter passwords. 3. Return to step 9 of normal flow.
Business Rules:	
Assumptions:	

2.2.3.37 Post on the discussion board

Table 2-39: UC-37 Post on the discussion board

Use Case ID:	UC-37
Use Case Name:	Post on the discussion board
Actors:	Primary Actors: diabetic patient
Description:	If the patient has any queries regarding diabetes, he can post them on the discussion board. Other patients can read that query and reply to it.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on posting in discussion board.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Post-1: Patient's query will be posted on the discussion board. Post-2: Query will be visible to all other patients.
Normal Flow:	1. Patient selects discussion board. 2. Patient selects new post. 3. Application will display a page where the patient can write its query. 4. Patient writes its query.
	5. Patient indicates on posting it.
Alternative Flows: [Alternative Flow 1 – Not in Network]	
Exceptions:	
Business Rules:	

Assumptions:	

2.2.3.38 *Reply on the discussion board*

Table 2-40: UC-38 Reply on the discussion board

Use Case ID:	UC-38
Use Case Name:	Reply on the discussion board
Actors:	Primary Actors: diabetic patient
Description:	Diabetic patients can select a particular query on the discussion board, read it and reply to it.
Trigger:	The use-case would be triggered as soon as the diabetic patient would indicate on posting in discussion board.
Preconditions:	Pre-1: The patient is logged into the application.
Postconditions:	Pos-1: Patient’s reply will be posted. Pos-2: Patient’s reply will be visible to all other patients.
Normal Flow:	1. Patient’s selects the query to which he wants to reply. 2. Patient indicates on replying. 3. Application will prompt the user to write his reply 4. Patient’s writes his reply 5. Patient indicates on posting the reply.
Alternative Flows:	
Exceptions:	
Business Rules	
Assumptions:	1. Patients will reply to the query if they have an answer regarding the query posted by other patients-.

2.2.4 Event Response Table

Table 2-41: ERT-1 Whenever a newly added medication has active agents, user have allergy with.

ID	Event	Data Element	System State	Exception Condition	System Response
1	Whenever a newly added medication has active agents, user have allergy with.	Allergic Reaction History, Medicine active Ingredients	Currently no warning for medication usage is shown.	1. Medication ingredients data is not available.	The warning message is shown to user about the medication having active agent user had allergy in the past

Table 2-42: ERT-2 Whenever a newly added medication has the probability of side effects

ID	Event	Data Element	System State	Exception Condition	System Response
2	Whenever a newly added medication has the probability of side effects.	Medication Details	Currently no alert is shown regarding medication side effects.	1. Medication Side effects are not available.	An alert is generated for the awareness of side effects the medication may can cause.

Table 2-43: ERT-3 When the medication intake time is reached.

ID	Event	Data Element	System State	Exception Condition	System Response
3	When the medication intake time is reached.	Current Medical Prescription	Currently no alarm is activated reminding the medication taking.	1. An alarm is snoozed for user's selected time interval.	When the medication intake time of the day is reached the alarm is activated and shown to the user.

Table 2-44: ERT-4 Whenever a newly added meal has

ID	Event	Data Element	System State	Exception Condition	System Response
4	Whenever a newly added meal has ingredients, user has allergy with.	Allergic Reaction History, Meal Ingredients	Currently no warning for food consumption is shown.	1.Food ingredients data is not available.	A warning message is shown to user about the food containing ingredients user has allergy with.

Table 2-45:ERT-5 When the user intakes a meal

ID	Event	Data Element	System State	Exception Condition	System Response
5	When the user intakes a meal with more than allowed calories.	Insulin Information, Meal Information	Currently no change is made to the user's bolus insulin intake, allowed carbohydrates and a diet chart.	1.The user's medication may not include insulin, then only diet chart and allowed carbohydrates get updated.	The user's bolus insulin units are increased to a level where the calories overconsumption can be balanced. Moreover, allowed carbohydrates for the day and the diet chart are also updated to balance the overconsumption.

Table 2-46: ERT-6 Whenever the user saves a meal intake.

ID	Event	Data Element	System State	Exception Condition	System Response
6	Whenever the user saves a meal intake.	Meal Calories Information, Food types, Meal Ingredients	Currently there is no notification shown.		A notification is shown to enter the current blood sugar concentration.

Table 2-47: ERT-7 Whenever the user adds a new blood sugar instance

ID	Event	Data Element	System State	Exception Condition	System Response
7	Whenever the user adds a new blood sugar instance with after meal event.	Blood Sugar Record	The current bolus unit and allowed carbohydrates and a diet chart are declared according to old blood sugar reading.		The allowed carbohydrates, diet chart and bolus insulin units are changed to maintain normal blood sugar.

Table 2-48: ERT-8 Whenever the user adds a meal

ID	Event	Data Element	System State	Exception Condition	System Response
8	Whenever the user adds a meal consumption through image	Meal Ingredients	Currently no food options are visible for selection.	1. There is no such meal information available.	The food options similar to the selected food are made visible to the user

Table 2-49: ERT 9 Whenever a user selects a food

ID	Event	Data Element	System State	Exception Condition	System Response
9	Whenever a user selects a food from the food options after taking food image.	Meal Ingredients	Currently calories for the food are undefined.		The calories for the food are calculated based on the quantity are made visible to the user.

Table 2-50: ERT 10 Whenever the user adds a new health factor

ID	Event	Data Element	System State	Exception Condition	System Response
10	Whenever the user adds a new health factor (bloodsugar, bloodpressure or cholesterol) instance.	Blood Pressure Record, Bloodsugar Record, Carbohydrates Record	The current statistics display the result of all the old record for the heath factor.		The statistics for the heath factor immediately change to display the new instance along with old record.

Table 2-51: ERT 11 When the Exercise routine is not initiated for a whole day

ID	Event	Data Element	System State	Exception Condition	System Response
11	When the Exercise routine is not initiated for a whole day.	Exercise plan, Food and Exercise database, sugar level history.	The exercise routine for the current day is not initiated.	1.The exercise routine is completed halfway to the completion	The exercise routine for a day is marked as incomplete.

Table 2-52: ERT 12 When the bloodsugar average is low for the day.

ID	Event	Data Element	System State	Exception Condition	System Response
12	When the bloodsugar average is low for the day.	Blood-sugar Record, Exercise Routine Status.	The exercise routine is planned but not initiated.		The exercise routine for a day is marked as cancelled.

Table 2-53: ERT 13 When the exercise routine was

ID	Event	Data Element	System State	Exception Condition	System Response
13	When the exercise routine was incomplete and blood-sugar average was also high for the day.	Blood-Sugar Record, Exercise Routine Status.	Currently the diet and exercise plan remain unchanged.		The user’s diet and exercise chart changes to a level where the bloodsugar can be maintained.

2.3 Functional Requirements

Following section contains the functional requirements related to Smart Diabetolog application.

2.3.1 Register

2.3.1.1 Register.Fullname

Identifier	**FR-2.3.1.1**
Title	Register.Fullname
Requirement	The patient shall be able to enter his full name.
Source	Kashaf Fatima
Rationale	The patient’s full name should be added to setup his account to save his medical information.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.1.2 Register.Email

Identifier	**FR-2.3.1.2**
Title	Register.Email
Requirement	The patient shall be able to enter his email.
Source	Kashaf Fatima
Rationale	The patient’s email should be added to uniquely identify each patient from other.
Business (if Rule required)	BR-1
Dependencies	
Priority	High

2.3.1.3 Register.Email.alreadyExist

Identifier	**FR-2.3.1.3**
Title	Register.Email.alreadyExist
Requirement	If the email is already registered into the application, the application shall prompt the patient to either change the email or cancel registration.
Source	Kashaf Fatima
Rationale	The patient's email should uniquely identify each account, so email must be unique for each for every patient. Application must restrict patients to register more than one account.
Business Rule (if required)	BR-2
Dependencies	
Priority	High

2.3.1.4 Register.Email.Wrong

Identifier	**FR-2.3.1.4**
Title	Register.Email.Wrong
Requirement	If the email entered by the patient does not meet the standards of email address the application shall be able to prompt the user to recheck email.
Source	Kashaf Fatima
Rationale	Email will be used to make patients profile so application must confirm it to be right.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.1.5 Register.Gender

Identifier	**FR-2.3.1.5**
Title	Register.Gender
Requirement	The patient shall be able to enter his gender.
Source	Kashaf Fatima
Rationale	Gender information will be used by application to create personalized plans.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.1.6 Register.Weight

Identifier	**FR-2.3.1.6**
Title	Register.Weight
Requirement	The patient shall be able to enter his weight.
Source	Kashaf Fatima
Rationale	Weight information will be used to calculate BMI of the patient which will further be used to create plans.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.1.7 Register.Height

Identifier	**FR-2.3.1.7**
Title	Register.Height
Requirement	The patient shall be able to enter his height
Source	Kashaf Fatima
Rationale	Height information will be used to calculate BMI of the patient which will further be used to create plans.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.1.8 Register.Password

Identifier	**FR-2.3.1.8**
Title	Register.Password
Requirement	The system shall allow the patient to set a password for his account.
Source	Kashaf Fatima
Rationale	The patient should be able to secure his account through password.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.1.9 Register.Password.weak

Identifier	**FR-2.3.1.9**
Title	Register.Password.weak
Requirement	If patient, try to set up a weak password, the application shall be able to prompt the user to set up a password with at least 6 alphabets and digits.
Source	Kashaf Fatima

Rationale	The patient’s password must not be something that can be easily guessed to secure his medical information.
Business (if Rule required)	
Dependencies	
Priority	medium

2.3.1.10 Register.Password.Match

Identifier	**FR-2.3.1.10**
Title	Register.Password.Match
Requirement	If password matches the re password, the application shall allow the patient to submit registration form.
Source	Kashaf Fatima
Rationale	Application should re-assure the password as patient will be allowed to access application by email and password.
Business (if Rule required)	
Dependencies	
Priority	medium

2.3.1.11 Register.Password.Match.Not

Identifier	**FR-2.3.1.11**
Title	Register.Password.Match.Not
Requirement	If the password and re-password does not match, the application shall be able to prompt the patient to re-check password, to submit the form.
Source	Kashaf Fatima
Rationale	Application should re-assure the password as patient will be allowed to access application by email and password.
Business Rule (if required)	
Dependencies	
Priority	medium

2.3.1.12 . Register.Save.Code.Send

Identifier	**FR-2.3.1.12**
Title	Register.Save.Code.Send
Requirement	After the completion of the register form if the patient indicates on saving the form the application shall be able to send code on patients email to verify.
Source	Kashaf Fatima
Rationale	Application should make sure that the patient's email is his own.
Business Rule (if required)	
Dependencies	
Priority	medium

2.3.1.13 Register.Save.Code.Enter

Identifier	**FR-2.3.1.13**
Title	Register.Save.Code.Enter
Requirement	The application shall be able to allow the patient to enter code to verify him.
Source	Kashaf Fatima
Rationale	Application should re-assure the password as patient will be allowed to access application by email and password.
Business Rule (if required)	
Dependencies	
Priority	medium

2.3.1.14 Register.Save.Code.Match

Identifier	**FR-2.3.1.14**
Title	Register.Save.Code.Match
Requirement	If the code entered by the patient matches the code send by the application the application shall be able to create the patient's account.
Source	Kashaf Fatima
Rationale	Application should allow the creation of the account after complete verification.
Business Rule (if required)	
Dependencies	

Priority	medium

2.3.1.15 Register.Save.Code.Match.Not

Identifier	**FR-2.3.1.15**
Title	Register.Save.Code.Match.Not
Requirement	If the code entered by the patient does not matches the code send by the application the application shall be able to prompt the user to either re-check the email or cancel the registration.
Source	Kashaf Fatima
Rationale	Application should re-assure the password as patient will be allowed to access application by email and password.
Business Rule (if required)	
Dependencies	
Priority	medium

2.3.2 Login

2.3.2.1 Login.Email.Enter

Identifier	**FR-2.3.2.1**
Title	Login.Email.Enter
Requirement	The patient shall be able to enter the email to login in his account.
Source	Kashaf Fatima
Rationale	Application should allow the patient to login to his account by adding his email and password.
Business Rule (if required)	
Dependencies	BR-1
Priority	High

2.3.2.2 Login.Email.notExist

Identifier	**FR-2.3.2.2**

Title	Login.Email.notExist
Requirement	If the patient tries to login with a non-registered email, the application shall be able to prompt the patient about wrong email and ask him to reenter email.
Source	Kashaf Fatima
Rationale	The patient's personal information must be protected.
Business (if Rule required)	
Dependencies	BR-1
Priority	High

2.3.2.3 Login.Password

Identifier	**FR-2.3.2.3**
Title	Login.Password
Requirement	The patient must be able to enter his account using his password, set at time of registration.
Source	Kashaf Fatima
Rationale	The patient's personal information must be protected.
Business (if Rule required)	
Dependencies	BR-1
Priority	High

2.3.2.4 Login.Password.Wrong

Identifier	**FR-2.3.2.4**
Title	Login.Password.Wrong
Requirement	If the patient enters wrong password the application shall be able to prompt user about wrong password and ask him to re enter it.

Source	Kashaf Fatima
Rationale	The patient personal information must be protected.
Business Rule (if required)	
Dependencies	BR-1
Priority	High

2.3.2.5 Login.LostConnection

Identifier	**FR-2.3.2.5**
Title	Login.lostConnection
Requirement	If the patient tries to enter with no internet connection the application shall be able to prompt the patient to reconnect internet.
Source	Kashaf Fatima
Rationale	The internet must be available during login to authenticate user.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.3 Add new oral Medication

2.3.3.1 Prescription.Select

Identifier	**FR-2.3.3.1**
Title	Prescription.Select
Requirement	The system shall allow the user to select the prescription.
Source	Fatima Shahzad
Rationale	The user may want to choose a particular prescription
Business Rule (if required)	BR-2, BR-3

Dependencies	
Priority	Medium

2.3.3.2 Prescription.New

Identifier	**FR-2.3.3.2**
Title	Prescription.New
Requirement	The system shall allow the user to create a new prescription.
Source	Fatima Shahzad
Rationale	The user may be prescribed a a new prescription.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	Medium

2.3.3.3 Prescription.Name

Identifier	**FR-2.3.3.3**
Title	Prescription.Name
Requirement	The system shall allow the user to enter a name for the newly created prescription.
Source	Fatima Shahzad
Rationale	The user may want to name their prescription specifically.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	Medium

2.3.3.4 Medicine.Category.Select

Identifier	**FR-2.3.3.4**
Title	Medicine.Category.Select
Requirement	The user shall be able to select the medication category from oral medication or insulin medication.
Source	Fatima Shahzad
Rationale	The user may be on non-diabetic medication along with diabetic medications and wants to keep the record of both.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	High

2.3.3.5 Medicine.Category.Add

Identifier	**FR-2.3.3.5**
Title	Medicine.Category.Add
Requirement	The system shall allow the user to add a medication instance in the selected category.
Source	Fatima Shahzad
Rationale	The user may want to add all the medication he is currently using so that the system can remind him by showing different alarms.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4
Priority	High

2.3.3.6 Medicine.Category.Add.Name

Identifier	**FR-2.3.3.6**
Title	Medicine.Category.Add.Name
Requirement	The system shall allow the user to write the name of the medication manually.
Source	Fatima Shahzad

Rationale	Medication name would help the user to recognize the medicine when alarms would be shown.
	Moreover, the allergic reaction predictions would also be accessed on the basis of name.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5
Priority	High

2.3.3.7 Medicine.Category.Add.Name.Picture

Identifier	**FR-2.3.3.7**
Title	Medicine.Category.Add.Name.Picture
Requirement	The system shall allow the user to take the picture of the name of the medication.
Source	Fatima Shahzad
Rationale	The user may not be interested in writing the medication name manually.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5
Priority	High

2.3.3.8 Medicine.Category.Add.Unit

Identifier	**FR-2.3.3.8**
Title	Medicine.Category.Add.Unit
Requirement	The system shall allow the user to write the unit of the medication used in mg(milligram).
Source	Fatima Shahzad
Rationale	Different milligrams of the same medication are available.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5
Priority	High

2.3.3.9 Medicine.Category.Add.Time

Identifier	**FR-2.3.3.9**
Title	Medicine.Category.Add.Time
Requirement	The system shall allow the user to select the time of the day the medication would be used.
Source	Fatima Shahzad
Rationale	An alarm reminding the medication taking would be displayed on this time to the user.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5, FR-4.6
Priority	High

2.3.3.10 Medicine.Category.Add.Time.Multiple

Identifier	**FR-2.3.3.10**
Title	Medicine.Category.Add.Time.Multiple
Requirement	The system shall allow the user to select multiple time intervals of the day when the medication would be used.
Source	Fatima Shahzad
Rationale	An alarm reminding the medication taking would be displayed on recorded time intervals to the user.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5, FR-4.6
Priority	High

2.3.3.11 Medicine.Category.Add.Dose

Identifier	**FR-2.3.3.11**
Title	Medicine.Category.Add.Dose
Requirement	The system shall allow the user to enter the dose of the medication to be used according to prescription.
Source	Fatima Shahzad

Rationale	An alarm reminding the medication taking would be displayed with the dosage information so that the user won’t forget the quantity.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5, FR-4.6
Priority	High

2.3.3.12 Record.Save

Identifier	**FR-2.3.3.12**
Title	Record.Save
Requirement	If the user initiates to save the record, then the system shall be able to save the record.
Source	Fatima Shahzad
Rationale	The record for the entered instance would be stored.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-4.4, FR-4.5, FR-4.6
Priority	High

2.3.4 Add Insulin Medication

2.3.4.1 Insulin.Add

Identifier	**FR-2.3.4.1**
Title	Insulin.Add
Requirement	The system shall allow the user to add its injectable medication insulin.
Source	Fatima Shahzad
Rationale	Diabetes type-1 and few diabetes type-2 patients are on insulin, and they need to keep its injecting records on different time intervals.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	
Priority	High

2.3.4.2 Insulin.Add.Second

Identifier	**FR-2.3.4.2**
Title	Insulin.Add.Second
Requirement	The system shall allow the user to add its second injectable medication insulin.
Source	Fatima Shahzad
Rationale	Diabetic type-1 patients use to insulin one long and one fast acting insulin according to Basal-bolus regimen.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	
Priority	High

2.3.4.3 Insulin.Type

Identifier	**FR-2.3.4.3**
Title	Insulin.Type
Requirement	The system shall allow the user to select the type of insulin, from longacting and fast-acting insulin.
Source	Fatima Shahzad
Rationale	Long-acting insulin is used to manage the baseline glucose while the fastacting insulin manages the glucose peek for different meals hence they both would be managed separately.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	
Priority	High

2.3.4.4 Insulin.Name

Identifier	**FR-2.3.4.4**
Title	Insulin.Name
Requirement	The system shall allow the user to select the name of the insulin, the user would be using.
Source	Fatima Shahzad
Rationale	The insulin has different formula; hence its record should be maintained.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-5.1, FR-5.3
Priority	High

2.3.4.5 Insulin.Type.Fast.Sensitivity

Identifier	**FR-2.3.4.5**
Title	Insulin.Type.Fast.Sensitivity
Requirement	The system prompts the user to enter the insulin sensitivity factor (ISF) for the fast-acting insulin.
Source	Fatima Shahzad
Rationale	ISF helps to know how many glucose levels would be lowered by 1 unit of fast-acting insulin.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-5.1, FR-5.3, FR-5.4
Priority	High

2.3.4.6 Insulin.Type.Fast.CarbRatio

Identifier	**FR-2.3.4.6**
Title	Insulin.Type.Fast.CarbRatio
Requirement	The system prompts the user to enter the carb ratio for the fast-acting insulin.
Source	Fatima Shahzad

Rationale	Carb ratio is used to determine 1 unit insulin would cover how many carbohydrates.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-5.1, FR-5.3, FR-5.4
Priority	High

2.3.4.7 Insulin.Type.Long.Units

Identifier	**FR-2.3.4.7**
Title	Insulin.Type.Long.Units
Requirement	The system prompts the user to enter the units of the long-acting insulin the user would be using.
Source	Fatima Shahzad
Rationale	Long-acting insulin units would be required to adjust baseline blood sugar at normal level.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-5.1, FR-5.3, FR-5.4
Priority	High

2.3.4.8 Insulin.Type.Long.Time

Identifier	**FR-2.3.4.8**
Title	Insulin.Type.Long.Time
Requirement	The system prompts the user to enter the time the long-acting insulin the user would be taken.
Source	Fatima Shahzad
Rationale	Long-acting insulin are mostly taken at night before bed, and its effect continues for the next 24 hours.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-5.1, FR-5.3, FR-5.4, FR-5.7
Priority	High

2.3.4.9 Insulin.Type.Long.Time.No

Identifier	**FR-2.3.4.9**
Title	Insulin.Type.Long.Time.No
Requirement	If the user did not enter the time for the long-acting insulin then the system would assume it to be 10:00 PM.
Source	Fatima Shahzad
Rationale	An ideal time to bed is 10:00 PM. So, when the user forgets to select time, it would be selected as 10 PM because the long-acting insulin is normally used before going to bed.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-5.1, FR-5.3, FR-5.4, FR-5.7
Priority	High

2.3.5 View Current Oral Medication

2.3.5.1 Medicine.Category.History.Created

Identifier	**FR-2.3.5.1**
Title	Medicine.Category.History.Created
Requirement	If the history containing the medication related to the category is maintained then the system shall be able to display them to the user.
Source	Fatima Shahzad
Rationale	The user would be able to view all the medications related to the category, he is currently using as a prescription.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.5.2 Medicine.Category.History.Created.No

Identifier	**FR-2.3.5.2**
Title	Medicine.Category.History.Created.No
Requirement	If the medication history related to the category is not already created then the system shall be able to display a message "No records are present".
Source	Fatima Shahzad
Rationale	The user would be aware that he had not entered any current medication.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.5.3 Medicine.Category.View

Identifier	**FR-2.3.5.3**
Title	Medicine.Category.View
Requirement	The system shall allow the user to view all the user entered medications related to the selected category.
Source	Fatima Shahzad
Rationale	The user would be able to examine all the medications he had already entered into the system category wise.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.6 Update Oral Medication

2.3.6.1 Medicine.Category.View

Identifier	**FR-2.3.6.1**
Title	Medicine.Select.Category.Instance.Select
Requirement	The system shall allow the user to select any particular medication instance present in the displayed records of the medication type category.
Source	Fatima Shahzad
Rationale	The user may want to view the medication information it saved in detail.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-6.3
Priority	Medium

2.3.6.2 Medicine.Select.Category.Instance.Select.Display

Identifier	**FR-2.3.6.2**
Title	Medicine.Select.Category.Instance.Select.Display
Requirement	The system shall be able to display all the information related to the selected medication instance of the selected medication category.
Source	Fatima Shahzad
Rationale	The user may want to view the stored information about the medication.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-7.1
Priority	Medium

2.3.6.3 Medicine.Select.Category.Instance.Select.Update

Identifier	**FR-2.3.6.3**
Title	Medicine.Select.Category.Instance.Select.Update
Requirement	The system shall allow the user to update the information of the selected medication instance.
Source	Fatima Shahzad

Rationale	The user may want to change the current medication details.
Business Rule (if required)	BR-2, BR-3
Dependencies	FR-6.3, FR-7.1
Priority	Medium

2.3.6.4 Instance.Select.Update.Cancel

Identifier	**FR-2.3.6.4**
Title	Instance.Select.Update.Cancel
Requirement	The system shall allow the user to cancel updating of the selected instance.
Source	Fatima Shahzad
Rationale	The user may have initiated the updating action by mistake and wants to cancel it.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	Medium

2.3.6.5 Instance.Select.Update.Delete

Identifier	**FR-2.3.6.5**
Title	Instance.Select.Update.Delete
Requirement	The system shall allow the user to delete the selected instance.
Source	Fatima Shahzad
Rationale	The user may want to delete the instance.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	High

2.3.6.6 Success.Message

Identifier	**FR-2.3.6.6**

Title	Success.Message
Requirement	The system shall display a message showing “Changes are successfully made” to the user.
Source	Fatima Shahzad
Rationale	The user would be satisfied that the changes he made are successfully updated.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	Medium

2.3.6.7 Success.Message.No

Identifier	**FR-2.3.6.7**
Title	Success.Message.No
Requirement	The system shall display a message showing “Changes failed! Try Again”.
Source	Fatima Shahzad
Rationale	The user would be made aware that the changes he made were not successfully updated.
Business Rule (if required)	BR-2, BR-3
Dependencies	
Priority	Medium

2.3.7 View current insulin medication

2.3.7.1 Insuline.History.Created

Identifier	**FR-2.3.7.1**
Title	Insuline.History.Created
Requirement	If the history containing the current insulin medication is maintained then the system shall be able to display them to the user.
Source	Fatima Shahzad

Rationale	The user would be able to view all the insulin he is currently injecting.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	
Priority	High

2.3.7.2 Insuline.History.Created.No

Identifier	**FR-2.3.7.2**
Title	Insulin.History.Created.No
Requirement	If the insulin history is not already created then the system shall be able to display a message “No records are present”.
Source	Fatima Shahzad
Rationale	The user would be aware that he had not entered any current insulin information.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	
Priority	High

2.3.7.3 Insuline.View

Identifier	**FR-2.3.7.3**
Title	Insulin.View
Requirement	The system shall allow the user to view the insulins he is currently injecting.
Source	Fatima Shahzad
Rationale	The user would be able to examine all the insulin he had already entered into the system.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-8.1
Priority	High

2.3.8 Update insulin medication

2.3.8.1 Insuline.Instance.Select

Identifier	**FR-2.3.8.1**
Title	Insulin.Instance.Select
Requirement	The system shall allow the user to select any particular insulin medication instance.
Source	Fatima Shahzad
Rationale	The user may want to view the insulin medication information he saved, in detail.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-8.3
Priority	High

2.3.8.2 Insuline.Instance.Select.Display

Identifier	**FR-2.3.8.2**
Title	Insulin.Instance.Select.Display
Requirement	The system shall be able to display all the information related to the selected insulin instance.
Source	Fatima Shahzad
Rationale	The user may want to view the stored information about the insulin.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-9.1
Priority	High

2.3.8.3 Insuline.Instance.Select.Update

Identifier	**FR-2.3.8.3**
Title	Insulin.Instance.Select.Update
Requirement	The system shall allow the user to update the information of the selected insulin instance.

Source	Fatima Shahzad
Rationale	The user may want to change the current insulin details.
Business Rule (if required)	BR-2, BR-4, BR-8
Dependencies	FR-9.1, FR-9.2
Priority	High

2.3.9 . View Current Medication Prescription

2.3.9.1 Medicine.Prescription.Display

Identifier	**FR-2.3.9.1**
Title	Medicine.Prescription.Display
Requirement	The system shall display the current medication prescription user is consuming to the user.
Source	Fatima Shahzad
Rationale	The user may want to check all the medications it is currently consuming.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.10 View Prescription History

2.3.10.1 Prescription.Available

Identifier	**FR-2.3.10.1**
Title	Prescription.Available
Requirement	If the prescriptions are available, the system shall be able to display the available prescriptions to the user.
Source	Fatima Shahzad
Rationale	The user may want to view all the available prescription.

Business Rule (if required)	BR-1, BR-2
Dependencies	
Priority	Medium

2.3.10.2 Prescription.AvailableNo

Identifier	**FR-2.3.10.2**
Title	Prescription.Available.No
Requirement	If no prescription is available, the system shall be able to display the message that no prescriptions are added yet.
Source	Fatima Shahzad
Rationale	The user shall be made aware that no prescriptions are added yet.
Business Rule (if required)	BR-1, BR-2
Dependencies	
Priority	Medium

2.3.10.3 Prescription.View

Identifier	**FR-2.3.10.3**
Title	Prescription.View
Requirement	The system shall allow the user to view all the stored medication prescriptions.
Source	Fatima Shahzad
Rationale	The user may want to examine the prescriptions he had used in the past
Business Rule (if required)	BR-1, BR-2
Dependencies	
Priority	Medium

2.3.10.4 Prescription.View.Title

Identifier	**FR-2.3.10.4**
Title	Prescription.View.Title
Requirement	The system shall allow the user to view all the medication prescriptions by their title name.
Source	Fatima Shahzad
Rationale	The user may want to search the prescriptions by using its title name.
Business Rule (if required)	BR-1, BR-2
Dependencies	
Priority	Medium

2.3.10.5 Prescription.View.Month

Identifier	**FR-2.3.10.5**
Title	Prescription.View.Month
Requirement	The system shall allow the user to view all the medication prescriptions according to their months.
Source	Fatima Shahzad
Rationale	The user may want to search the prescriptions by using the month.
Business Rule (if required)	BR-1, BR-2
Dependencies	
Priority	Medium

2.3.10.6 Prescription.Instance.Select

Identifier	**FR-2.3.10.6**
Title	Prescription.Instance.Select
Requirement	The system shall allow the user to select any particular medication prescription instance present in the displayed prescriptions.
Source	Fatima Shahzad
Rationale	The user may want to view the detailed information about the prescription.

Business Rule (if required)	BR-1, BR-2
Dependencies	FR-11.1
Priority	Medium

2.3.10.7 Prescription.Instance.Select.Display

Identifier	**FR-2.3.10.7**
Title	Prescription.Instance.Select.Display
Requirement	The system shall be able to display all the information related to the selected prescription instance.
Source	Fatima Shahzad
Rationale	The user may want to view the stored information about the prescription.
Business Rule (if required)	BR-1, BR-2
Dependencies	FR-11.1, FR-11.4
Priority	Medium

2.3.11 Change Prescription Title

2.3.11.1 Prescription.Instance.Title.Change

Identifier	**FR-2.3.11.1**
Title	Prescription.Instance.Title.Change
Requirement	The system shall allow the user to change the title of the selected prescription instance.
Source	Fatima Shahzad
Rationale	The user may want to change the title of any prescription.
Business Rule (if required)	BR-1, BR-2
Dependencies	FR-11.1, FR-11.4, FR-11.6
Priority	Medium

2.3.12 Add Allergic Reaction History

2.3.12.1 AllergyReaction.Add

Identifier	**FR-2.3.12.1**
Title	AllergyReaction.Add
Requirement	The system shall allow the user to add any allergic reaction history he experienced in the past.
Source	Fatima Shahzad
Rationale	The user would get a warning if he would try to use any medication or food he had allergy in the past.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.12.2 AllergyReaction.Select.Category

Identifier	**FR-2.3.12.2**
Title	AllergyReaction.Select.Category
Requirement	The system shall allow the user to select a category of the allergic reaction from diabetic medication, non-diabetic medication, Insulin, or food.
Source	Fatima Shahzad
Rationale	The diabetic patient would be able to enter any of the allergic reaction he had experienced in the past.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.12.3 AllergyReaction.Category.Name

Identifier	**FR-2.3.12.3**
Title	AllergyReaction.Category.Name
Requirement	The user shall be able to write the name of the selected category type instance he had allergic reaction in the past.
Source	Fatima Shahzad
Rationale	The name of the instance would help to locate its presence in any of the consumptions the user would have in the future event.
Business Rule (if required)	BR-2
Dependencies	FR-13.2
Priority	Medium

2.3.12.4 AllergyReaction.Category.Name.Symptoms

Identifier	**FR-2.3.12.4**
Title	AllergyReaction.Category.Name.Symptoms
Requirement	The user shall be able to select the symptoms he experienced with the selected category instance as a result of allergic reaction in the past.
Source	Fatima Shahzad
Rationale	All the details related to the allergic reaction incident would be displayed to the user when he would be about to consume any food or medication with the same name or ingredients.
Business Rule (if required)	BR-2
Dependencies	FR-13.2, FR-13.3
Priority	Medium

2.3.12.5 AllergyReaction.Category.Name.Description

Identifier	**FR-2.3.12.5**
Title	AllergyReaction.Category.Name.Description
Requirement	The user shall be able to add a small description about the incident for its ease.
Source	Fatima Shahzad

Rationale	The user would be able to add any details about the allergic reaction that he might forget in the future.
Business Rule (if required)	BR-2
Dependencies	FR-13.2, FR-13.3
Priority	Medium

2.3.12.6 AllergyReaction.Select.Category.Multiple

Identifier	**FR-2.3.12.6**
Title	AllergyReaction.Select.Category.Multiple
Requirement	The system shall allow the user to add more than one allergic reaction for the selected category.
Source	Fatima Shahzad
Rationale	The user would not need to repeat the step of first selecting the category then add new allergic reaction, he can add all the allergies related to the category in one go.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.13 View Allergic Reaction History

2.3.13.1 AllergyReaction.Category.History.Created

Identifier	**FR-2.3.13.1**
Title	AllergyReaction.Category.History.Created
Requirement	If the history containing the allergic reactions related to the category is maintained then the system shall be able to display them to the user.
Source	Fatima Shahzad
Rationale	The user would be able to view its allergic reactions related to the history.
Business Rule (if required)	BR-2

Dependencies	
Priority	Medium

2.3.13.2 AllergyReaction.Category.History.Created.No

Identifier	**FR-2.3.13.2**
Title	AllergyReaction.Category.History.Created.No
Requirement	If the history related to the category is not already created then the system shall be able to display a message "No records are present".
Source	Fatima Shahzad
Rationale	The user would be made aware that he had not maintained any allergic reaction history yet.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.13.3 AllergyReaction.Category.Select

Identifier	**FR-2.3.13.3**
Title	AllergyReaction.Category.Select
Requirement	The system shall allow the user to select an allergic reaction category from diabetic, non-diabetic medication, insulin or food.
Source	Fatima Shahzad
Rationale	The user may be allergic to any of the allergic reaction categories.
Business Rule (if required)	BR-2
Dependencies	
Priority	Medium

2.3.13.4 AllergyReaction.Select.Category.View

Identifier	**FR-2.3.13.4**
Title	AllergyReaction.Select.Category.View

Requirement	The system shall allow the user to view all the user entered allergic reactions related to the selected category.
Source	Fatima Shahzad
Rationale	The user would be able to view the reactions he had already entered into the system.
Business Rule (if required)	BR-2
Dependencies	FR-14.1, FR-14.3
Priority	Medium

2.3.14 View Allergic Reaction Instance

2.3.14.1 AllergyReaction.Select.Category.Instance.Select

Identifier	**FR-2.3.14.1**
Title	AllergyReaction.Select.Category.Instance.Select
Requirement	The system shall allow the user to select an allergic instance present in the displayed records of the allergic reaction category.
Source	Fatima Shahzad
Rationale	The user may want to view the allergic reaction information in detail.
Business Rule (if required)	BR-2
Dependencies	FR-14.4
Priority	Medium

2.3.14.2 AllergyReaction.Category.Instance.Display

Identifier	**FR-2.3.14.2**
Title	AllergicReaction.Category.Instance.Display
Requirement	The system shall be able to display all the information related to the selected instance of the allergic reaction category.
Source	Fatima Shahzad
Rationale	The user may want to analyze the stored information about the allergic reaction experience.

Business Rule (if required)	BR-2
Dependencies	FR-14.4, FR-15.1
Priority	Medium

2.3.15 Update Allergic Reaction History

2.3.15.1 AllergyReaction.Category.Instance.Update

Identifier	**FR-2.3.15.1**
Title	AllergyReaction.Category.Instance.Update
Requirement	The system shall allow the user to update the information of the allergic reaction instance.
Source	Fatima Shahzad
Rationale	The user may want to change the allergic reaction details.
Business Rule (if required)	BR-2
Dependencies	FR-14.4, FR-15.1
Priority	Medium

2.3.16 View Tracker

2.3.16.1 Tracker.View

Identifier	**FR-2.3.16.1**
Title	Tracker.View
Requirement	The system shall allow the user to view the tracker.
Source	Fatima Shahzad
Rationale	The user may want to track all of its health-related details.
Business Rule (if required)	BR-2, BR-5, BR-6, BR-7
Dependencies	
Priority	High

2.3.16.2 Tracker.Category.Select

Identifier	**FR-2.3.16.2**
Title	Tracker.Category.Select
Requirement	The system shall allow the user to select a health category he wants to track.
Source	Fatima Shahzad
Rationale	The user may want to track all health details for a particular category.
Business Rule (if required)	BR-2, BR-5, BR-6, BR-7
Dependencies	
Priority	High

2.3.16.3 Tracker.Category.Result.Display

Identifier	**FR-2.3.16.3**
Title	Tracker.Category.Result.Display
Requirement	The system shall be able to display the records of the selected category.
Source	Fatima Shahzad
Rationale	The user may want to view its record.
Business Rule (if required)	BR-2, BR-5, BR-6, BR-7
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.16.4 Tracker.Category.Result.Empty.Display

Identifier	**FR-2.3.16.4**
Title	Tracker.Category.Result.Empty.Display
Requirement	If the system does not have record for the selected tracker category, the system shall display a message “No records for the category exists”.
Source	Fatima Shahzad
Rationale	The user should be made aware that he had not entered any records for the category.

Business Rule (if required)	BR-2, BR-5, BR-6, BR-7
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.17 Stop Health Factor Tracking

2.3.17.1 Tracker.Setting.View

Identifier	**FR-2.3.17.1**
Title	Tracker.Setting.View
Requirement	The system shall allow the user to view the settings of the health factor tracker.
Source	Fatima Shahzad
Rationale	The user may want to view the factors the health factors the tracker is currently tracking
Business Rule (if required)	BR-2
Dependencies	FR-17.1
Priority	Medium

2.3.17.2 Tracker.Factor.Stop

Identifier	**FR-2.3.17.2**
Title	Tracker.Factor.Stop
Requirement	The system shall allow the user to stop the tracking of any health factor other than blood-sugar.
Source	Fatima Shahzad
Rationale	The user may not want to track all the health factors other than bloodsugar.
Business Rule (if required)	BR-2
Dependencies	FR-17.1
Priority	Medium

2.3.18 Add Blood Sugar

2.3.18.1 Tracker.BloodSugar.Add

Identifier	**FR-2.3.18.1**
Title	Tracker.BloodSugar.Add
Requirement	The system shall be able to add new blood sugar reading.
Source	Fatima Shahzad
Rationale	The user may want to keep the record of its blood sugar concentration.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.18.2 Tracker.BloodSugar.Concentration.Add

Identifier	**FR-2.3.18.2**
Title	Tracker.BloodSugar.Concentration.Add
Requirement	The system shall allow the user to add blood sugar concentration for the blood sugar instance.
Source	Fatima Shahzad
Rationale	The user’s blood sugar may deviate.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1
Priority	High

2.3.18.3 Tracker.BloodSugar.Concentration.Unit.Add

Identifier	**FR-2.3.18.3**
Title	Tracker.BloodSugar.Concentration.Unit.Add
Requirement	The system shall allow the user to select blood sugar concentration unit for the blood sugar instance.
Source	Fatima Shahzad

Rationale	The user blood sugar unit may mg/dL or mmol/L.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1, FR-19.2
Priority	High

2.3.18.4 Tracker.BloodSugar.Event.Add

Identifier	**FR-2.3.18.4**
Title	Tracker.BloodSugar.Event.Add
Requirement	The system shall allow the user to select the event when the blood sugar was measured.
Source	Fatima Shahzad
Rationale	The user blood sugar may be measured before or after the breakfast, meal or exercise. Its record is necessary to drive blood sugar management.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1, FR-19.2
Priority	High

2.3.18.5 Tracker.BloodSugar.Time.Add

Identifier	**FR-2.3.18.5**
Title	Tracker.BloodSugar.Time.Add
Requirement	The system shall allow the user to select the time of the day when the blood sugar was measured.
Source	Fatima Shahzad
Rationale	The user blood sugar may be measured at different times of the day.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1, FR-19.2
Priority	High

2.3.18.6 Tracker.Category.Instance.Note.Add

Identifier	**FR-2.3.18.6**
Title	Tracker.Category.Instance.Note.Add
Requirement	The system shall allow the user to write a short note about the new blood sugar instance of selected category.
Source	Fatima Shahzad
Rationale	The user may want to write any details about the instance which it might forget afterwards.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1, FR-19.2
Priority	High

2.3.18.7 Tracker.BloodSugar.Low

Identifier	**FR-2.3.18.7**
Title	Tracker.BloodSugar.Low
Requirement	If the blood sugar concentration of the user is low(hypoglycemia) the system shall be able to suggest remedies to increase blood sugar concentration in the body
Source	Fatima Shahzad
Rationale	The user may not have an experience to deal with hypoglycemia.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1, FR-19.2
Priority	Low

2.3.18.8 Tracker.BloodSugar.High

Identifier	**FR-2.3.18.8**
Title	Tracker.BloodSugar.High
Requirement	If the blood sugar concentration of the user is high(hyperglycemia) the system shall be able to suggest remedies to lower blood sugar concentration in the body.
Source	Fatima Shahzad

Rationale	The user may not have an experience to deal with hyperglycemia.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2, FR-19.1, FR-19.2
Priority	High

2.3.18.9 Message.Stored.Success

Identifier	**FR-2.3.18.9**
Title	Message.Stored.Success
Requirement	The system shall display a message that “Data successfully stored”.
Source	Fatima Shahzad
Rationale	The user may want to know whether their entered information for the instance was stored or not.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-4.9
Priority	Medium

2.3.18.10 Message.Stored.Fail

Identifier	**FR-2.3.18.10**
Title	Message.Stored.Fail
Requirement	The system shall display a message that “Data failed to store”.
Source	Fatima Shahzad
Rationale	The user may want to know whether their entered information for the instance was stored successfully or not.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-4.9
Priority	Medium

2.3.19 Update Blood Sugar

2.3.19.1 Tracker.BloodSugar.Instance.Select

Identifier	**FR-2.3.19.1**
Title	Tracker.BloodSugar.Instance.Select
Requirement	The system shall allow the user to select an instance of blood sugar record.
Source	Fatima Shahzad
Rationale	The user may want to view blood sugar instance details.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.19.2 Tracker.BloodSugar.Instance.Update

Identifier	**FR-2.3.19.2**
Title	Tracker.BloodSugar.Instance.Update
Requirement	The system shall allow the user to update the selected blood sugar instance details.
Source	Fatima Shahzad
Rationale	The user may want to make any changes in the details he previously entered for the instance.
Business Rule (if required)	BR-2, BR-6
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.20 Add Blood Pressure

2.3.20.1 Tracker.BloodSugar.Add

Identifier	**FR-2.3.20.1**
Title	Tracker.BloodPressure.Add
Requirement	The system shall be able to add new blood pressure instance.
Source	Fatima Shahzad

Rationale	The user may want to keep the record of its blood pressure concentration.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.20.2 Tracker.BloodSugar.Systolic.Add

Identifier	**FR-2.3.20.2**
Title	Tracker.BloodPressure.Systolic.Add
Requirement	The system shall allow the user to add systolic blood pressure reading.
Source	Fatima Shahzad
Rationale	The user’s blood pressure may deviate.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-17.1, FR-17.2, FR-21.1
Priority	High

2.3.20.3 Tracker.BloodSugar.Diastolic.Add

Identifier	**FR-2.3.20.3**
Title	Tracker.BloodPressure.Diastolic.Add
Requirement	The system shall allow the user to add diastolic blood pressure reading.
Source	Fatima Shahzad
Rationale	The user’s blood pressure readings may deviate.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-16.1, FR-16.2, FR-19.1
Priority	High

2.3.20.4 Tracker.BloodSugar.Time.Add

Identifier	**FR-2.3.20.4**
Title	Tracker.BloodPressure.Time.Add
Requirement	The system shall allow the user to select the time of the day when the blood pressure was measured.
Source	Fatima Shahzad
Rationale	Blood pressure could have been measured at any time of the day.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-17.1, FR-17.2, FR-21.1
Priority	High

2.3.20.5 Tracker.BloodPressure.High

Identifier	**FR-2.3.20.5**
Title	Tracker.BloodPressure.High
Requirement	If the blood pressure of the user remains above hypertension stage and stage 2 for few days then the system shall be able to display the warning showing hypertension crisis.
Source	Fatima Shahzad
Rationale	For hypertension crisis the user needs to consult a doctor.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-17.1, FR-17.2, FR-21.1, FR-21.2, FR-21.3
Priority	High

2.3.21 Update Blood Pressure

2.3.21.1 Tracker.BloodPressure.Instance.Select

Identifier	**FR-2.3.21.1**
Title	Tracker.BloodPressure.Instance.Select
Requirement	The system shall allow the user to select an instance of blood pressure record.
Source	Fatima Shahzad
Rationale	The user may want to view blood pressure instance details.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.21.2 Tracker.BloodPressure.Instance.Update

Identifier	**FR-2.3.21.2**
Title	Tracker. BloodPressure.Instance.Update
Requirement	The system shall allow the user to update the selected blood pressure instance details.
Source	Fatima Shahzad
Rationale	The user may want to make any changes in the details of the selected blood pressure instance.
Business Rule (if required)	BR-2, BR-7
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.22 Add Cholesterol

2.3.22.1 Tracker.Cholesterol.Add

Identifier	**FR-2.3.22.1**
Title	Tracker.Cholesterol.Add
Requirement	The system shall be able to add new cholesterol instance.

Source	Fatima Shahzad
Rationale	The user may want to keep the record of its cholesterol values.
Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.22.2 Tracker.Cholesterol.Add

Identifier	**FR-2.3.22.2**
Title	Tracker.Cholesterol.Value.Add
Requirement	The system shall allow the user to add Cholesterol value for the newly created instance.
Source	Fatima Shahzad
Rationale	The user's cholesterol level may deviate.
Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2, FR-22.1
Priority	High

2.3.22.3 Tracker.Cholesterol.LDL.ADD

Identifier	**FR-2.3.22.3**
Title	Tracker.Cholesterol.LDL.Add
Requirement	The system shall allow the user to add LDL(bad) Cholesterol value.
Source	Fatima Shahzad
Rationale	The user may want to keep the record of its LDL cholesterol values.
Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2, FR-22.1
Priority	High

2.3.22.4 Tracker.Cholesterol.HDL.Add

Identifier	**FR-2.3.22.4**
Title	Tracker.Cholesterol.HDL.Add
Requirement	The system shall allow the user to add HDL (good) Cholesterol value.
Source	Fatima Shahzad
Rationale	The user may want to keep the record of its HDL cholesterol values.
Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2, FR-23.1
Priority	High

2.3.22.5 Tracker.Cholesterol.Triglycerides.Add

Identifier	**FR-2.3.22.5**
Title	Tracker.Cholesterol.Triglycerides.Add
Requirement	The system shall allow the user to add triglyceride value.
Source	Fatima Shahzad
Rationale	The user may want to keep the record of its triglycerides values.
Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2, FR-23.1
Priority	High

2.3.23 Update Cholesterol

2.3.23.1 Tracker.Cholesterol.Instance.Select

Identifier	**FR-2.3.23.1**
Title	Tracker.Cholesterol.Instance.Select
Requirement	The system shall allow the user to select an instance of cholesterol record.
Source	Fatima Shahzad
Rationale	The user may want to view cholesterol instance details.

Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.23.2 Tracker.Cholesterol.Instance.Update

Identifier	**FR-2.3.23.2**
Title	Tracker. Cholesterol.Instance.Update
Requirement	The system shall allow the user to update the selected cholesterol instance details.
Source	Fatima Shahzad
Rationale	The user may want to make any changes in the details of the selected cholesterol instance.
Business Rule (if required)	BR-2, BR-5
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24 View Statistics

2.3.24.1 Tracker.Category.Select.Statistics.View

Identifier	**FR-2.3.24.1**
Title	Tracker.Category.Select.Statistics.View
Requirement	The system shall allow the user to view the statistics of the record stored in their selected category.
Source	Fatima Shahzad
Rationale	The user may want to analyze the statistics of their health factors.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24.2 Tracker.Category.Select.Statistics.Today

Identifier	**FR-2.3.24.2**
Title	Tracker.Category.Select.Statistics.Today
Requirement	The system shall allow the user to view the statistics of a particular health factors category for today.
Source	Fatima Shahzad
Rationale	The user may want to analyze the selected category statistics stored today.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24.3 Tracker.Category.Select.Statistics.Display

Identifier	**FR-2.3.24.3**
Title	Tracker.Category.Select.Statistics.Display
Requirement	The system shall be able to display the statistics of the selected category to the user.
Source	Fatima Shahzad
Rationale	The user may want to analyze the statistics of their health factors.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24.4 Tracker.Category.Select.Statistics.Week

Identifier	**FR-2.3.24.4**
Title	Tracker.Category.Select.Statistics.Week
Requirement	The system shall allow the user to view the statistics of the selected health factor category for a week.
Source	Fatima Shahzad

Rationale	The user may want to analyze the health factors statistics for the selected category stored in a week.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24.5 Tracker.Category.Select.Statistics.Month

Identifier	**FR-2.3.24.4**
Title	Tracker.Category.Select.Statistics.Month
Requirement	The system shall allow the user to view the statistics of their selected health factor for a month.
Source	Fatima Shahzad
Rationale	The user may want to analyze the particular health factor statistics stored in a month.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24.6 Tracker.Category.Select.Statistics.HalfYear

Identifier	**FR-2.3.24.6**
Title	Tracker.Category.Select.Statistics.HalfYear
Requirement	The system shall allow the user to view the statistics of their selected health factor for 6 months.
Source	Fatima Shahzad
Rationale	The user may want to analyze the health factor statistics stored in a half year.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.24.7 Tracker.Category.Select.Statistics.Year

Identifier	**FR-2.3.24.7**
Title	Tracker.Category.Select.Statistics.Year
Requirement	The system shall allow the user to view the statistics of their selected health factor category for year.
Source	Fatima Shahzad
Rationale	The user may want to analyze the selected health factor statistics stored in a year.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	High

2.3.25 View All Factors Statistics

2.3.25.1 Tracker.Combined.Statistics

Identifier	**FR-2.3.25.1**
Title	Tracker.Combined.Statistics
Requirement	The system shall allow the user to view the combined statistics for all the available health factors.
Source	Fatima Shahzad
Rationale	The user may want to analyze the selected health factor statistics stored in a year.
Business Rule (if required)	BR-1, BR-2, BR-3
Dependencies	FR-17.1, FR-17.2
Priority	Medium

2.3.26 Add meal

2.3.26.1 Meal.Add.Manually

Identifier	**FR-2.3.26.1**
Title	Meal.Add.Manually
Requirement	If patient consumes meal not suggested by diet chart, the patient shall be able to add meal
Source	Kashaf Fatima
Rationale	The calories of every meal should be saved to keep track of patients diets and to set his diet plan, exercise, and medicine dosage accordingly.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.26.2 Meal.Add.Images

Identifier	**FR-2.3.26.2**
Title	Meal.Add.Images
Requirement	The patient shall be able to add meal by taking images of the meal.
Source	Kashaf Fatima
Rationale	Patient may find it difficult to add meals on its own so taking images of the food application should be able to calculate calories and save them.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.26.3 Meal.Add.Images.PrectName.Right

Identifier	**FR-2.3.26.3**
Title	Meal.Add.Images.PredictName.Right
Requirement	If patient add meal through taking image the application shall be able to predict the name of the food in the image.
Source	Kashaf Fatima
Rationale	Patients may find it difficult to add meals on their own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.4 Meal.Add.Images.PrectName.Right.Select

Identifier	**FR-2.3.26.4**
Title	Meal.Add.Images.PredictName.Right.Select
Requirement	The patient shall be able to select the name of the dish from names predicted by the application and confirm it.
Source	Kashaf Fatima
Rationale	Patients may find it difficult to add meals on their own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.5 Meal.Add.Images.PrectName.Wrong

Identifier	**FR-2.3.26.5**
Title	Meal.Add.Images.PredictName.Wrong
Requirement	If the predicted names of dishes by the application are wrong, the application shall be able to prompt the user to add meal details manually.
Source	Kashaf Fatima
Rationale	Patients may find it difficult to add meals on their own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.6 Meal.Add.Images.PrectName.Right.PredictInformation

Identifier	**FR- 2.3.26.6**
Title	Meal.Add.Images.PredictName.Right.PredictInformation
Requirement	If the patient selects the right name of the dish from the predicted names, the system shall be able to predict calories, carbohydrates, fibers and protein of the meal.
Source	Kashaf Fatima
Rationale	Patients may find it difficult to add meals on their own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.7 Meal.Add.Images.PredictName.Right.PredictInformation.Confirm

Identifier	**FR- 2.3.26.7**
Title	Meal.Add.Images.PredictName.Right.PredictInformation.Confirm
Requirement	After predicting the information about the meal, the system shall be able to ask the patient to confirm it.
Source	Kashaf Fatima
Rationale	Patients may find it difficult to add meals on their own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.8 Meal.Add.Images.PredictName.Right.PredictInformation.Confirm.Right

Identifier	**FR- 2.3.26.8**
Title	Meal.Add.Images.PredictName.Right.PredictInformation.Confirm.Right
Requirement	If the patient indicates on predicted information about the meal is right the system shall be able to save the meal into the database successfully and show status message indicating meal saved.
Source	Kashaf Fatima
Rationale	Patients may find it difficult to add meals on their own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.9 Meal.Add.Images.PredictName.Right.PredictInformation.Confirm.Wrong

Identifier	**FR- 2.3.26.9**
Title	Meal.Add.Images.PredictName.Right.PredictInformation.Confirm.Wrong
Requirement	If the patient indicates on predicted information regarding the meal is wrong the system shall allow the user to change it.
Source	Kashaf Fatima
Rationale	Patient may find it difficult to add meals on its own so taking images of the food application should be able to calculate calories and save them.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.10 Meal.Add.Information

Identifier	**FR- 2.3.26.10**
Title	Meal.Add.Information
Requirement	Patient shall be able to add dishes name, total no of calories in it, carbohydrates, fiber, and proteins.
Source	Kashaf Fatima
Rationale	A patient's diet plan is made according to his information so a change in his meal consumption will require to update diet plan.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.26.11 Meal.Add.MealType

Identifier	**FR- 2.3.26.11**
Title	Meal.Add.MealType
Requirement	The system shall be able to ask the patient to save meal type (breakfast, lunch, dinner, snack) while saving meal not present in diet plan.
Source	Kashaf Fatima
Rationale	The patient's meal type is needed to update his diet plan according to it.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.26.12 Meal.Add.Time.Manually

Identifier	**FR- 2.3.26.12**
Title	Meal.Add.Time.Manually
Requirement	When the patient triggers to add calories, the application shall be able to ask the patient to enter time of consumption of meal.
Source	Kashaf Fatima
Rationale	Timing of the patient's meal consumption should bet saved in order to keep his blood sugar on track.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.26.13 Meal.Add.Time.Clock

Identifier	**FR- 2.3.26.13**
Title	Meal.Add.Time.Clock
Requirement	When the patient triggers to add calories, the application shall be able to add time of consumption of meal through clock by setting up its needles.

Source	Kashaf Fatima
Rationale	Patient may find it difficult to add time manually and find it easy to just set clock needles.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.26.14 Meal.Add.Time.Ahead.Confirm

Identifier	**FR- 2.3.26.14**
Title	Meal.Add.Time.Ahead.Confirm
Requirement	If the patient tries to enter time ahead of current time the application shall be able to prompt user to confirm.
Source	Kashaf Fatima
Rationale	Application should make sure if patient has not mistakenly set the time.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.26.15 Meal.Add.Time.Ahead.Confirm.Ok

Identifier	**FR- 2.3.26.15**
Title	Meal.Add.Time.Ahead.Confirm.Ok
Requirement	If the patient selects ok for ahead time message the application shall be able to continue the add calories process for patient.
Source	Kashaf Fatima
Rationale	Patients should be given authority to save the meal before eating too if they want to.

Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.26.16 Meal.Add.Time.Ahead.Confirm.Cancel

Identifier	**FR-2.3.26.16**
Title	Meal.Add.Time.Ahead.Confirm.Cancel
Requirement	If patient selects cancel for ahead time message the application shall be able to ask the user to re-enter time.
Source	Kashaf Fatima
Rationale	Patients should be asked to re-enter time to avoid any mistake and to keep his information authentic.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.27 View diet chart

2.3.27.1 DietChart.View

Identifier	**FR-2.3.27.1**
Title	DietChart.View
Requirement	The patient shall be able to view diet chart to check the information regarding his past meals of an year.
Source	Kashaf Fatima
Rationale	The patient should be allowed to view diet chart, containing information about his past meals.
Business Rule (if required)	

Dependencies	
Priority	Medium

2.3.27.2 DietChart.View.Today

Identifier	**FR-2.3.27.2**
Title	DietChart.View.Today
Requirement	The system shall allow the patient to view saved meals information of the current day.
Source	Kashaf Fatima
Rationale	The patient should be allowed to view diet chart, containing information about his past meals.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.27.3 DietChart.View.Empty

Identifier	**FR-2.3.27.3**
Title	DietChart.View.Empty
Requirement	If the patient triggers to display diet chart without storing any meal info the system shall be able to display empty diet chart.
Source	Kashaf Fatima
Rationale	Diet chart is the information about previous meal if patient has not recorded any meal, yet it should be empty
Business (if Rule required)	
Dependencies	
Priority	High

2.3.28 View today's meal plan

2.3.28.1 TodayMealPlan.View

Identifier	**FR-2.3.28.1**
Title	TodayMealPlan.View
Requirement	The patient shall be able to view meal plan containing the information about all his meals for the entire day.
Source	Kashaf Fatima
Rationale	Patient meal plan should be calculated according to his details.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.28.2 TodayMealPlan.View.Select

Identifier	**FR-2.3.28.2**
Title	TodayMealPlan.View.Select
Requirement	The patient shall be able to select one type of meal (Breakfast, snack, lunch, dinner) and view it.
Source	Kashaf Fatima
Rationale	Patients should be able to select and view a single meal type from the entire day’s meal plan.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.28.3 TodayMealPlan.View.Select.Breakfast

Identifier	**FR-2.3.28.3**
Title	TodayMealPlan.View.Select.Breakfast
Requirement	If the patient selects breakfast from the meal plan, the application shall be able to display all the dishes suggested for breakfast.
Source	Kashaf Fatima
Rationale	Patients should be able to view breakfast only and view its dishes.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.28.4 TodayMealPlan.View.Select.Snack

Identifier	**FR-2.3.28.4**
Title	TodayMealPlan.View.Select.Snack
Requirement	If the patient selects Snack from the meal plan, the application shall be able to display all the dishes suggested for Snacks.
Source	Kashaf Fatima
Rationale	Patients should be able to view snack only and view its dishes.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.28.5 TodayMealPlan.View.Select.Lunch

Identifier	**FR-2.3.28.5**
Title	TodayMealPlan.View.Select.Lunch
Requirement	If the patient selects Lunch from the meal plan, the application shall be able to display all the dishes suggested for Lunch.
Source	Kashaf Fatima

Rationale	Patients should be able to view Lunch only and view its dishes.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.28.6 TodayMealPlan.View.Select.Dinner

Identifier	**FR-2.3.28.6**
Title	TodayMealPlan.View.Select.Dinner
Requirement	If the patient selects dinner from the meal plan, the application shall be able to display all the dishes suggested for dinner.
Source	Kashaf Fatima
Rationale	Patients should be able to view dinner only and view its dishes.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.28.7 TodayMealPlan.View.NoInformation

Identifier	**FR-2.3.28.7**
Title	TodayMealPlan.View.NoInformation
Requirement	If the patient has not given his information yet the application shall be able to prompt the patient to enter his details to view his meal plan.
Source	Kashaf Fatima
Rationale	Patient meal plan should be calculated according to his details so if he has not yet given his information the meal plan should not open.
Business (if Rule required)	

Dependencies	
Priority	High

2.3.29 View recipe

2.3.29.1 Recipe.Select

Identifier	**FR-2.3.29.1**
Title	Recipe.Select
Requirement	The patient shall be able to select the recipe to be views
Source	Kashaf Fatima
Rationale	Patient should be able select a single recipe incase he wants to view it
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.29.2 Recipe.Select.View

Identifier	**FR-2.3.29.2**
Title	Recipe.Select.View
Requirement	The patient shall be able to view recipes of the selected dishes specified in the meal plan.
Source	Kashaf Fatima
Rationale	Patients should be allowed to check recipes of the dishes to view their ingredients along with their calories information.
Business (if Rule required)	

Dependencies	
Priority	Low

2.3.30 Save meal from meal plan

2.3.30.1 Meal.ConsumedFromMealPlan.Save

Identifier	**FR-2.3.30.1**
Title	Meal.ConsumedFromMealPlan.Save
Requirement	If the patient has consumed one of the meals from the meal plan of current day, the patient shall be able to save that meal into the diet chart
Source	Kashaf Fatima
Rationale	Patient diet chart should be maintained to keep track of his meals.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.30.2 Meal.ConsumedFromMealPlan.Save.Time

Identifier	**FR-2.3.30.2**
Title	Meal.ConsumedFromMealPlan.Save.Time
Requirement	If patient indicates on saving meal from the meal plan, the application shall be able to prompt the user to save its consumption time along with it.
Source	Kashaf Fatima
Rationale	Meal’s consumption time should be saved along with the meal to make adjusted for the next meals.
Business (if Rule required)	

Dependencies	
Priority	Low

2.3.31 View BMI

2.3.31.1 BMI.View

Identifier	**FR-2.3.31.1**
Title	BMI.View
Requirement	The patient shall be able to view his calculated BMI
Source	Kashaf Fatima
Rationale	Patient should be able to check his BMI calculated by application by using his information.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.31.2 BMI.View.NoInformation

Identifier	**FR-2.3.31.2**
Title	BMI.View.NoInformation
Requirement	If the patient triggers to display his BMI without providing his information the application shall be able to prompt the patient to enter his information to calculate it.
Source	Kashaf Fatima
Rationale	BMI shall be calculated using the length and weight of the patient to patient must add his information first before asking application to display his BMI.
Business (if Rule required)	

Dependencies	
Priority	High

2.3.32 View exercise routine

2.3.32.1 ExcerciseRoutine.Start

Identifier	**FR-2.3.32.2**
Title	ExcerciseRoutine.Start
Requirement	The patient shall be able to start his exercise routine of the current day.
Source	Kashaf Fatima
Rationale	The time of the exercise will be measured according to the times the patient takes his meals. The patient will be able to start the exercise on the specified time.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.32.2 ExcerciseRoutine.Pause

Identifier	**FR-2.3.32.2**
Title	ExcerciseRoutine.Pause
Requirement	If the exercise routine has started the patient shall be able to pause it.
Source	Kashaf Fatima
Rationale	Patient may feel shaky, or weak during exercise indicating their blood sugar is dropping. At that time they will have to stop exercising, for this a pause is needed.
Business Rule (if required)	

Dependencies	
Priority	Medium

2.3.32.3 ExcerciseRoutine.Resume

Identifier	**FR-2.3.32.3**
Title	ExcerciseRoutine.Resume
Requirement	After pausing the exercise if the patient wants to continue working out, the application shall be able to allow the patient to resume the started exercise routine.
Source	Kashaf Fatima
Rationale	After a patient may find that their blood sugar is in range of 100 to 250 mg/dl, he may want to continue his exercise.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.32.4 ExcerciseRoutine.Stop

Identifier	**FR-2.3.32.4**
Title	ExcerciseRoutine.Stop
Requirement	After starting the exercise, if the patient wants to stop it the application shall be able to allow the patient to stop the exercise and suggest alternate walking suggestion.
Source	Kashaf Fatima
Rationale	Patients may sometimes find it hard to workout due to exhaustion or any other problem. For that patient may have the liberty to stop the exercise and convert to any easy type of physical activity.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.32.5 ExcerciseRoutine.Missed

Identifier	**FR-2.3.32.5**
Title	ExcerciseRoutine.Missed
Requirement	If the exercise routine is missed by a patient, the application shall be able to mark that day missed in the exercise history.
Source	Kashaf Fatima
Rationale	Patients may miss exercise due to any reason the system should record missed exercise in exercise history to make future exercise adjustments.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.32.6 ExcerciseRoutine.Missed.Update

Identifier	**FR-2.3.32.6**
Title	ExcerciseRoutine.Missed.Update
Requirement	If the exercise routine is missed by a patient, the application shall be able to update the remaining exercise of the week according to patient medical information.
Source	Kashaf Fatima
Rationale	Application should maintain patients' health still if the patient misses any exercise and make adjustment in his further exercise plan.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.32.7 ExcerciseRoutine.NoInformation

Identifier	**FR-2.3.32.7**
Title	ExcerciseRoutine.NoInformation

Requirement	If patient indications on starting the exercise routine without providing his information the application shall be able to prompt the patient to enter his information to make his exercise plan.
Source	Kashaf Fatima
Rationale	Patient physical activity level may change with his details, so it is mandatory for patient to provide his details before making his exercise plan.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.33 View exercise history

2.3.33.1 Exercise.History.View

Identifier	**FR-2.3.33.1**
Title	Exercise.History.View
Requirement	The patient shall be able to view his exercise history.
Source	Kashaf Fatima
Rationale	Patient may want to see when he exercised and when he skipped it for that a history must be maintained which will furthered be used to make changes in further exercises.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.34 View diabetes Information

2.3.34.1 DiabetesInformation.View

Identifier	**FR-2.3.34.1**
Title	DiabetesInformation.View
Requirement	If the patient indicates on displaying diabetes information, the application shall be able to display diabetes related information categorized in to different categories.
Source	Kashaf Fatima
Rationale	Educating patient may encourage them to put more effort in managing their lifestyle and keep them update.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.34.2 DiabetesInformation.View.Blog

Identifier	**FR-2.3.34.2**
Title	DiabetesInformation.View.Blog
Requirement	If the patient indicates on displaying blog from diabetes information, the application shall be able to display all the blog from authentic websites listed on screen.
Source	Kashaf Fatima
Rationale	Patients may want to educate themselves by reading different blogs
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.34.3 DiabetesInformation.View.Videos

Identifier	**FR-2.3.34.3**
Title	DiabetesInformation.View.Videos
Requirement	If the patient indicates on displaying videos from diabetes information, the application shall be able to display all the videos from authentic websites listed on screen.
Source	Kashaf Fatima
Rationale	Patient may want to educate themselves by watching diabetes related videos.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.34.4 DiabetesInformation.View.Articles

Identifier	**FR-2.3.34.4**
Title	DiabetesInformation.View.Articles
Requirement	If the patient indicates on displaying articles from diabetes information, the application shall be able to display all the articles from authentic websites listed on screen.
Source	Kashaf Fatima
Rationale	Patients may want to educate themselves by reading diabetes related articles.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.34.5 DiabetesInformation.View.News

Identifier	**FR-2.3.34.5**
Title	DiabetesInformation.View.News
Requirement	If the patient indicates on displaying news from diabetes information, the application shall be able to display all the authentic news related to diabetes
Source	Kashaf Fatima
Rationale	Patients may want to keep them up to date about diabetes related recent developments and other news.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.35 Participate in live chats

2.3.35.1 LiveChat.Send.Message.Text

Identifier	**FR-2.3.35.1**
Title	LiveChat.Send.Message.Text
Requirement	The patient shall be able to send text messages in the live chat.
Source	Kashaf Fatima
Rationale	Patients may want to interact with other patients to keep him motivated.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.35.2 LiveChat.Send.Message.Image

Identifier	**FR-2.3.35.2**
Title	LiveChat.Send.Message.Image
Requirement	The patient shall be able to send messages in the live chat.
Source	Kashaf Fatima
Rationale	Patients may want to interact using pictures.
Business (if Rule required)	
Dependencies	
Priority	low

2.3.35.3 LiveChat.Send.Message.ImageAndText

Identifier	**FR-2.3.35.1**
Title	LiveChat.Send.Message.ImageAndText
Requirement	The patient shall be able to send text messages along with images in the live chat.
Source	Kashaf Fatima
Rationale	Patients may want to interact with text and images.
Business (if Rule required)	
Dependencies	
Priority	low

2.3.36 Delete Account

2.3.36.1 Profile.View

Identifier	**FR-2.3.36.1**
Title	Profile.View
Requirement	The patient shall be able to select his profile to view it.
Source	Kashaf Fatima
Rationale	Patient may want to view his profile to make changes
Business Rule (if required)	
Dependencies	
Priority	low

2.3.36.2 Profile.View.Delete

Identifier	**FR-2.3.36.2**
Title	Profile.View.Delete
Requirement	The patient shall be able to delete his account.
Source	Kashaf Fatima
Rationale	Patient may want to delete his account.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.36.3 Profile.View.Delete.Confirm

Identifier	**FR-2.3.36.3**
Title	Profile.View.Delete.Confirm
Requirement	If the patient indicates on deleting the account, the application shall be able to confirm it from the patient.
Source	Kashaf Fatima
Rationale	Patient may really not want to delete his account application must confirm before deleting it.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.36.4 Profile.View.Delete.Confirm.Ok

Identifier	**FR-2.3.36.4**
Title	Profile.View.Delete.Confirm.Ok
Requirement	If the patient confirms deletion the application shall be able to remove the patient from registered patients list and delete his account.
Source	Kashaf Fatima
Rationale	Patient may not want to keep his account anymore and may want to delete.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.36.5 Profile.View.Delete.Confirm.Cancel

Identifier	**FR-2.3.36.5**
Title	Profile.View.Delete.Confirm.Cancel
Requirement	If the patient cancels deletion the application shall be able to stop the deleting account process and return to patients profile
Source	Kashaf Fatima
Rationale	Patient may accidently trigger delete profile.
Business (if Rule required)	
Dependencies	
Priority	low

2.3.37 Edit Account

2.3.37.1 Profile.View

Identifier	**FR-2.3.37.1**
Title	Profile.View
Requirement	The patient shall be able to view his profile to make changes.
Source	Kashaf Fatima
Rationale	Patient may want to just view or edit his profile.
Business (if Rule required)	
Dependencies	
Priority	low

2.3.37.2 Profile.View.Select

Identifier	**FR-2.3.37.2**
Title	Profile.View.Select
Requirement	The patient shall be able to select from category of the information to edit.
Source	Kashaf Fatima
Rationale	Patient info may change with time so he should be able to edit his profile.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.37.3 Profile.View.Select.Password

Identifier	**FR-2.3.37.3**
Title	Profile.View.Select.Password
Requirement	If the patient selects password from profile and indicates on changing it the application shall be able to change it.
Source	Kashaf Fatima
Rationale	Patient may want to change his password
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.37.4 Profile.View.Select.Password.Enter.Previous

Identifier	**FR-2.3.37.4**
Title	Profile.View.Select.Password.Enter.Previous
Requirement	If the patient indicates on changing the password the application shall be able to prompt the patient to enter his previous password to authorize him.
Source	Kashaf Fatima
Rationale	Patient must be confirmed before changing his password.
Business (if Rule required)	
Dependencies	
Priority	low

2.3.37.5 Profile.View.Select.Password.Enter.New

Identifier	**FR-2.3.37.5**
Title	Profile.View.Select.Password.Enter.New
Requirement	After authorizing the patient by asking patient’s previous password, the application shall be able to prompt the patient to enter his new password.
Source	Kashaf Fatima
Rationale	Patient may want to change his password.
Business (if Rule required)	
Dependencies	
Priority	low

2.3.37.6 Profile.View.Select.Password.Enter.New.ReEnter

Identifier	**FR-2.3.37.6**
Title	Profile.View.Select.Password.Enter.New.ReEnter
Requirement	The patient shall be able to re-enter the password to confirm it.
Source	Kashaf Fatima
Rationale	Patient may want to change his password.
Business Rule (if required)	
Dependencies	
Priority	low

2.3.37.7 Profile.View.Select.Weight

Identifier	**FR-2.3.37.7**
Title	Profile.View.Select.Weight
Requirement	If the patient selects weight from profile and indicates on changing it the application shall be able to change it.
Source	Kashaf Fatima
Rationale	Patient weight may change with time.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.37.8 Profile.View.Select.Weight.Enter

Identifier	**FR-2.3.37.8**
Title	Profile.View.Select.Weight.Enter
Requirement	The patient shall be able to enter his weight to update it.
Source	Kashaf Fatima
Rationale	Patient weight may change with time.

Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.37.9 Profile.View.Select.Height

Identifier	**FR-2.3.37.9**
Title	Profile.View.Select.Height
Requirement	If the patient selects height from profile and indicates on changing it the application shall be able to change it.
Source	Kashaf Fatima
Rationale	Patient height may change with time.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.37.10 Profile.View.Select.Height.Enter

Identifier	**FR-2.3.37.10**
Title	Profile.View.Select.Height.Enter
Requirement	The patient shall be able to enter his height to update it.
Source	Kashaf Fatima
Rationale	Patient height may change with time.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.37.11 Profile.View.Select.Picture

Identifier	**FR-2.3.37.11**
Title	Profile.View.Select.Picture
Requirement	If the patient selects Profile picture from profile and indicates on changing it the application shall be able to change it.
Source	Kashaf Fatima
Rationale	Patient may like to change his profile picture.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.37.12 Profile.View.Select.Picture.Select

Identifier	**FR-2.3.37.12**
Title	Profile.View.Select.Picture.Select
Requirement	If the patient indicates on changing the profile picture the application shall be able to allow the patient to select a picture from his device.
Source	Kashaf Fatima
Rationale	Patient weight may change with time.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.38 Post on the discussion board

2.3.38.1 DiscussionBoard.View

Identifier	**FR-2.3.38.1**
Title	DiscussionBoard.View
Requirement	If the patient indicates towards discussion board the application shall be able display discussion board with most recent queries at the top.
Source	Kashaf Fatima
Rationale	Patients may want to find an answer to a query related to diabetes.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.2 DiscussionBoard.Post.New.Enter

Identifier	**FR-2.3.38.2**
Title	DiscussionBoard.Post.New
Requirement	If the patient indicates on initiating a new query, the application shall be able to allow patient to enter query.
Source	Kashaf Fatima
Rationale	In case a patient does not find a query related to his query he should post his own to get answers.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.3 DiscussionBoard.Post.New.Post

Identifier	**FR-2.3.38.3**
Title	DiscussionBoard.Post.New.Post
Requirement	If patient indicates on posting the newly written query, the application shall be able to post it so other patient can read it.
Source	Kashaf Fatima
Rationale	Patients may want to find an answer to a query related to diabetes.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.4 DiscussionBoard.Post.New.Post.Visible

Identifier	**FR-2.3.38.4**
Title	DiscussionBoard.Post.New.Post.Visible
Requirement	After the patient indicates on posting the newly written query, the application shall be able to display that query to all other patient's discussion board.
Source	Kashaf Fatima
Rationale	Patients may want to help other patients by answering them.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.5 DiscussionBoard.Query.Select

Identifier	**FR-2.3.38.5**
Title	DiscussionBoard.Query.Select
Requirement	The patient shall be able to select a particular query from the list of queries in their discussion board.
Source	Kashaf Fatima
Rationale	Patients may find a query interesting and may want to open it.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.6 DiscussionBoard.Query.Select.View

Identifier	**FR-2.3.38.6**
Title	DiscussionBoard.Query.Select.View
Requirement	The patient shall be able to view a particular query from their discussion board.
Source	Kashaf Fatima
Rationale	Patients may find a query interesting and may want to open it.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.7 DiscussionBoard.Query.Select.View.Reply.Enter

Identifier	**FR-2.3.38.7**
Title	DiscussionBoard.Query.Select.View.Reply.Enter
Requirement	If the patient indicates on replying to a query, the application shall be able to allow the patient to enter the reply.
Source	Kashaf Fatima
Rationale	Patients may know the answer to a query posted by another patient and want to reply to him.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.8 DiscussionBoard.Query.Select.View.Reply.Enter.Post

Identifier	**FR-2.3.38.8**
Title	DiscussionBoard.Query.Select.View.Reply.Enter.Post
Requirement	As soon as the patient indicates on posting the reply, the application shall be able to post it so the other patient can also read it.
Source	Kashaf Fatima
Rationale	Patients reply may be helpful for another patient with similar query.
Business (if Rule required)	
Dependencies	
Priority	Low

2.3.38.9 DiscussionBoard.Query.Select.View.Reply.Enter.Post.Visible

Identifier	**FR-2.3.38.9**
Title	DiscussionBoard.Query.Select.View.Reply.Enter.Post.Visible
Requirement	After the patient indicates on posting the reply, the application shall be able to display that reply to all other patient's discussion board.
Source	Kashaf Fatima
Rationale	Patients reply may be helpful for another patient with similar query.
Business Rule (if required)	
Dependencies	
Priority	Low

2.3.39 FR's For Events

2.3.39.1 Medicine.ActiveAgents.Warning

Identifier	**FR-2.3.39.1**
Title	Medicine.ActiveAgent.Warning
Requirement	If the newly added medication by user have allergic active agent, the system shall be able to warn the user about the allergic reaction.
Source	Fatima Shahzad
Rationale	The warning will make the user aware about the allergic reaction the medicine may can cause.
Business Rule (if required)	
Dependencies	
Priority	Medium

2.3.39.2 Medicine.Probability.SideEffects

Identifier	**FR-2.3.39.2**
Title	Medicine.Probability.SideEffects
Requirement	If the newly added medication can cause any side effects, the system shall be able to alert the user about its probability.
Source	Fatima Shahzad
Rationale	The user may confuse the side effects with reaction.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.39.3 Medicine.Intake.Alarm

Identifier	**FR-2.3.39.2**
Title	Medicine.Intake.Alarm
Requirement	If the medicine intake time is reached, the system shall be able to display an alarm as reminder.
Source	Fatima Shahzad
Rationale	The user might forget about its medication intake.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.39.4 Meal.Intake.Allergic

Identifier	**FR-2.3.39.4**
Title	Meal.Intake.Allergic
Requirement	If the newly added meal may have ingredients user is allergic with, the system shall warn the user about its allergic reaction.
Source	Fatima Shahzad
Rationale	The user may not know that the meal ingredients may have its allergic food.
Business (if Rule required)	
Dependencies	
Priority	Medium

2.3.39.5 Meal.Intake.Calories.Overconsumption

Identifier	**FR-2.3.39.5**
Title	Meal.Intake.Calories.Overconsumption
Requirement	Whenever the user intakes the food with more than allowed calories, the system shall be able to update the allowed carbohydrates, diet chart and exercise plan.
Source	Fatima Shahzad
Rationale	The bolus insulin changes according to calories intake and blood-sugar peak after taking the meal.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.39.6 Meal.Intake.Sugar.Notification

Identifier	**FR-2.3.39.6**
Title	Meal.After.BolusInsulin
Requirement	Whenever the user saves the meal intake, the system shall be able to display a notification to measure the blood-sugar concentration.
Source	Fatima Shahzad
Rationale	After meal the blood-sugar peak is raised which needs to be baselined with bolus insulin.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.39.7 Sugar.Meal.After.BolusInsulin

Identifier	**FR-2.3.39.7**
Title	Meal.After.BolusInsulin
Requirement	Whenever the user enters the blood-sugar after meal intake, the system shall be able to update the bolus insulin units to baseline the blood-sugar peak.
Source	Fatima Shahzad
Rationale	The bolus insulin changes according to calories intake and blood-sugar peak after taking the meal.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.39.8 Sugar.Meal.Image.Add

Identifier	**FR-2.3.39.8**
Title	Sugar.Meal.Image.Add
Requirement	Whenever the user provides the meal image, the system shall be able to suggest names of the food
Source	Kashaf Fatima
Rationale	System shows the meal related items to make sure that the correct food is selected for calories count.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.39.9 Sugar.Meal.Image.Add.SelectName

Identifier	**FR-2.3.39.9**
Title	Sugar.Meal.Image.Add.SelectName
Requirement	As soon as the patient selects dish name suggested by the application, the application shall be able to calculate food calories from the image.
Source	Kashaf Fatima
Rationale	System will be able to calculate calories from the images of the food to make ease for patient.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.39.10 HealthFator.Add.Statistics.Update

Identifier	**FR-2.3.39.10**
Title	HealthFator.Add.Statistics.Update
Requirement	Whenever the user adds a new heath factor instance, the system shall update all the statistics according to the old and newly added record.
Source	Fatima Shahzad
Rationale	The statistics do change after adding instances to the health factors.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.39.11 HealthFator.Add.Statistics.Update

Identifier	**FR-2.3.39.11**
Title	ExerciseRoutine.Today.Initiated.Not
Requirement	If patient does not opens the todays exercise routine for the whole day, the application shall be able to mark exercise missed in the exercise history.
Source	Fatima Shahzad
Rationale	In order to update the exercise plan of the patient, application needs to check previous activity to confirm it does not becomes harmful for patient.
Business Rule (if required)	
Dependencies	
Priority	High

2.3.39.12 ExerciseRoutine.Today.Sugar.Low

Identifier	**FR-2.3.39.12**
Title	ExerciseRoutine.Today.Sugar.Low
Requirement	If patient’s blood sugar level is lower the his average the exercise routine for today will be marked cancelled.
Source	Fatima Shahzad
Rationale	More physical activity for patient during hypoglycemia can be dangerous.
Business (if Rule required)	
Dependencies	
Priority	High

2.3.39.13 ExerciseRoutine.Today.Incomplete.Sugar.High

Identifier	**FR-2.3.39.13**
Title	ExerciseRoutine.Today.Incomplete.Sugar.High
Requirement	If patient todays exercise routine was not completed and his blood sugar is also high, the application shall be able to update his diet and exercise plan to maintain his blood sugar level.
Source	Fatima Shahzad
Rationale	The diet chart with low carbohydrates, and exercise plan with more calories burn would help lower the blood sugar.
Business (if Rule required)	
Dependencies	
Priority	High

2.4 Non-Functional Requirements

The following section contains the non- functional requirements for Smart Diabetolog application.

2.4.1 Availability

AVL-1: If there is some maintenance or update, the system shall be able to make maintenance when the user is asleep after taking its bolus insulin at night

2.4.2 Installability

INS-1: An untrained user shall be able to perform initial installation of the application within an average of 5 minutes.
INS-2: When installing an upgraded version of the application, all customizations in the user's profile shall be retained and converted to the new version's data format if needed.
INS-3: The installation program shall verify the correctness of the download before beginning the installation process.

2.4.3 Performance

PER-1: If a patient indicates on saving an information, application shall be able to save user provided information in the database within 10 seconds and show status message indicating information saved.
PER-2: If a patient indicates on updating an information, application shall be able to update the information in the database within 10 seconds with status message showing information updated.
PER-3: The application shall be able to display updated changes in the system within 5 seconds of updating.
PER-4: The application shall be able to display statistics within 10 seconds.
PER-5: The application shall be able to read information from the database within 2 seconds.

2.4.4 Security

SEC-1: Only registered patients shall be able to access the system.
SEC-2: Patients shall be able to login to their accounts with passwords in order to secure their medical information from others.

2.4.5 Integrity

INT-1: The application shall confirm the input data before saving information in database by specifying proper format for input fields.
INT-2: The application shall be able to confirm that the user has provided meal images from both the top and side of the plate.

2.4.6 Interoperability

IOP-1: The system shall be able to import the recipes for diet chart from diabetes.uk, tarladalal.com and mayoclinic.org.
IOP-2: The system shall be able to import the medication ingredients from dailymed.nlm.nih.gov and medlineplus.gov.

2.4.7 Safety

SAF-1: If the user mentions the consumption of any food whose particular ingredient is mentioned in allergic reaction history, the system shall be able to show warning alarm to the user about the allergy.
SAF-2: If the user enters any new current medication, whose formula or active agent is similar to any medication present in allergic reaction history, the system shall be able to display a warning alarm with the medication details.
SAF-3: When the user adds new blood-sugar, blood-pressure or cholesterol readings, the system shall update the diet charts and exercise plans to maintain the user’s health factors at normal level.

SAF-4: In the meals plan the system shall display only those options whose calories won’t palpitate the user’s blood sugar and other health factors.

SAF-5: If the user is experiencing low sugar level in type-1 diabetes the system shall be able to cancel the exercise plan for today.

2.4.8 Robustness

ROB-1: If the user initiates to perform any action without entering the data required to perform that action, the system shall point to the missing data entries with red line with a message to fill in the data.

2.4.9 Usability

USA-1: A new patient shall be able to add his information within an average of 2 minutes and a maximum of 4 minutes, 95 percent of the time.

USA-2: A new patient shall be able to view diet and exercise plan within an average of 3 minutes and a maximum of 4 minutes, 98 percent of the time.

USA-3: A new patient shall be able to find tracker within 1 to 2 steps after reaching the landing page of the application.

USA-4: The application shall be able to avoid errors by specifying optional with optional data fields while performing any action.

USA-5: The new user shall be able to learn using the system by reading the tags on the home screen like tracker, diet and exercise within 1 or 2 minutes.

2.5 External Interface Requirements

The following are the external requirements for Smart Diabetolog.

2.5.1 User Interfaces Requirements

UI-1: The application shall allow the user to enable dark mode.

UI-2: The application shall use cool and white tones color scheme.

UI-3: All the buttons in the application shall be well meaning.

UI-4: The application shall display warnings with danger sign.

UI-5: The application shall display alerts with yellow color sign.
UI-6: Each screen is consistent with respect to color and buttons.

UI-7: Each screen has a button which directs it to main screen.

2.5.2 Software interfaces

SI-1: The application shall be able scrap diabetes information like blogs and videos from authentic websites through a programing interface.

SI-2: The application shall be able to scrap active agents of medicines from authentic websites through a programing interface.

SI-3: The application shall be able to scrap food recipes from authentic websites through a programming interface.

2.5.3 Hardware interfaces

No hardware interfaces are identified.

2.5.4 Communications interfaces

CI-1: When the user registers, the system shall be able to send the confirmation email before completing the registration process.
CI-2: If the user initiates to forget the password, the system shall be able to send an email to the registered account for its verification and password changing.

3 Design and Architecture

The following section contains the architectural design, software process model and data design of the smart diabetolog application.

3.1 Architectural Design

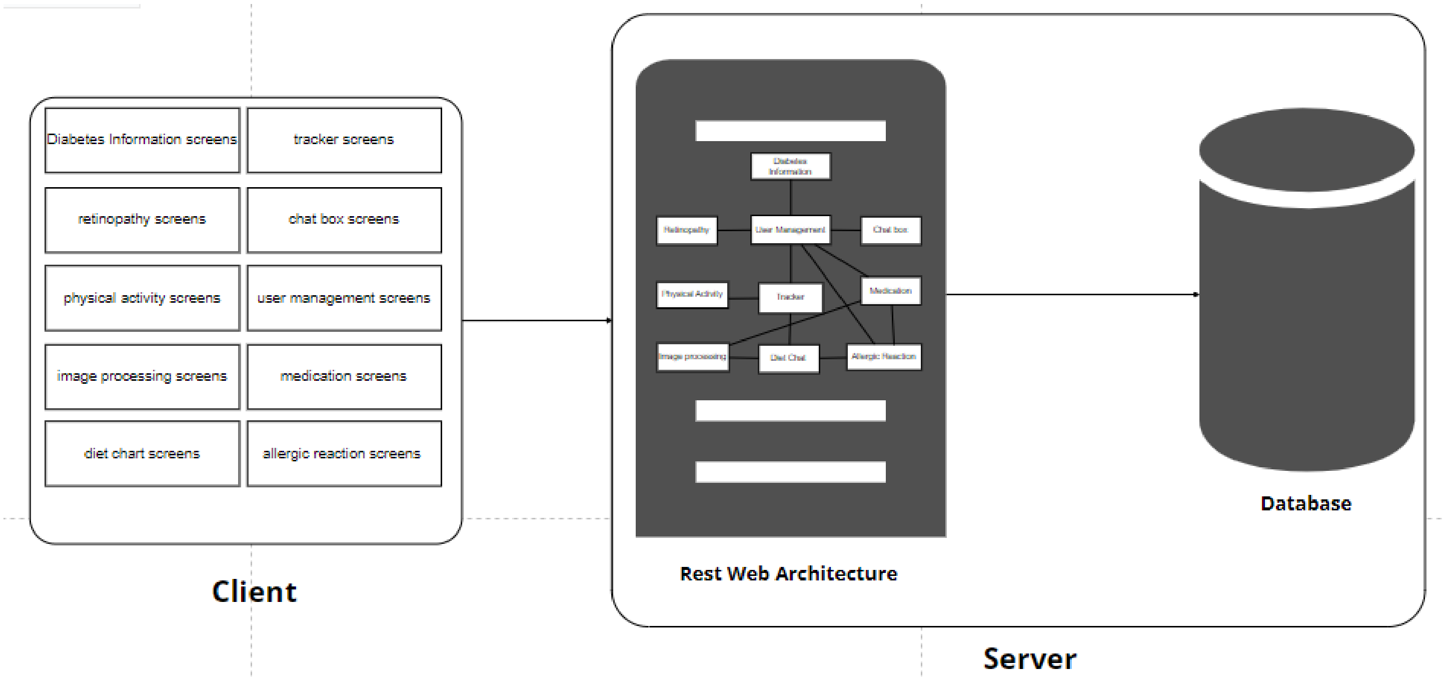
The architecture used in Smart Diabetelog would be two-tiered client-server architecture. The client would be responsible for sending a request to the server like adding and saving a new blood sugar instance, login into the account, determining the meal calories through food image, view diabetic information, send message in chat box etc. For every process from login to image processing, the client would be sending a http request to the rest architecture-based server.

The server follows Rest Architecture because it matches the stateless and idempotent nature of http protocol.

The server would entertain the request by first receiving the request, then processing it for the result through different algorithms and models if needed. The server would also be connected to the cloud database which would be responsible for data storage. All the data send from client to server through requests would be stored in this cloud database and would be used by server for processing afterwards.

The client server architecture is used because all the user's added information and machine learning models training, and implementation needs to be deployed somewhere else (at server) where the client has access rather than at client side because otherwise the storage capacity of the client side would be greatly burdened.

Table 3-1: Architectural design diagram



3.2 Design Models

The Smart Diabetolog will be using the functional approach so the following models can be created:

3.2.1 Activity Diagram

The following section contains Activity diagrams of Smart Diabetolog Application.

3.2.1.1 Login

Figure 3-1 contains an activity diagram for login where diabetic patient enters email and password. After matching the email and password the account is logged in for the patient according to his details.

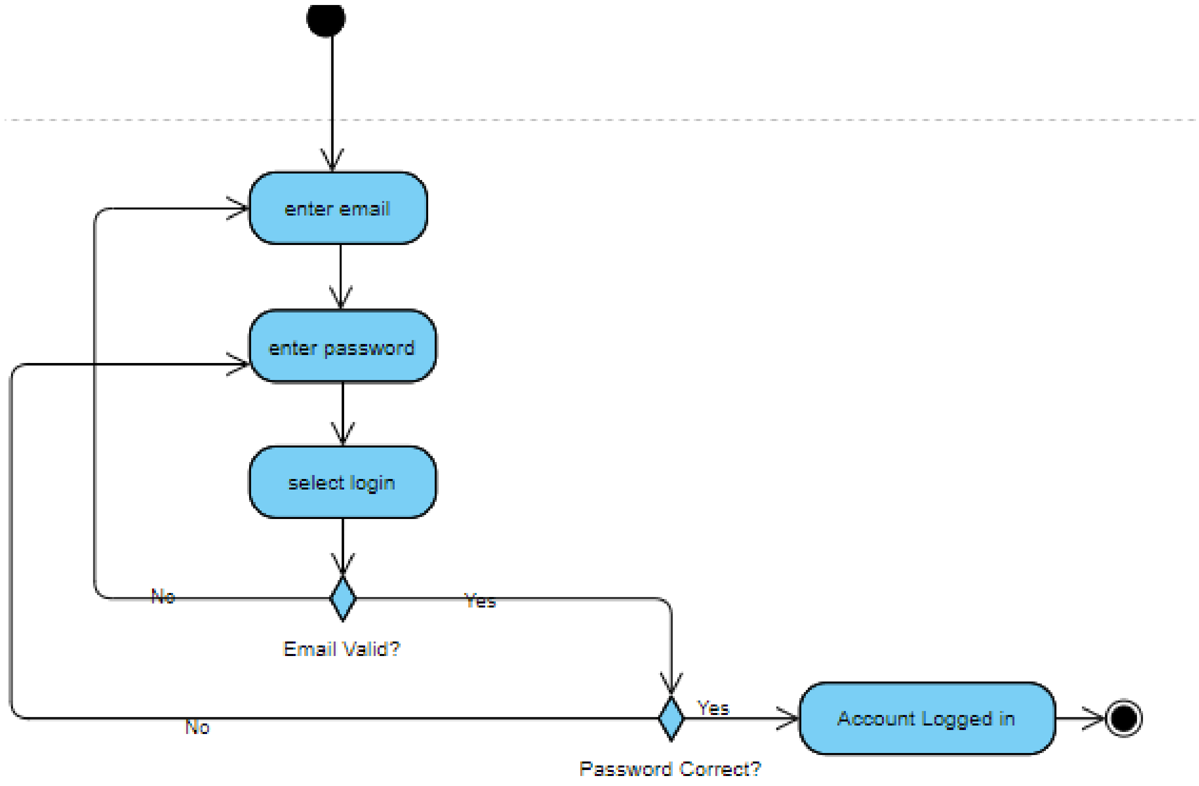


Figure 3-1: Login Activity Diagram

3.2.1.2 Register:

Figure 3-2 contains an activity diagram for the registration process. In this process the user enters the required details. The application ensures the user owns the email by sending verification code on his email. The patient then enters this code and verifies his email. The account for the patient is then registered according to the given details.

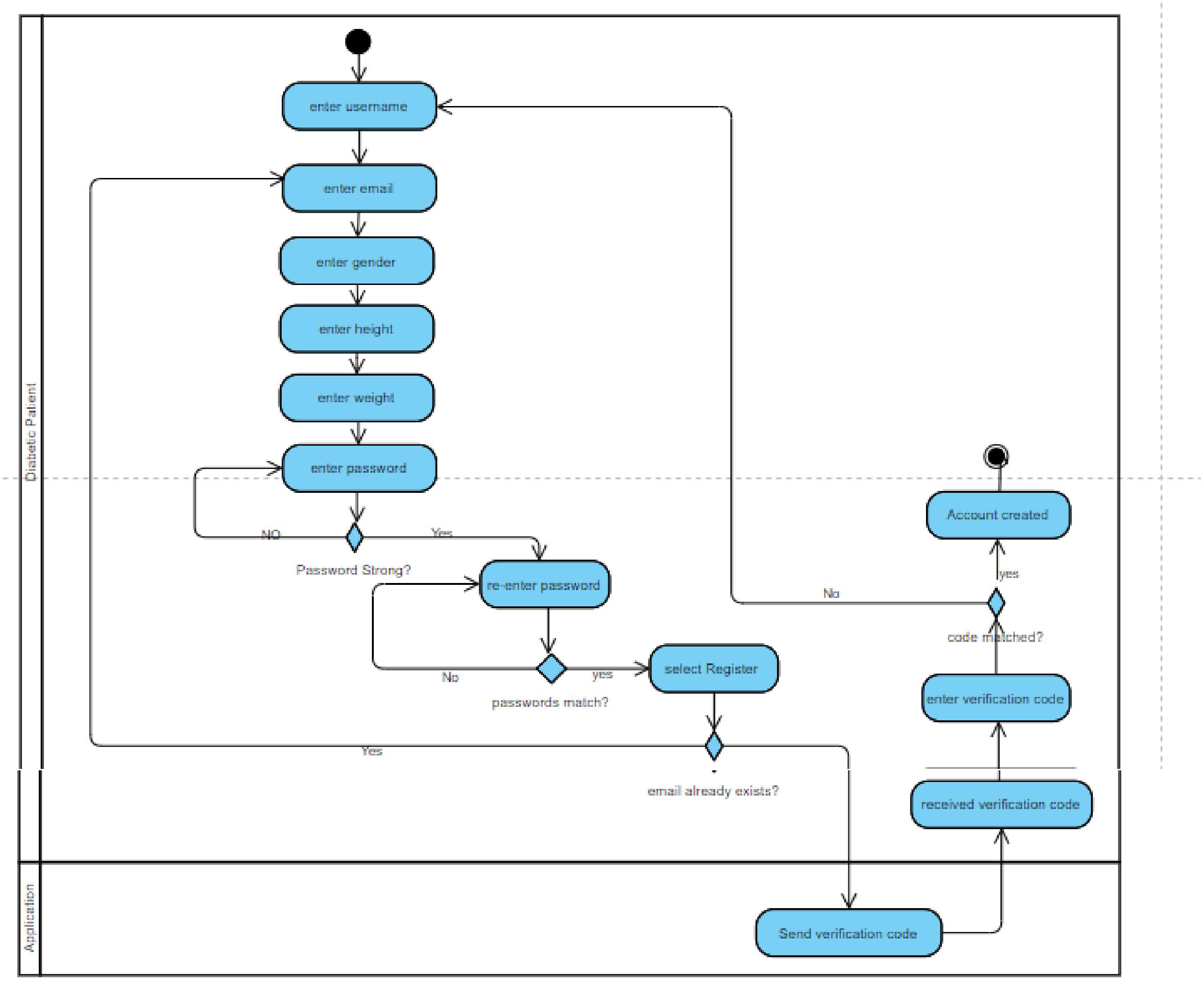


Figure 3-2: Register Activity Diagram

3.2.1.3 Add new meal

Figure 3-3 contain an activity diagram for flow of adding meal. The diabetic patient selects the way through which he wants to save his meal. He can either select manual way in which he enters the required information then selects time and save the meal. Or he can add meal through image in which user adds the image of front and side of meal. The system predicts the dish name and when user selects name of the dish the calories information of the dish is predicted. The patient can save this information after confirming it.

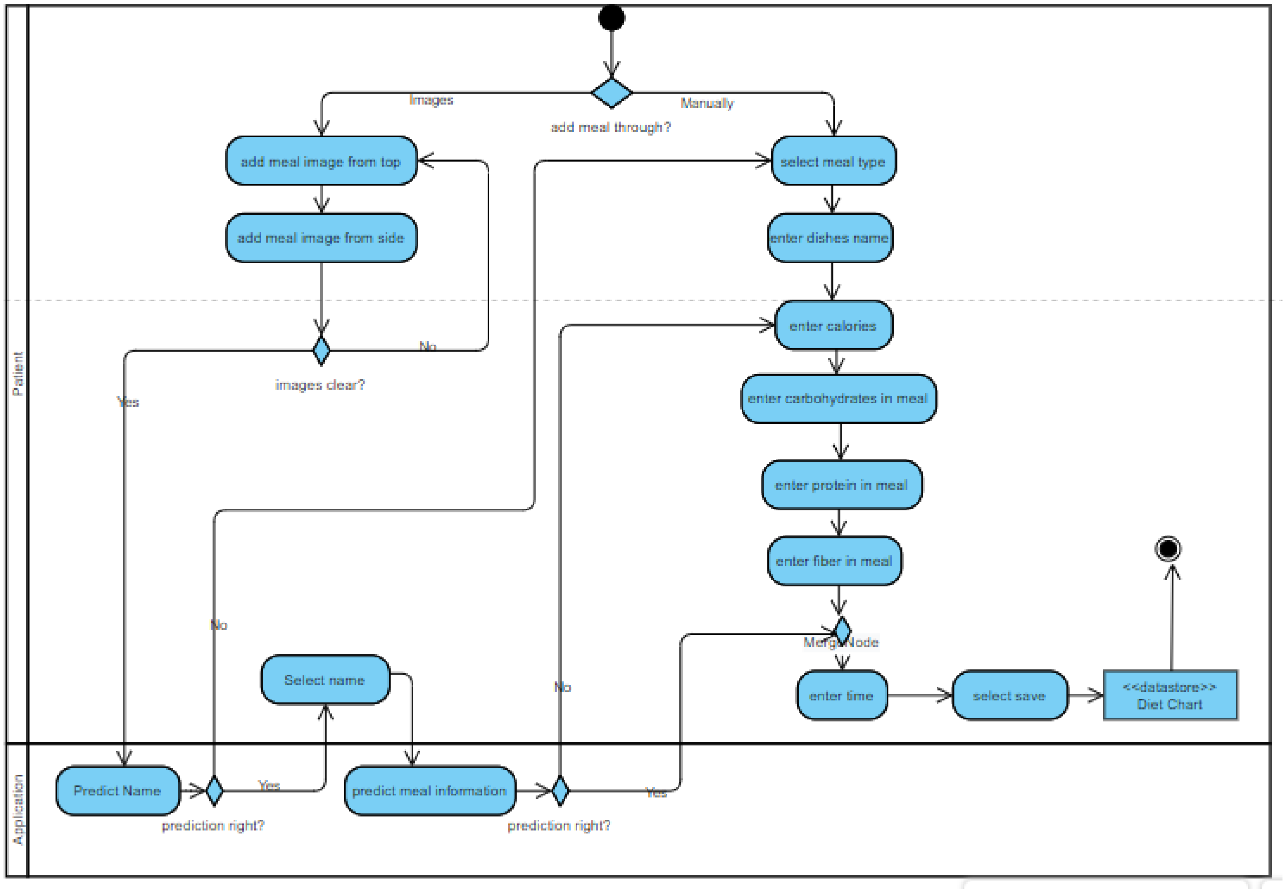


Figure 3-3: Add Meal Activity Diagram

3.2.1.4 Edit Profile

Figure 3-4 displays activity diagram for editing profile of the patient. He can edit the details of the profile by selecting what he wants to edit. Then changing the information of the selected item. The updated changes will be saved to the database.

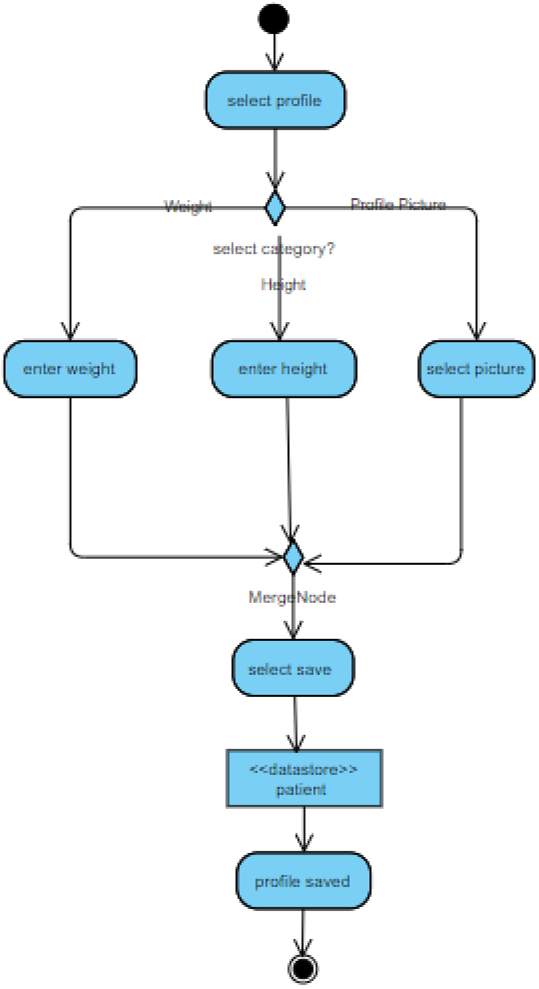


Figure 3-4: Edit Profile Activity Diagram

3.2.1.5 View allergic reaction history:

Figure 3-5 contains an activity diagram for viewing allergic reaction history of the patient. The user can view the allergic reaction history based on medication or food.

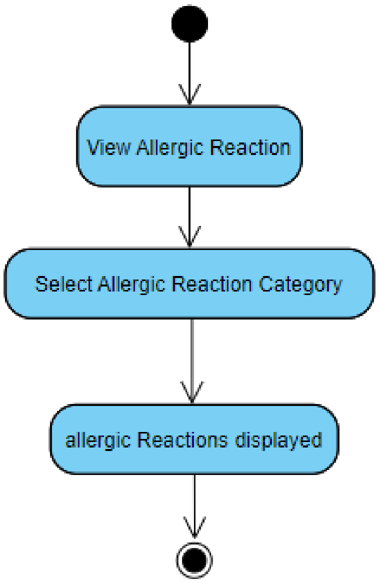

View Allergic Reaction
Select Allergic Reaction Category
allergic Reactions displayed

Figure 3-5: View Allergic Reaction History Activity Diagram

3.2.1.6 View allergic reaction:

Figure 3-6 contains an activity diagram for the particular instance of allergic reaction details to the user.

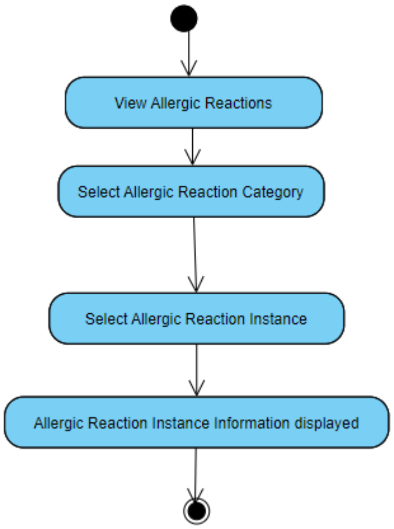

View Allergic Reactions
Select Allergic Reaction Category
Select Allergic Reaction Instance
Allergic Reaction Instance Information displayed

Figure 3-6: View Allergic Reaction Activity Diagram

3.2.1.7 Add allergic reactions:

Figure 3-7 displays an activity diagram which shows how a user adds a particular instance of allergic reaction of selected allergic reaction category into system.

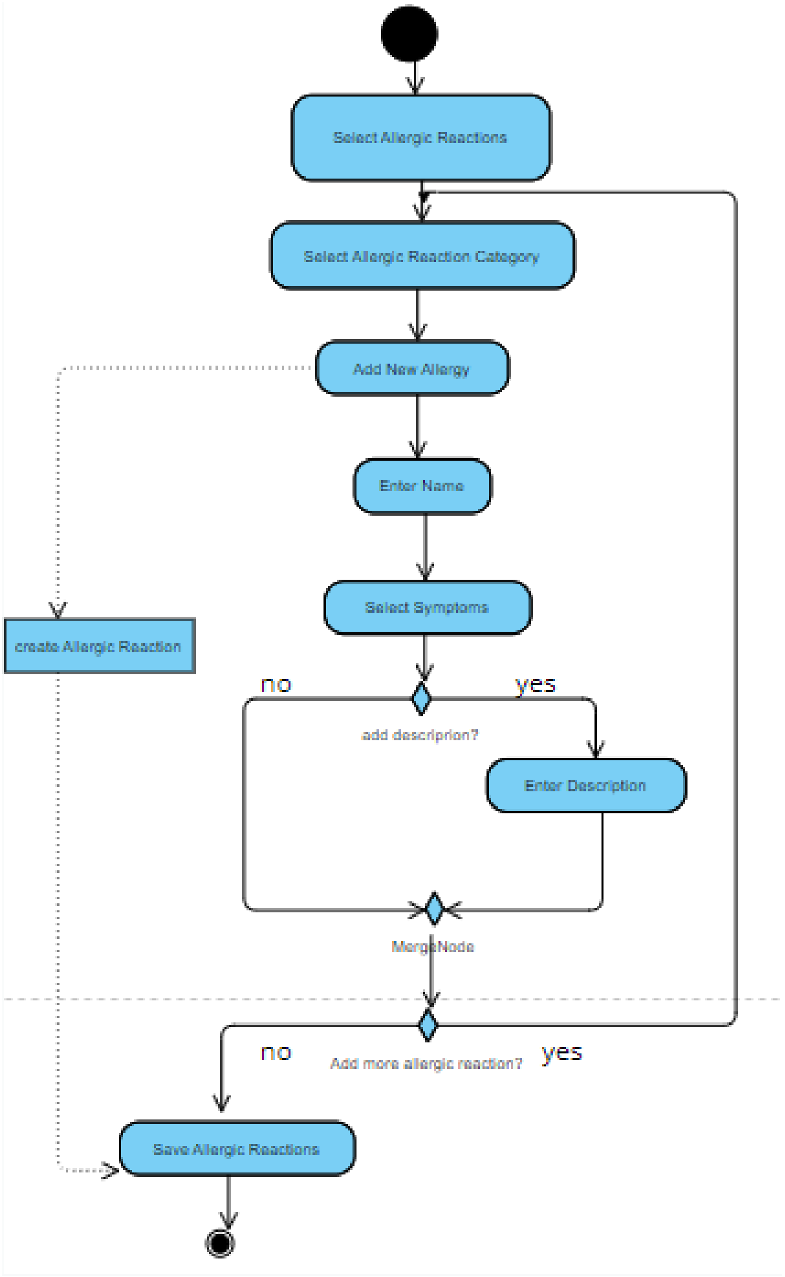


Figure 3-7: Add Allergic Reaction Activity Diagram

3.2.1.8 Update allergic reaction history:

Figure 3-8 displays an activity diagram which shows how the user updates a particular instance of the selected allergic reaction of a particular category.

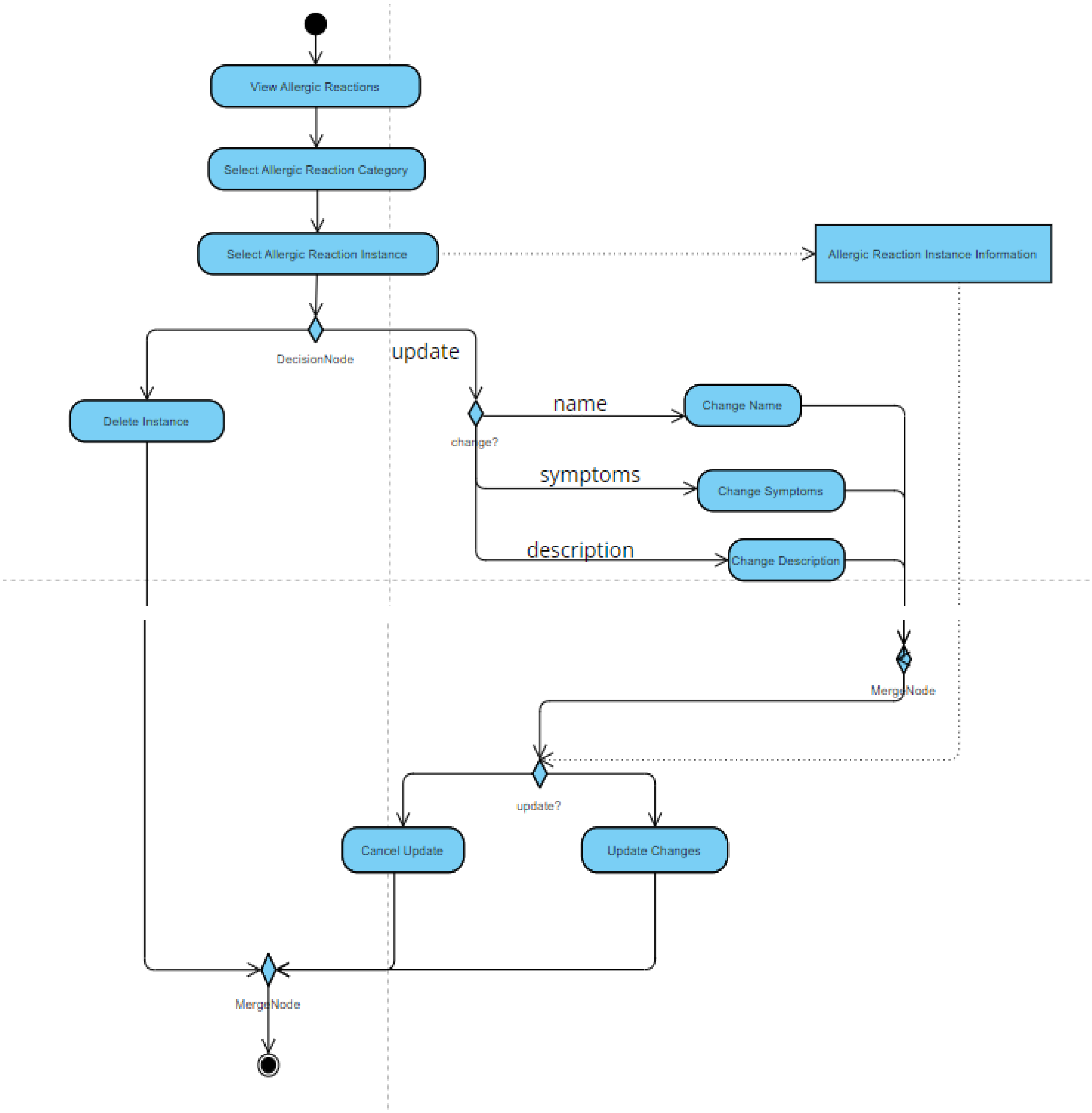


Figure 3-8: Update allergic reaction history Activity Diagram

3.2.1.9 Add new oral medication:

Figure 3-9 displays an activity diagram that shows how the user adds an oral medication into a new or already existing prescription.

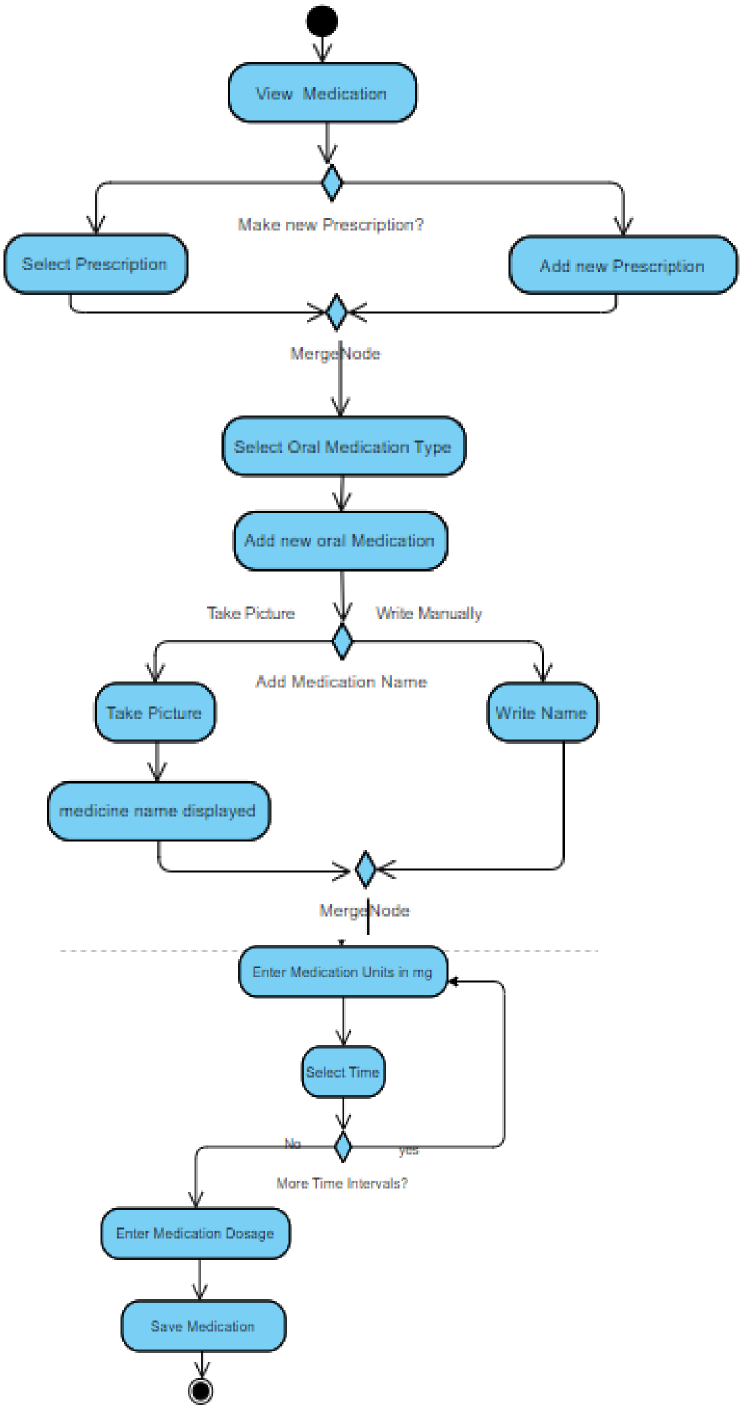
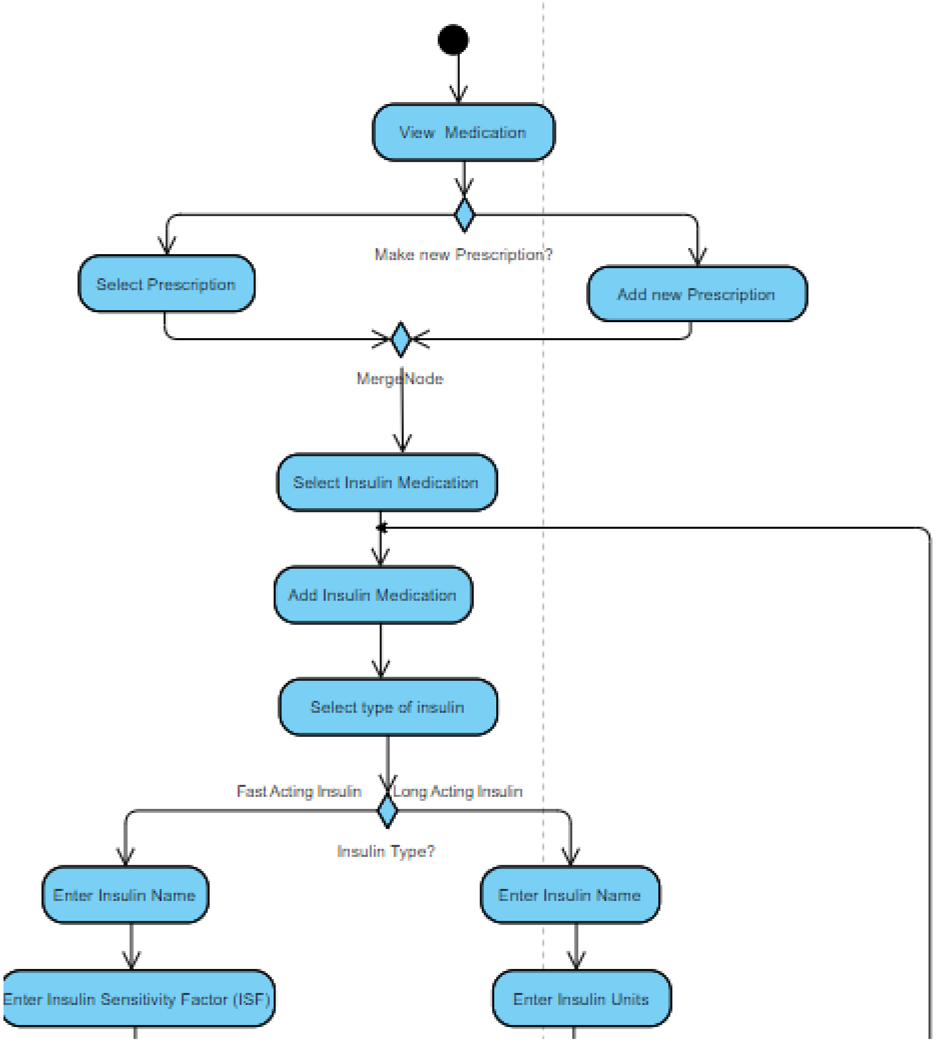


Figure 3-9: Add new oral medication Activity Diagram

3.2.1.10 Add insulin medication:

Figure 3-2 shows an activity diagram that shows how the user adds an insulin medication into a new or already existing prescription.



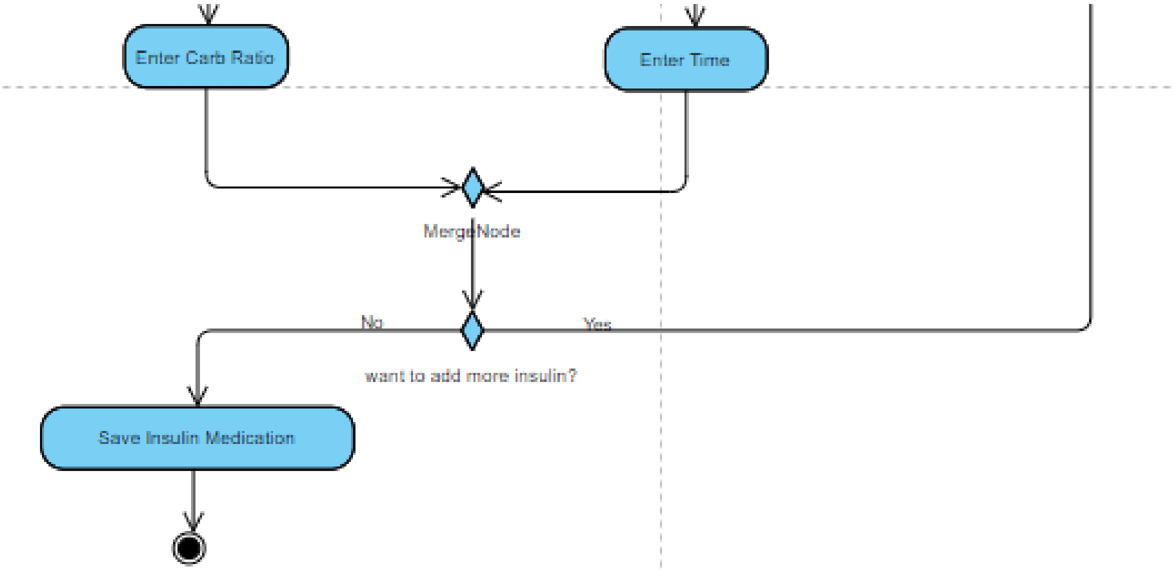


Table 3-2: Add insulin medication activity diagram

3.2.1.11 *View current oral medication*

Figure 3-11 contains an activity diagram showing how the user views all the oral medications of the current prescription.

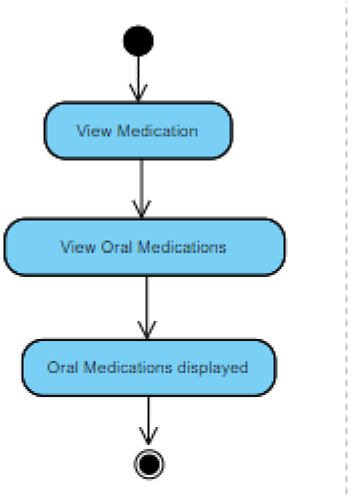


Figure 3-11: View current oral medication Activity Diagram

3.2.1.12 View current insulin medication

Figure 3-12 contains an activity diagram to shows how the user views current insulin of the current prescription.

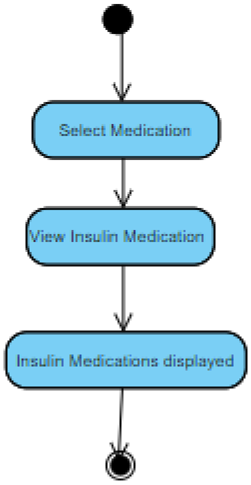


Figure 3-12: View current insulin medication Activity Diagram

3.2.1.13 View current medication prescription:

Figure 3-13 contains an activity diagram that shows how the user views all the medications in the current prescription.

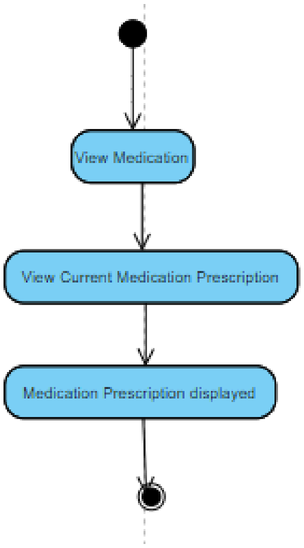

View Medication
View Current Medication Prescription
Medication Prescription displayed

Figure 3-13: View current medication prescription Activity Diagram

3.2.1.14 **View statistics***:*

Figure 3-14 displays an activity diagram that shows how the user views the statistics of the particular health factor.

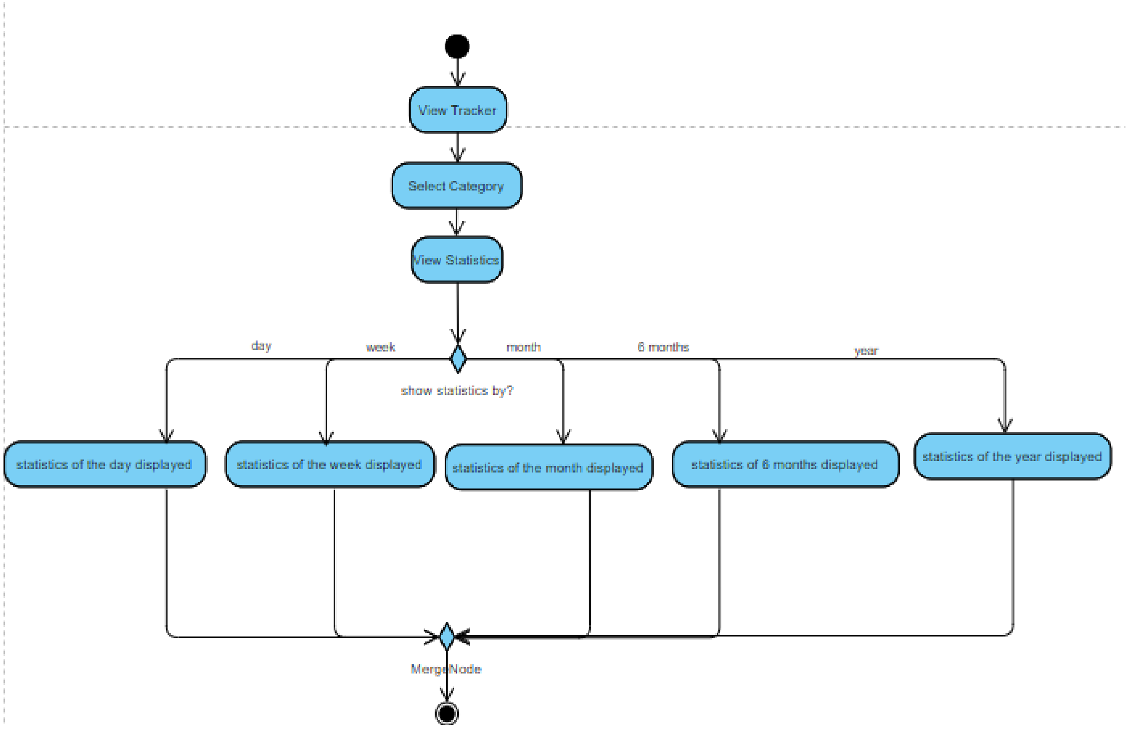

View Tracker
Select Category
View Statistics
day
week
month
6 months
year
show statistics by?
statistics of the day displayed
statistics of the week displayed
statistics of the month displayed
statistics of 6 months displayed
statistics of the year displayed
MergeNode

Figure 3-14: View statistics Activity Diagram

3.2.1.15 Update oral medication:

Figure 3-15 displays an activity diagram that shows how the user updates the oral medications of any stored prescription.

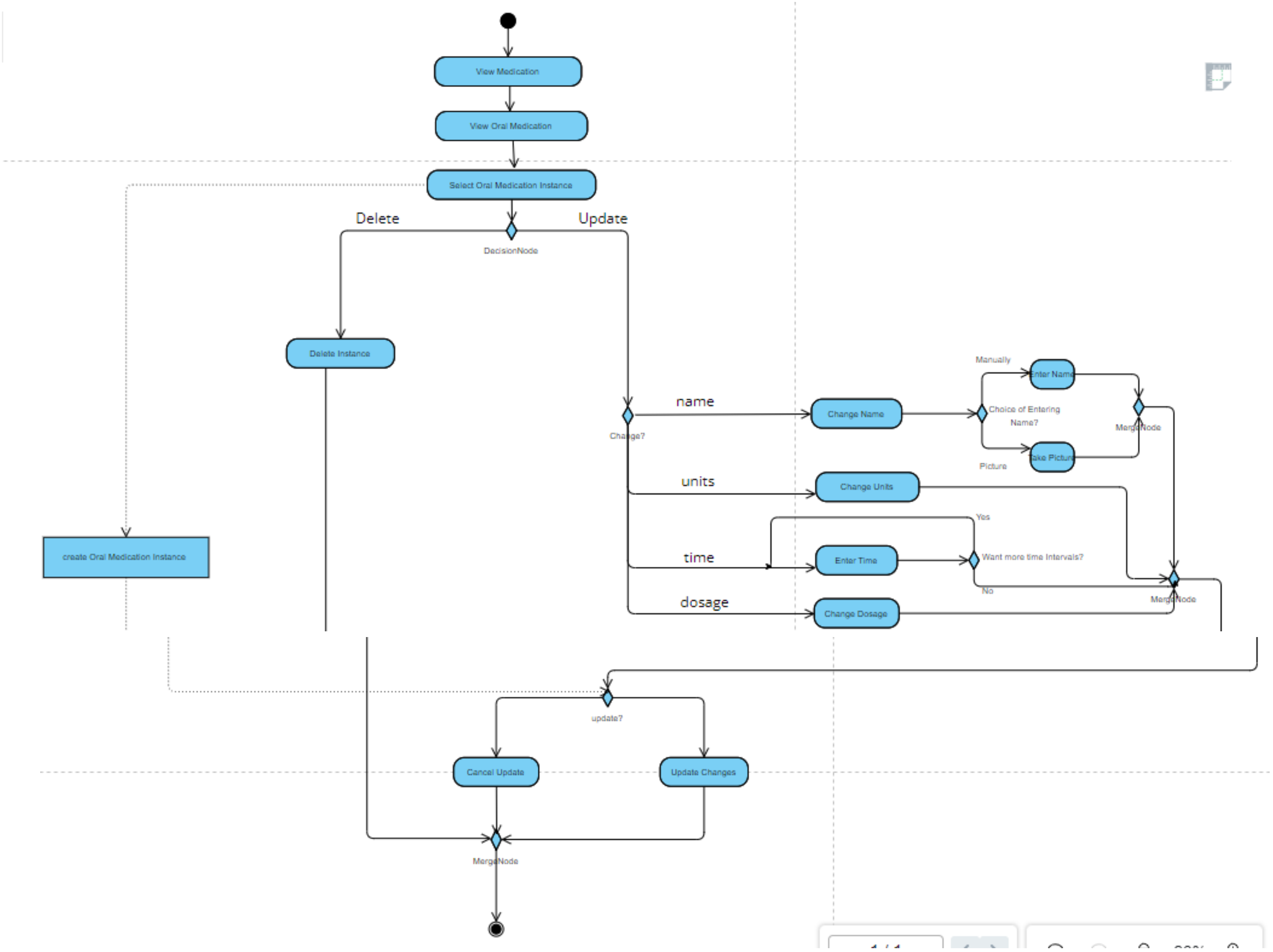


Figure 3-15: Update oral medication Activity Diagram

3.2.1.16 Update insulin medication:

Figure 3-16 displays an activity diagram that shows how the user updates the insulin medication of any prescription.

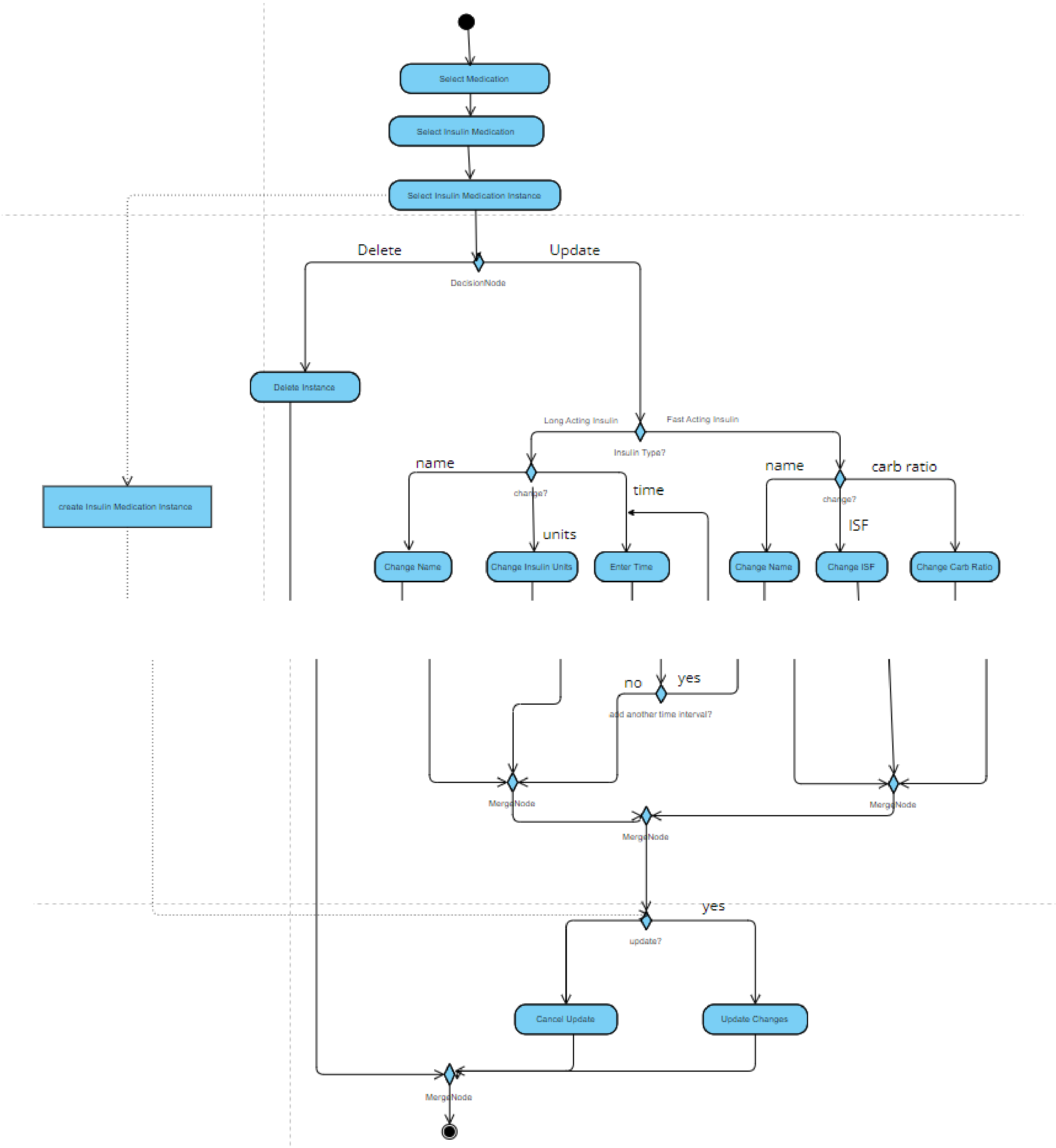


Figure 3-16: Update Insulin Medication Activity Diagram

3.2.1.17 View prescription history

Figure 3-17 displays an activity diagram that shows how the user views any particular stored prescription details.

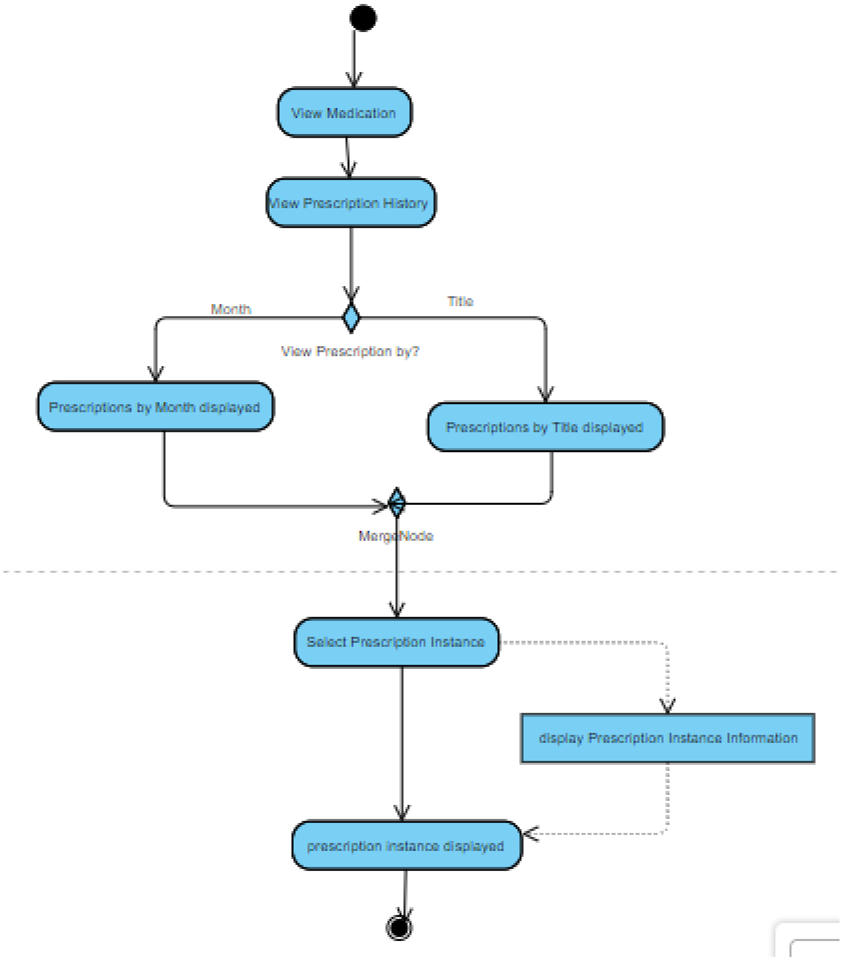

View Medication
View Prescription History
Month
Title
View Prescription by?
Prescriptions by Month displayed
Prescriptions by Title displayed
MergeNode
Select Prescription Instance
display Prescription Instance Information
prescription instance displayed

Figure 3-17: View Prescription History Activity Diagram

3.2.1.18 Change prescription title:

Figure 3-18 displays an activity diagram that shows how the user changes the prescription title of any stored prescription.

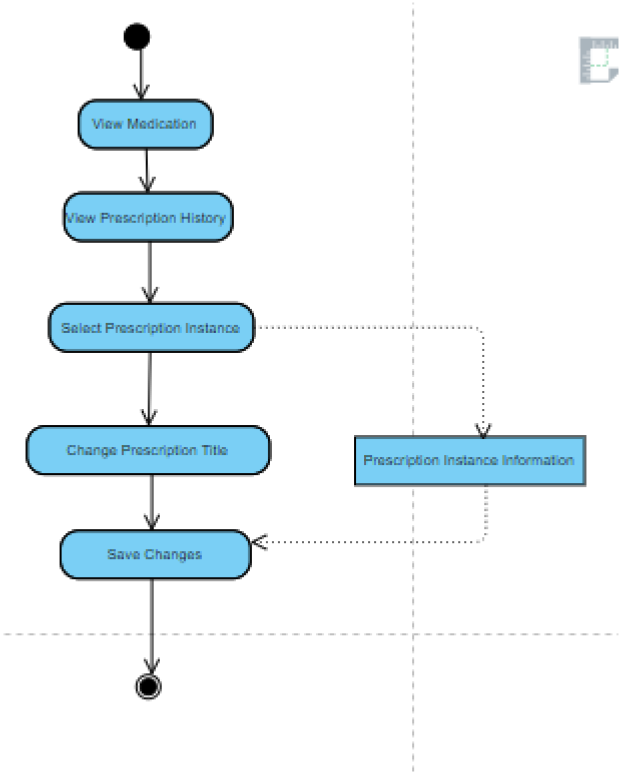

View Medication
View Prescription History
Select Prescription Instance
Change Prescription Title
Prescription Instance Information
Save Changes

Figure 3-18: Change prescription title Activity Diagram

3.2.1.19 View tracker:

Figure 3-19 displays an activity diagram that shows how the user views any stored health factor instances in detail.

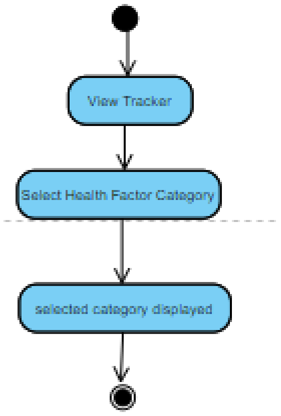


Figure 3-19: View Tracker Activity Diagram

3.2.1.20 Add blood pressure:

Figure 3-20 displays an activity diagram that shows how the user adds a new blood pressure instance into the tracker.

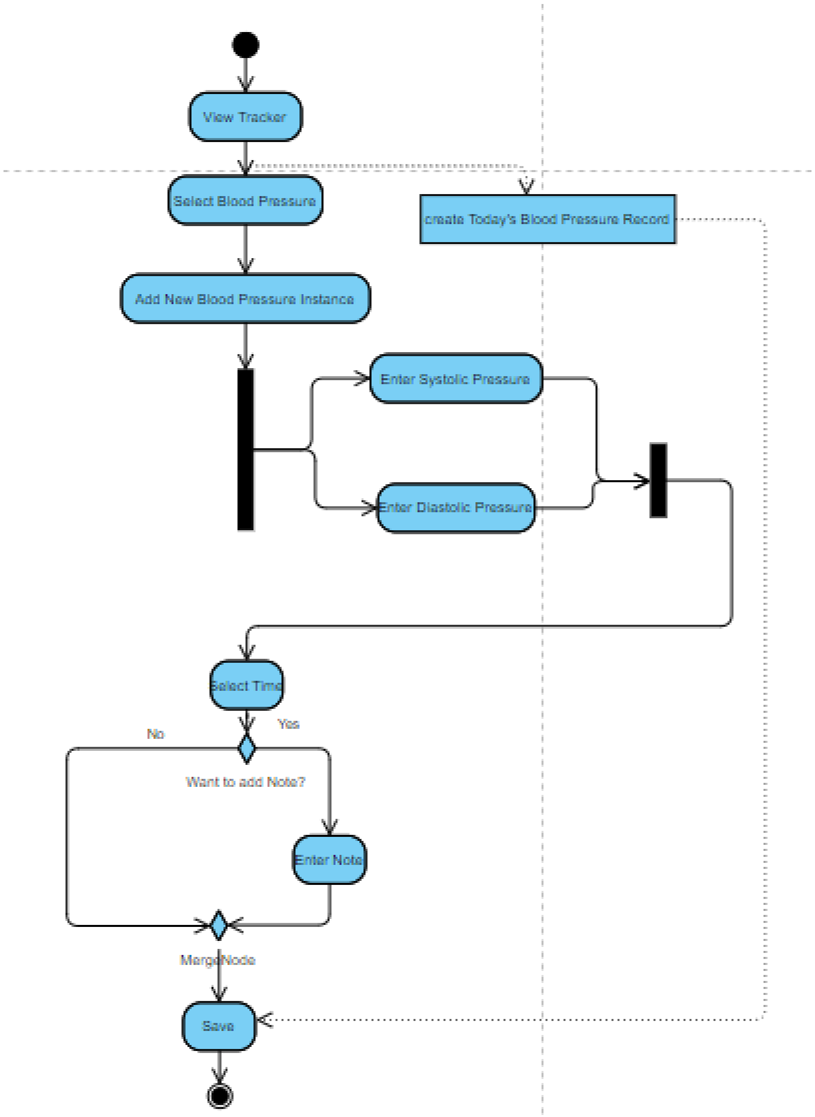


Figure 3-20: Add blood pressure Activity Diagram

3.2.1.21 Add blood sugar

Figure 3-21 displays an activity diagram that shows how the user adds a new blood sugar instance into the tracker.

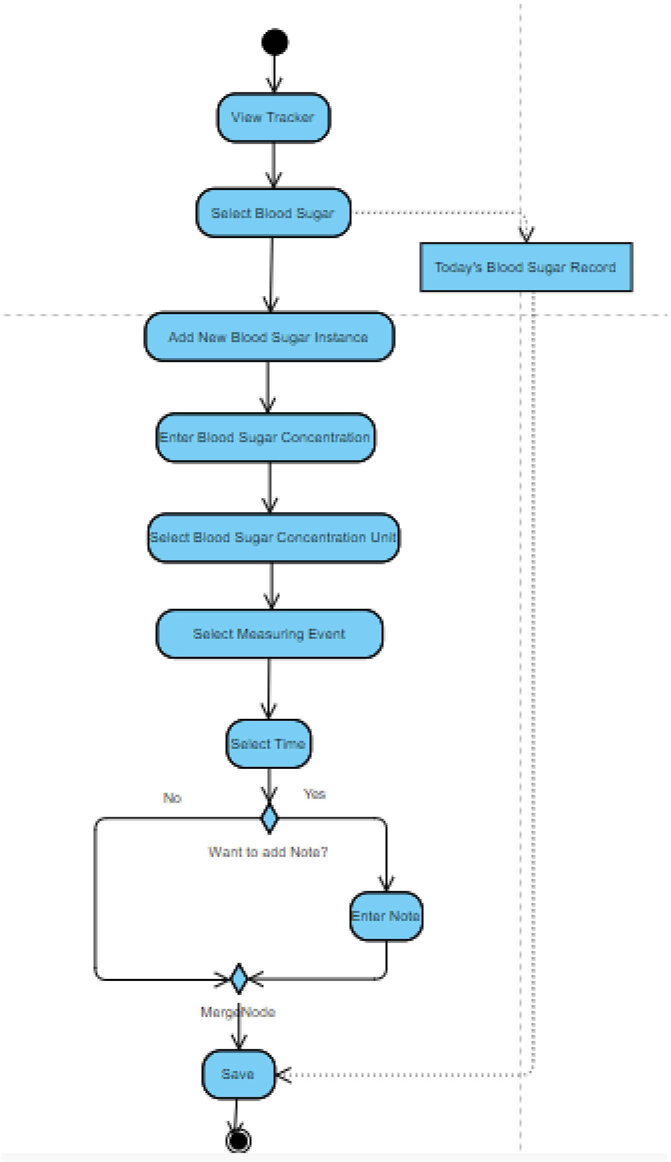

View Tracker
Select Blood Sugar
Today's Blood Sugar Record
Add New Blood Sugar Instance
Enter Blood Sugar Concentration
Select Blood Sugar Concentration Unit
Select Measuring Event
Select Time
No
Yes
Want to add Note?
Enter Note
MergeNode
Save

Figure 3-21: Add Blood Activity Diagram

3.2.1.22 Add cholesterol:

Figure 3-22 displays an activity diagram that shows how the user adds a new cholesterol instance into the tracker.

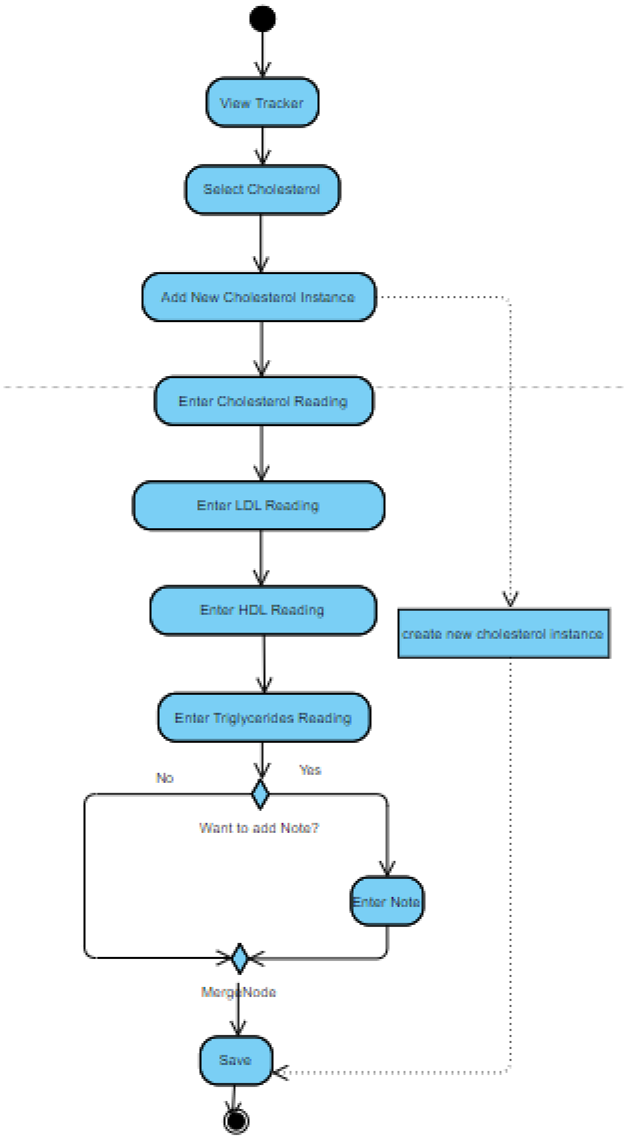

View Tracker
Select Cholesterol
Add New Cholesterol Instance
Enter Cholesterol Reading
Enter LDL Reading
Enter HDL Reading
create new cholesterol instance
Enter Triglycerides Reading
No
Yes
Want to add Note?
Enter Note
MergeNode
Save

Figure 3-22: Add cholesterol Activity Diagram

3.2.1.23 Update blood pressure:

Figure 3-23 displays an activity diagram that shows how the user updates any blood pressure instance from the tracker.

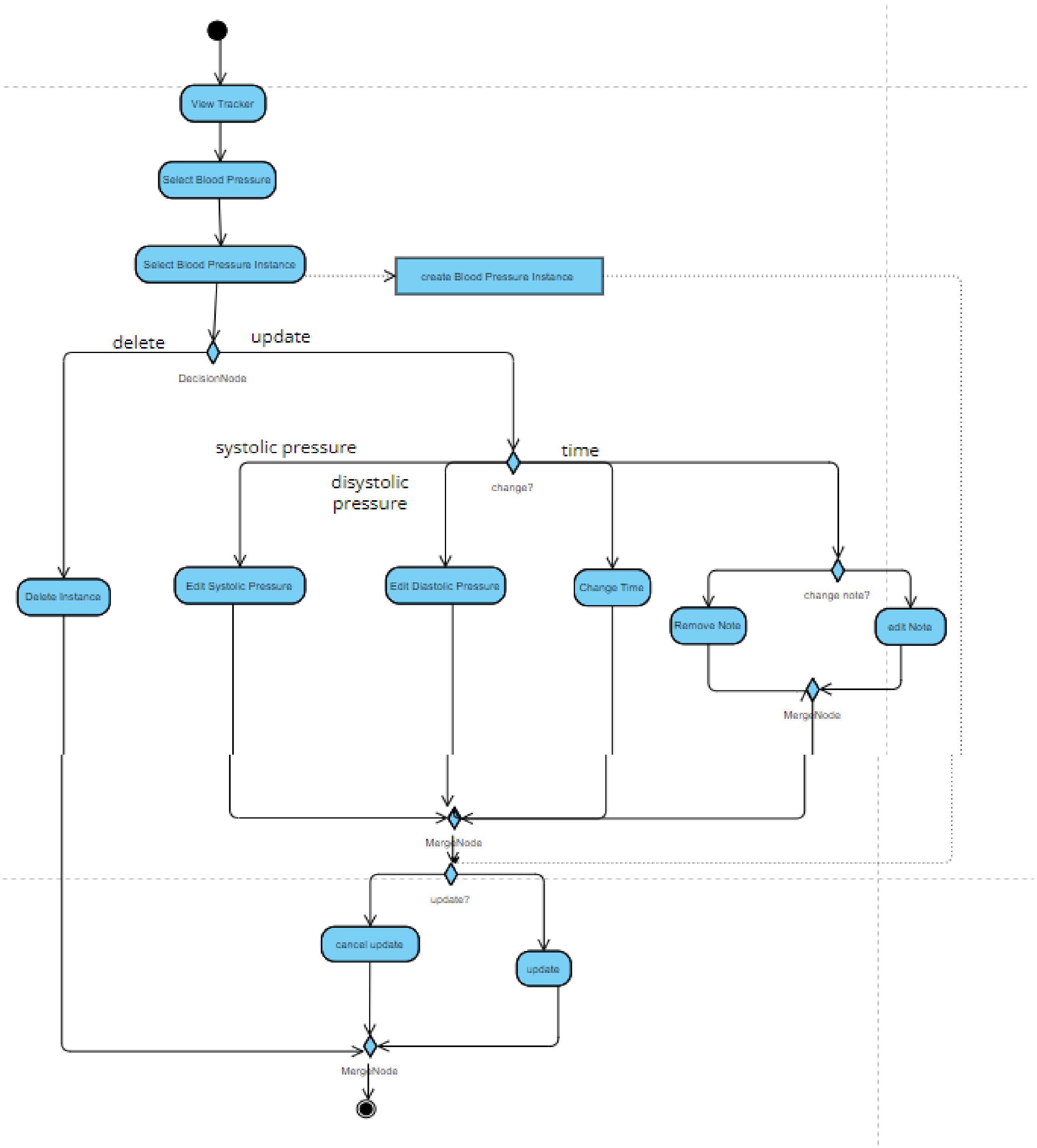


Figure 3-23: Update blood pressure Activity Diagram

3.2.1.24 Update blood sugar:

Figure 3-24 displays an activity diagram that shows how the user updates any blood sugar instance from the tracker.

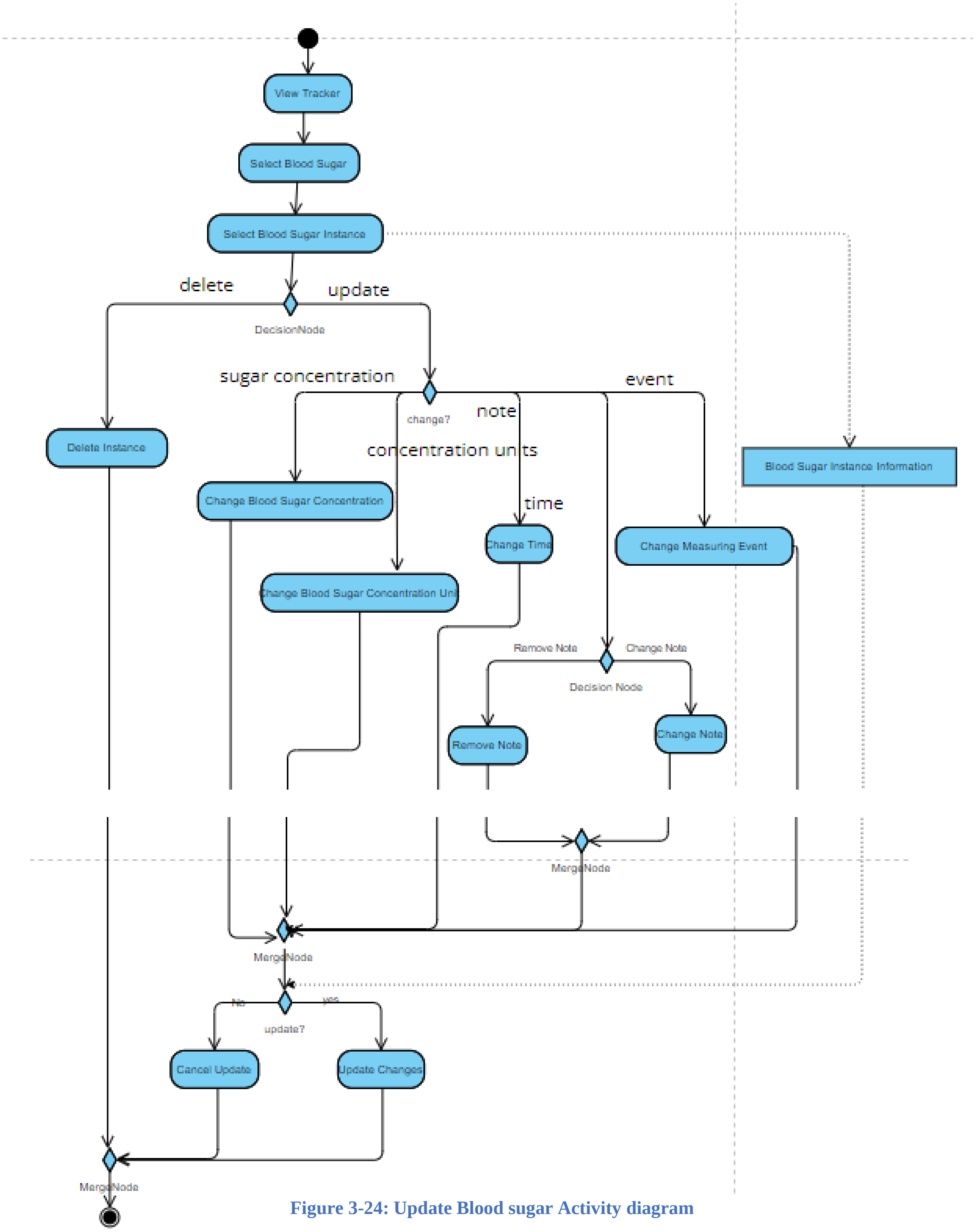


Figure 3-24: Update Blood sugar Activity diagram

3.2.1.25 Update cholesterol:

Figure 3-26 displays an activity diagram that shows how the user updates any cholesterol instance from the tracker.

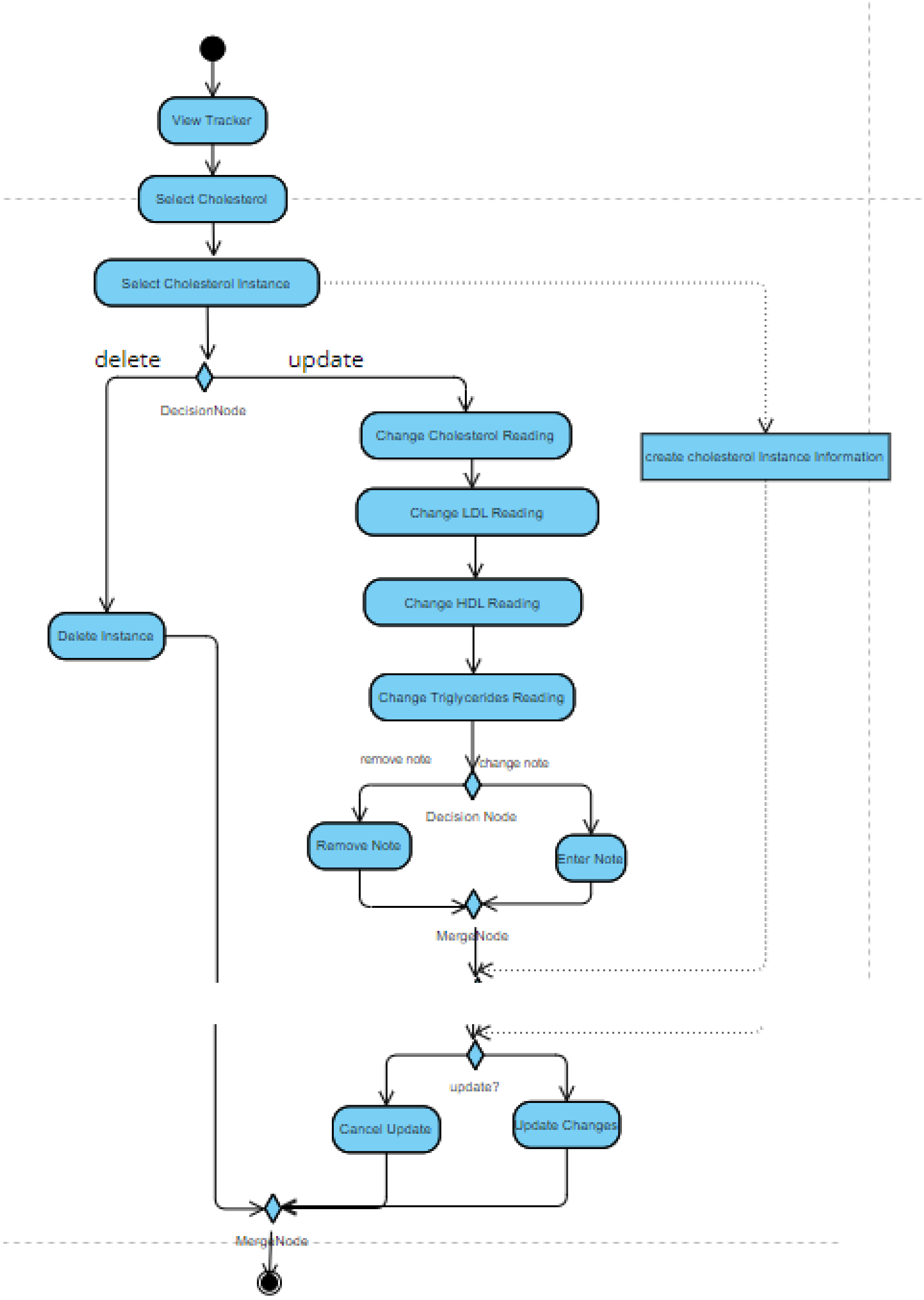


Figure 3-26: Update cholesterol Activity Diagram

3.2.1.26 View all factor statics:

Figure 3-27 displays an activity diagram that shows how the user can view the statistics of all health factors.

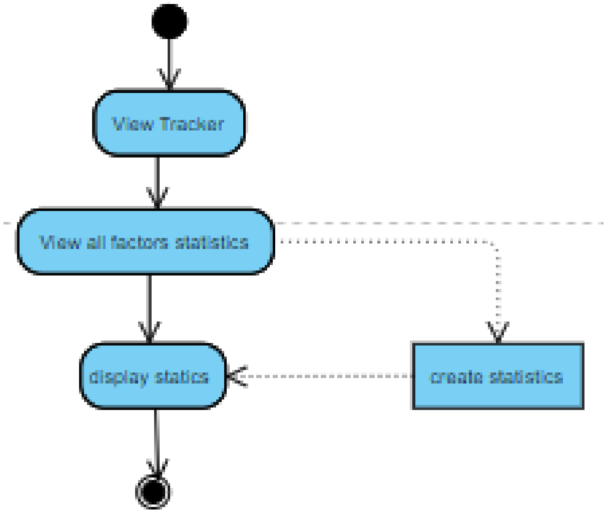


Figure 3-27: View all factor statics Activity Diagram

3.2.1.27 Change health factor tracking:

Figure 3-28 displays an activity diagram that shows how the user can stop or start the tracking of any particular heath factor in the tracker.

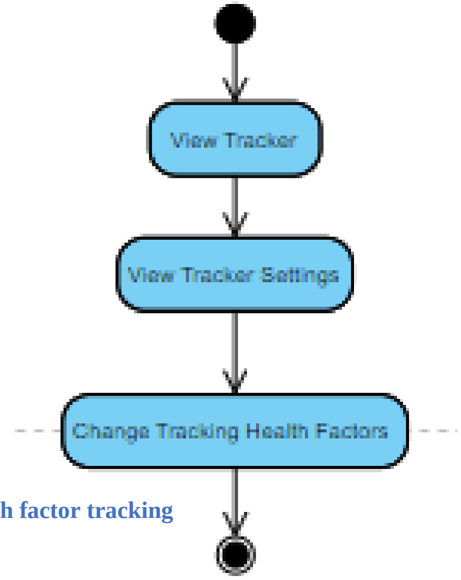


Figure 3-28: Change health factor tracking activity diagram.

3.2.1.28 View Today meal plan:

Figure 3-29 displays an image below that shows flow of viewing meal plan. The diabetic patient first selects meal plan, and which will be shown from the database. The patient can further select meal plan for a particular time and can view it suggested dishes.

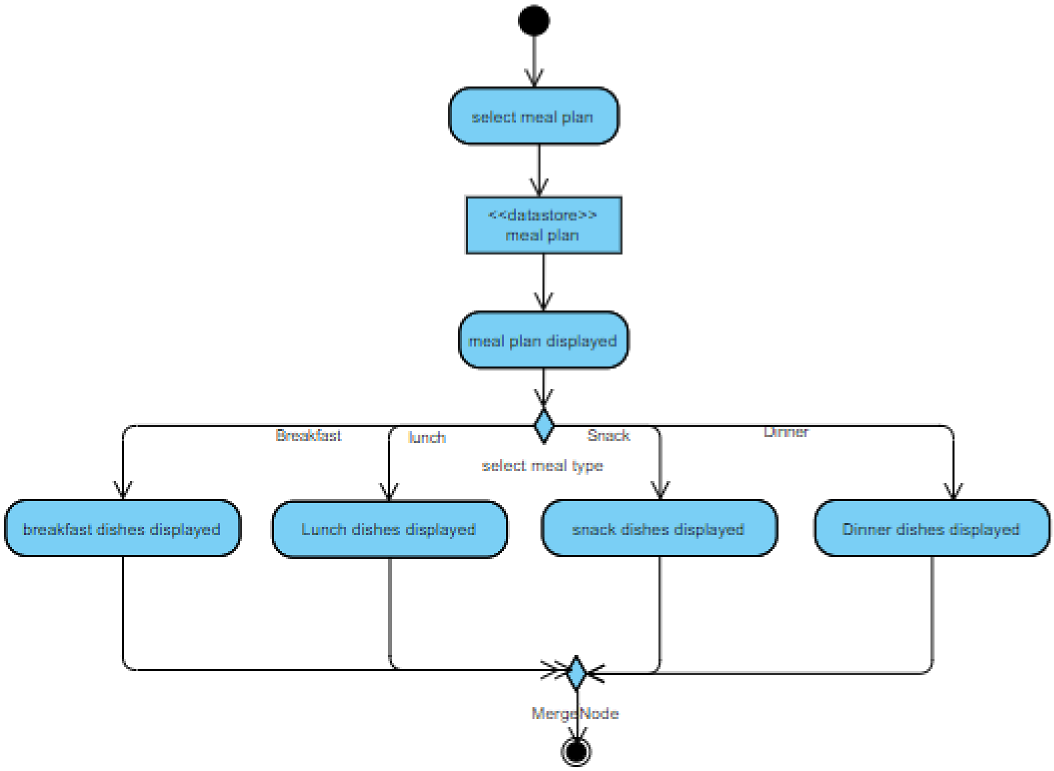


Figure 3-29: View Today meal plan Activity Diagram

3.2.1.29 Save meal from meal plan

Figure 3-30 display an activity diagram of the process flow for saving meal from meal plan to meal history. Patients can directly save meals to the database.

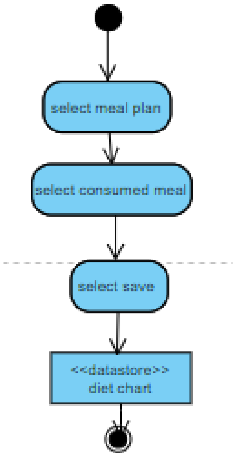


Figure 3-30: Save meal from meal plan Activity Diagram

3.2.1.30 View diet chart:

The figure 3-31 below shows process flow for viewing diet chart. The diet chart contains the history of the patient previous meals which are stored in database. This history will be displayed from database.

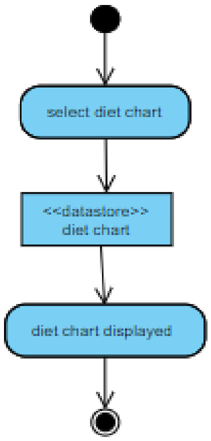


Figure 3-31: View diet chart Activity Diagrams

3.2.1.31 View BMI:

Figure 3-32 displays activity diagram for viewing BMI. The patient’s height and weight will be taken, and BMI will be calculated and then displayed to the patient.

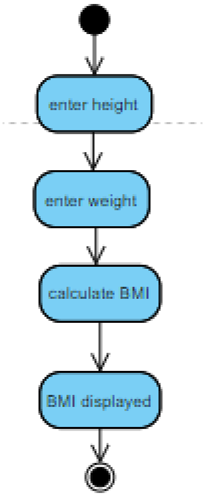


Figure 3-32: View BMI Activity Diagram

3.2.1.32 Start exercise:

Figure 3-33 displays an activity diagram for starting exercise. After starting the exercise, it can be stopped. The exercise can also be paused and resumed by the patient.

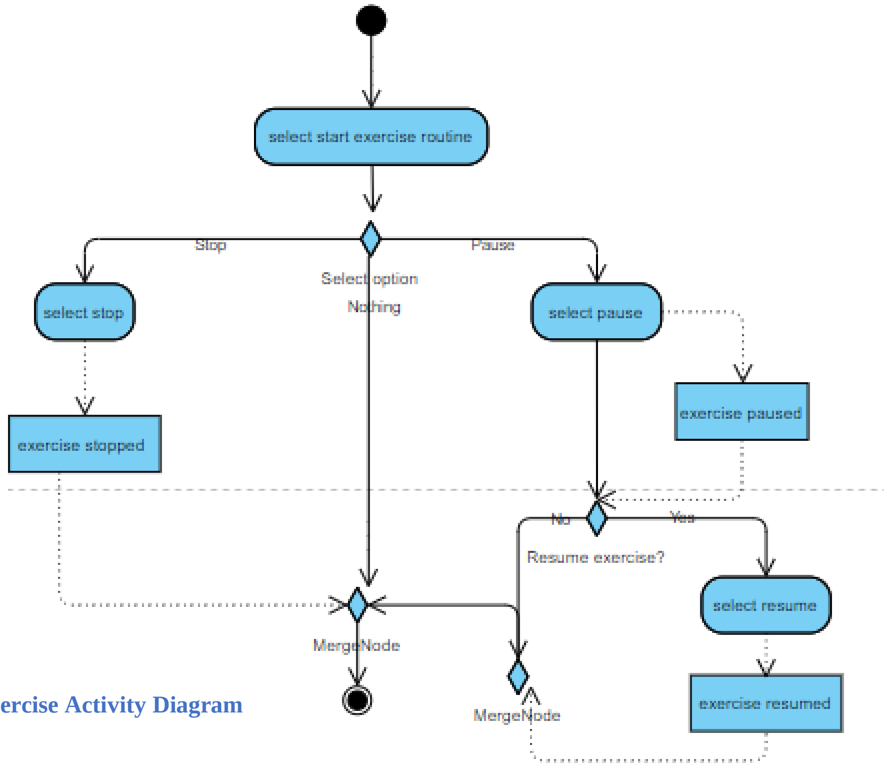


Figure 3-33: Start exercise Activity Diagram

3.2.1.33 View exercise history

figure 3-34 displays an activity diagram for viewing exercise history. The exercise history contains the history of a patient’s physical activity. This physical activity details will be stored in a database which will be displayed upon patients request.

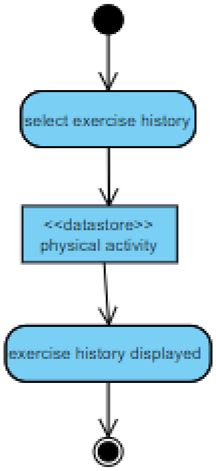


Figure 3-34: View exercise history Activity Diagram

3.2.1.34 Participate in live chat:

Figure 3-35 displays an activity diagram for sending message in live chat. the patient can send messaged along with images in live chat to talk to other patients.

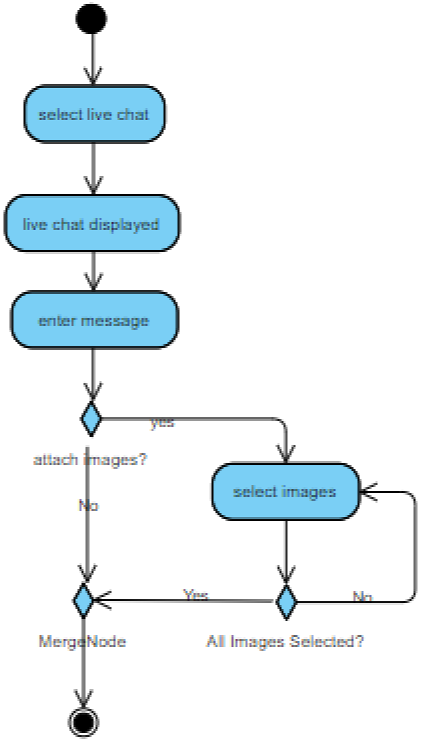


Figure 3-35: Participate in live chat Activity Diagram

3.2.1.35 View diabetes information:

Figure 3-36 displays an activity diagram for the process of viewing diabetes related information. The patient will request diabetes information by specifying the type of information i.e. blogs, news, videos and the content will be displayed to the patient by the system.

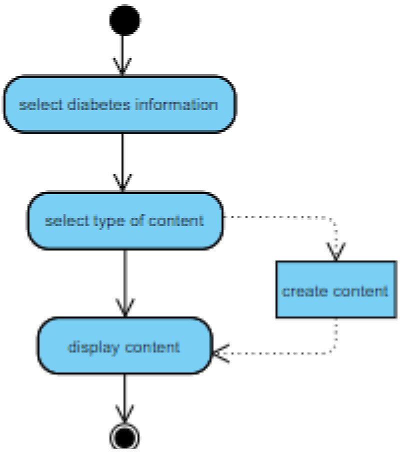


Figure 3-36: View diabetes information Activity Diagram

3.2.1.36 Post in discussion board:

Figure 3-37 displays an activity diagram for posting on the discussion board. The patient can post a query on the discussion board.

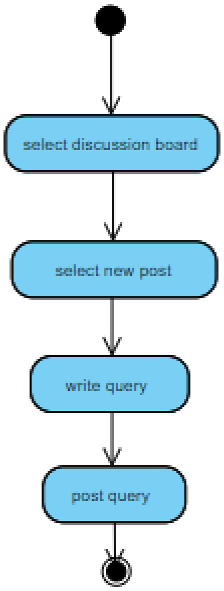


Figure 3-37: Post in discussion board Activity Diagram

3.2.1.37 Reply to query in discussion board

Figure 3-38 displays an activity diagram for replying to a query on the discussion board. Patients can reply by either scrolling in the discussion board and selecting a particular query or by searching a query from discussion board. Then he can reply by selecting post the reply will be visible to all other patient after posting.

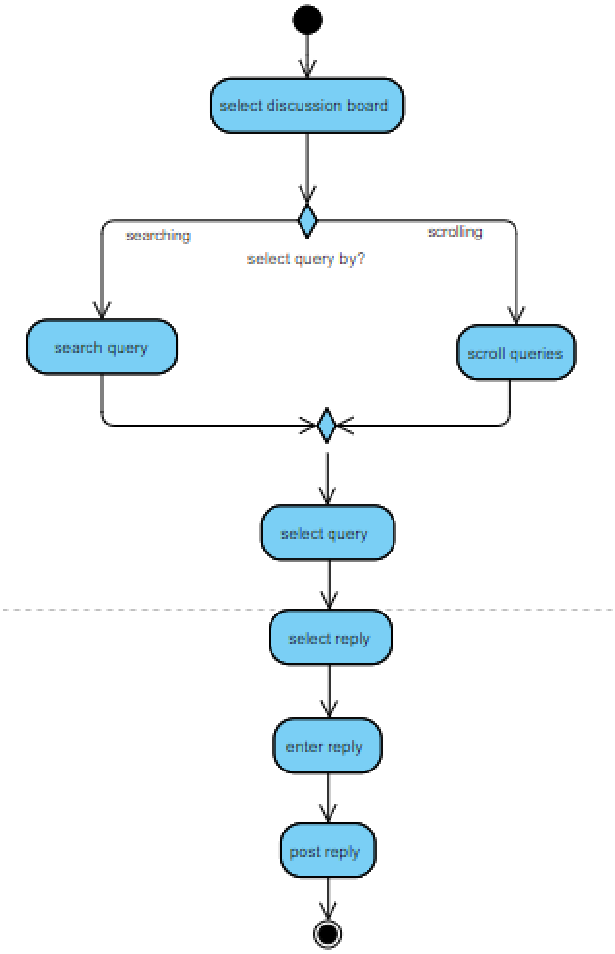


Figure 3-38: Reply to query in discussion board Activity Diagram

3.2.1.38 Delete account:

Figure 3-39 displays an activity diagram for the process of deleting account. The patient can delete account by selecting delete account, the account will be deleted from user's details database along with all patient's information.

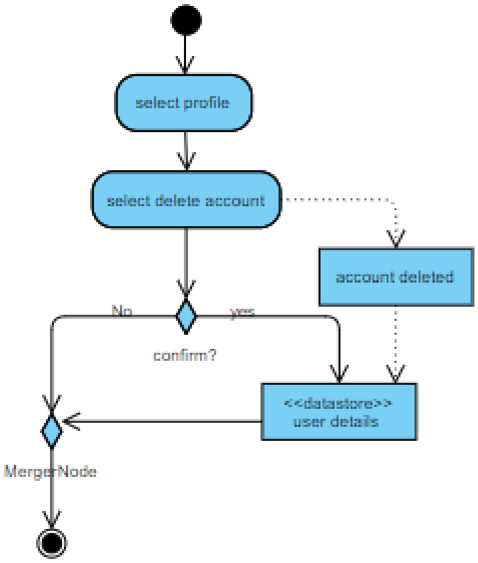


Figure 3-39: Delete Account Activity Diagram

3.2.2 Data flow diagrams

3.2.2.1 Level 0 DFD (Context Diagram)

Figure 3-40 displays a context diagram of Smart Diabetolog. The context diagram represents how external system and actors are interacting with our system. The diabetic patient interacts with our system and system responds to the patient by interacting with external systems.

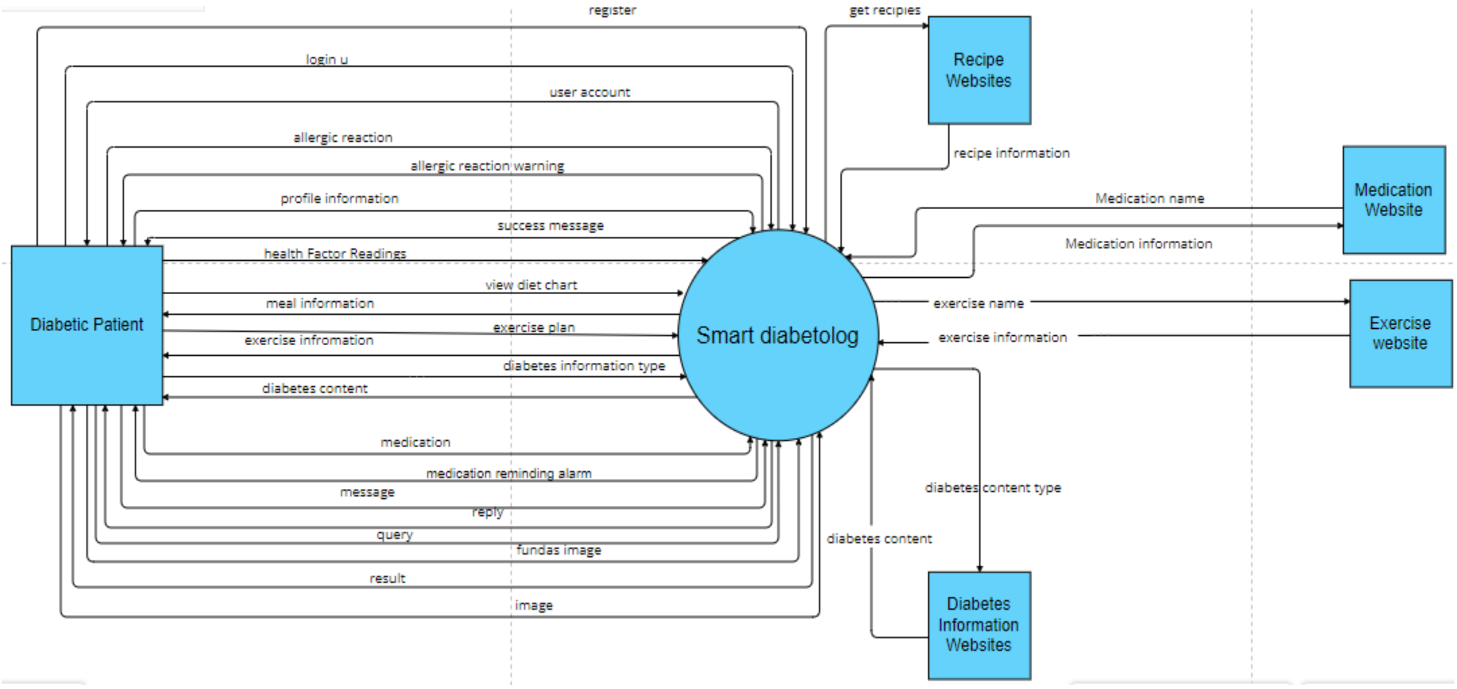


Figure 3-40: Level 0 DFD

3.2.2.2 Level 1 DFD

Figure 3-41 displays a level 1 data flow diagram of Smart Diabetolog. It represents the main processes of our system. The patient interacts with our system and perform the following processes represented in this diagram. The system in return interacts with external system to complete these processes.

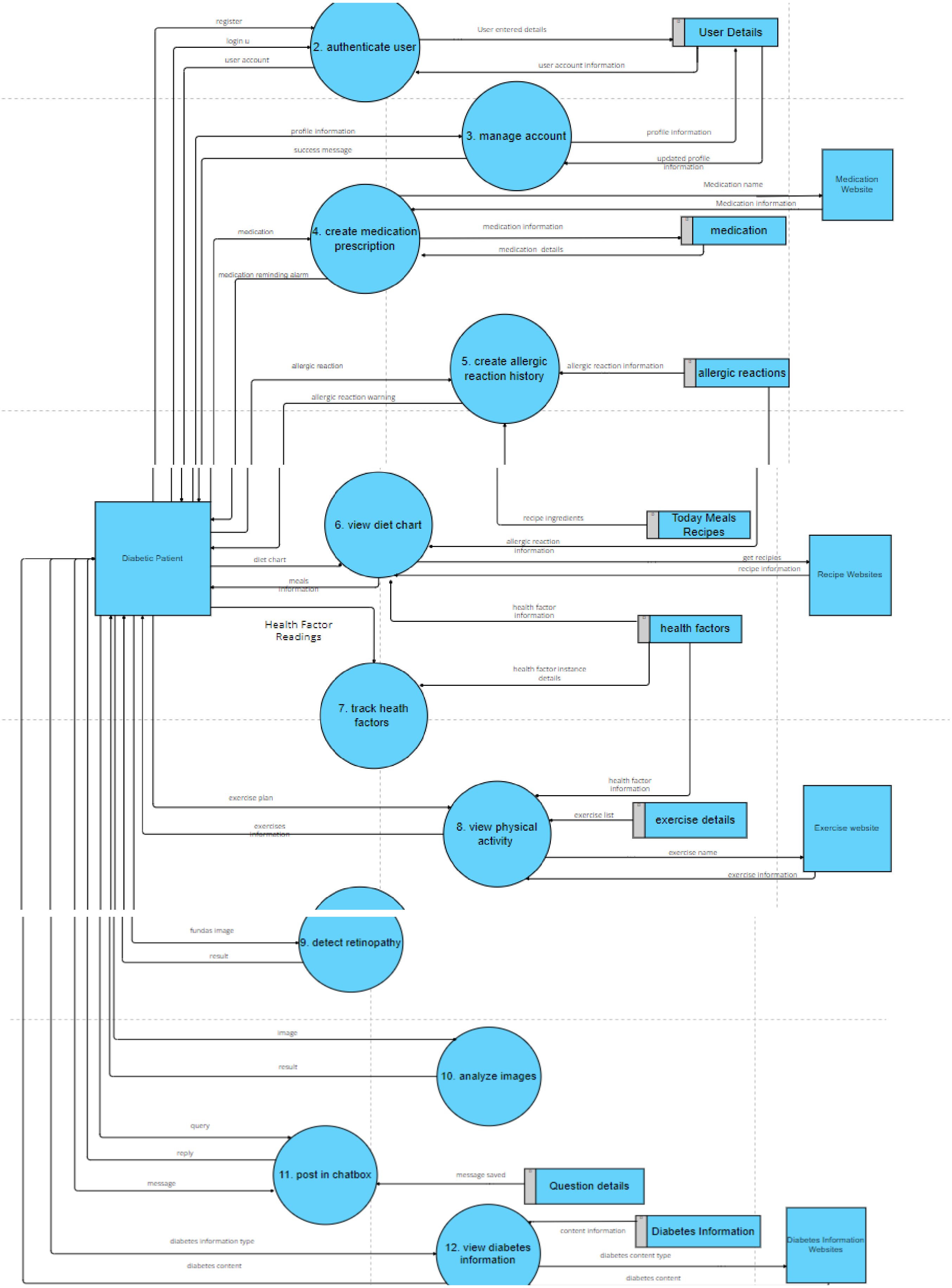


Figure 3-41: Level 1 DFD

3.2.2.3 Level 2 DFD

Figure 3-42 displays level 2 data flow diagram of Smart diabetology. In this process diabetic patients register in our application and then logging in. Our application confirms the patient by checking it details in user Details Database and then provides account access by completing the process.

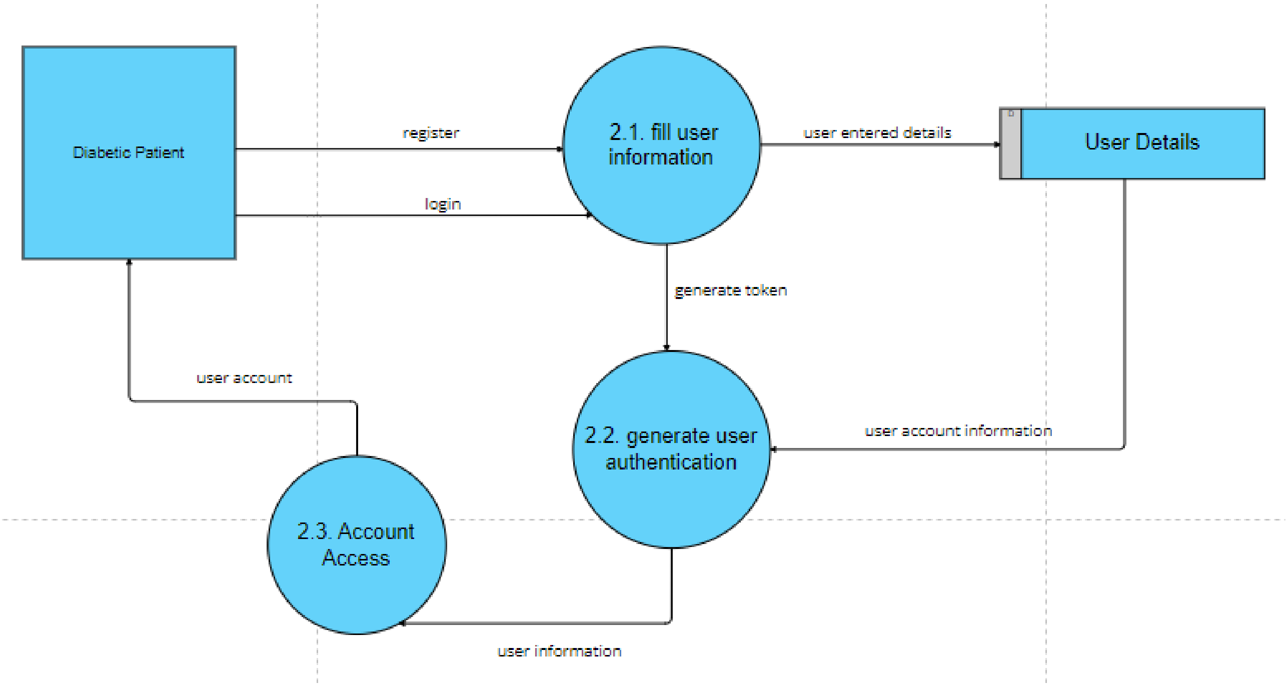


Figure 3-42: Level 2 DFD

3.2.2.4 Level 3 DFD

Figure 3-43 displays level 3 data flow diagram for the process of managing account. The diabetic patient can change his profile information and save it, or he can delete his profile. The changes made will be updated in user Details database.

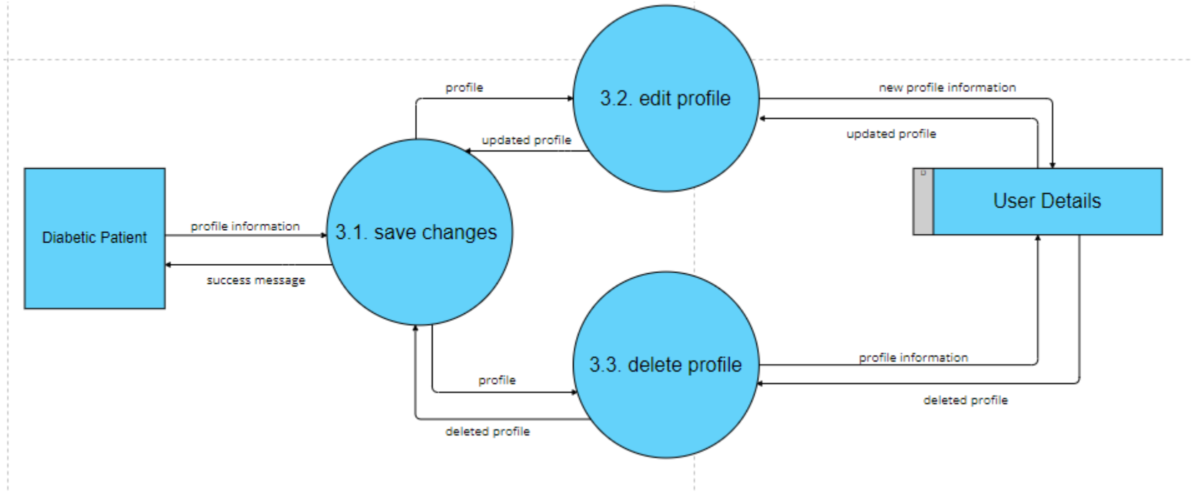


Figure 3-43: Level 3 DFD

3.2.2.5 Level 4 DFD

Figure 3-44 displays a level 4 data flow diagram for the process of creating medication prescription by choosing the type of medicine then filling the details like name, dosage and time of medicine and finally saving the medicine in medication database. The alarms according to consumption time will be automatically set.

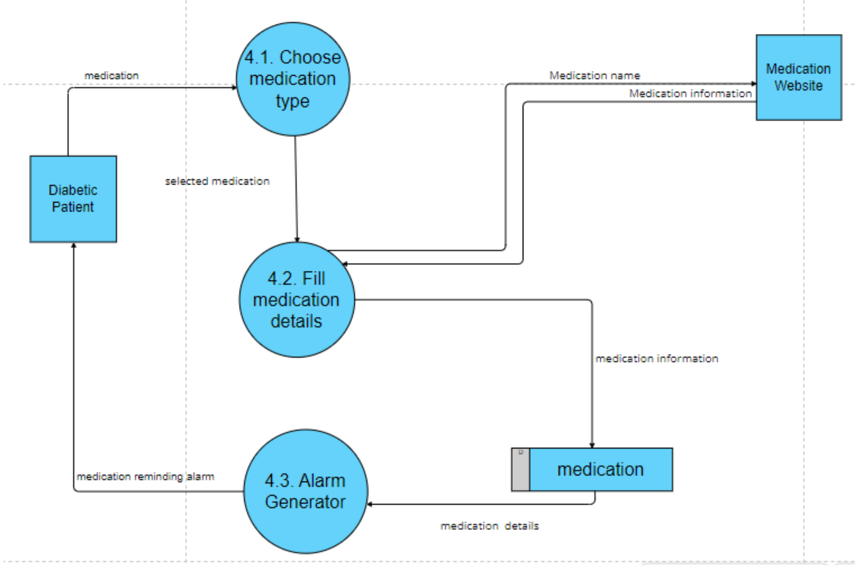


Figure 3-44: Level 4 DFD

3.2.2.6 Level 5 DFD

Figure 3-45 shows the data flow diagram for the process of saving allergic reaction. The diabetic patient will save any allergy he has with medication or food by creating an allergic reaction in allergic reaction database. This is automatically show warnings when patient will try to make a medication prescription with any allergic agent or try to consume any food with allergic reaction.

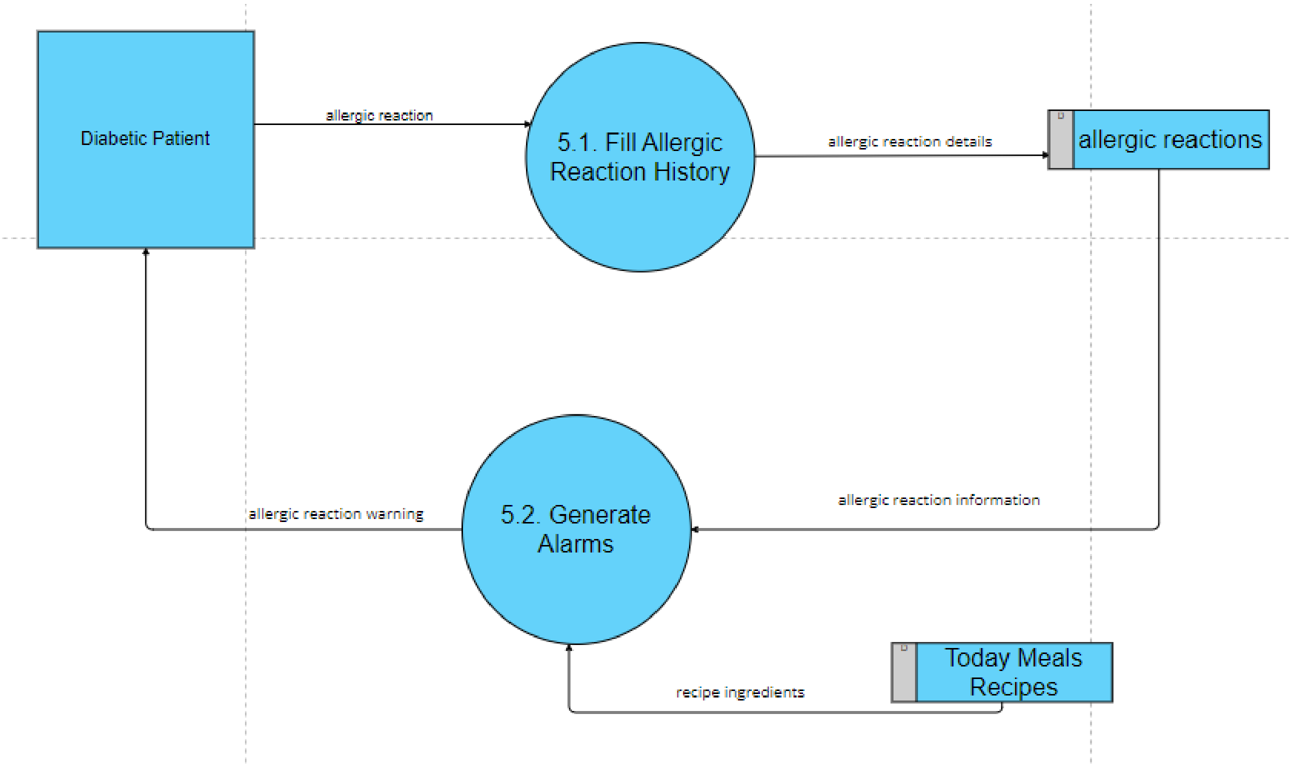

Diabetic Patient
allergic reaction
5.1. Fill Allergic Reaction History
allergic reaction details
allergic reactions
allergic reaction warning
5.2. Generate Alarms
allergic reaction information
recipe ingredients
Today Meals Recipes

Figure 3-45: Level 5 DFD

3.2.2.7 Level 6 DFD

Figure 3-46 shows the data flow diagram for the process of viewing meal plan. The diabetic patient initiates the process of viewing meal plan. The meal plan will be created by keeping in view his health factor details, his allergic reaction details. According to his calories count the recipes will be fetched for external interfaces and a customized meal plan will be created.

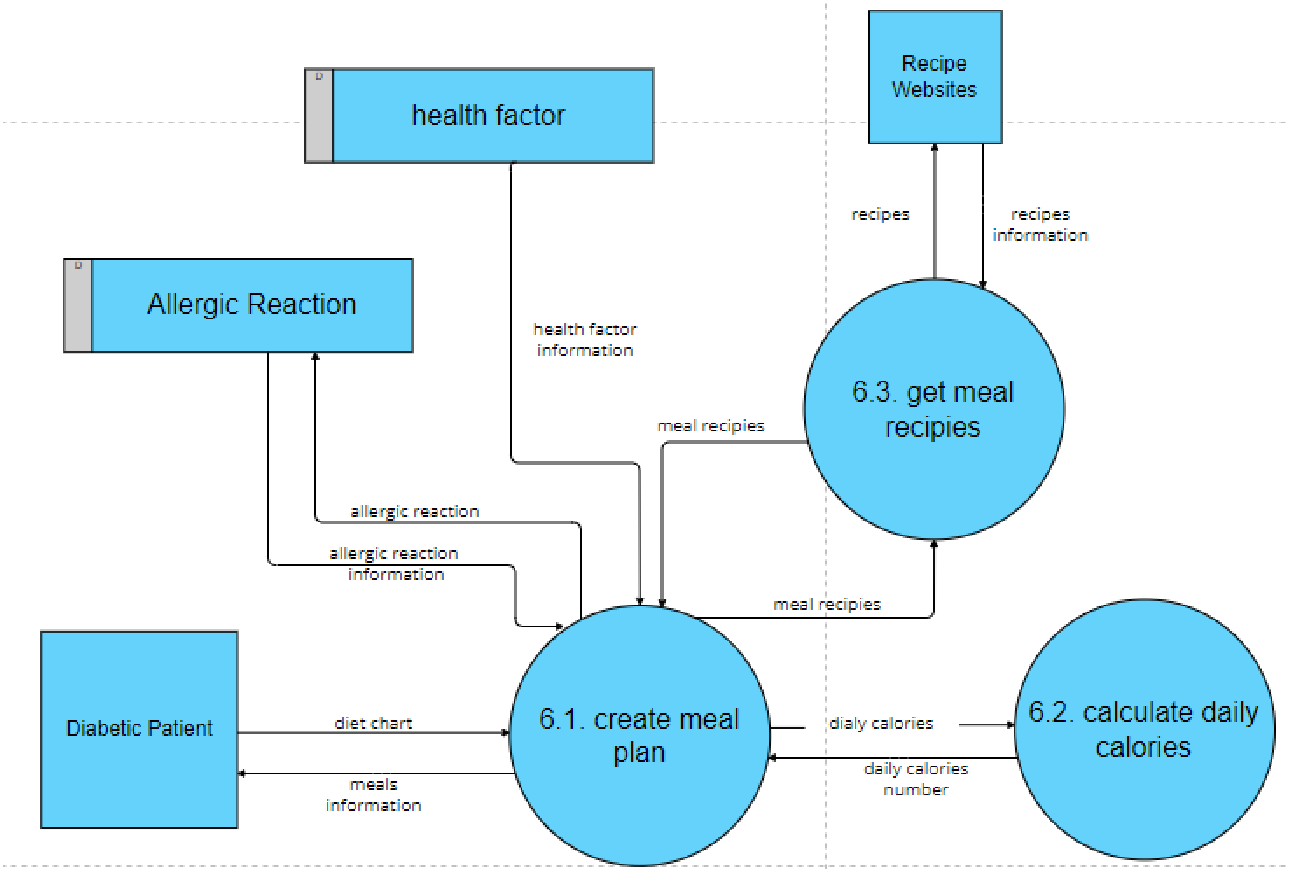


Figure 3-46: Level 6 DFD

3.2.2.8 Level 7 DFD

Figure 3-47 shows the data flow diagram for the process of customizing health factors. The diabetic patient will be allowed to select the health factors beside blood sugar which he wants to provide. On these selected factors his future plans will be created. The diabetic patient can customize blood pressure and cholesterol.

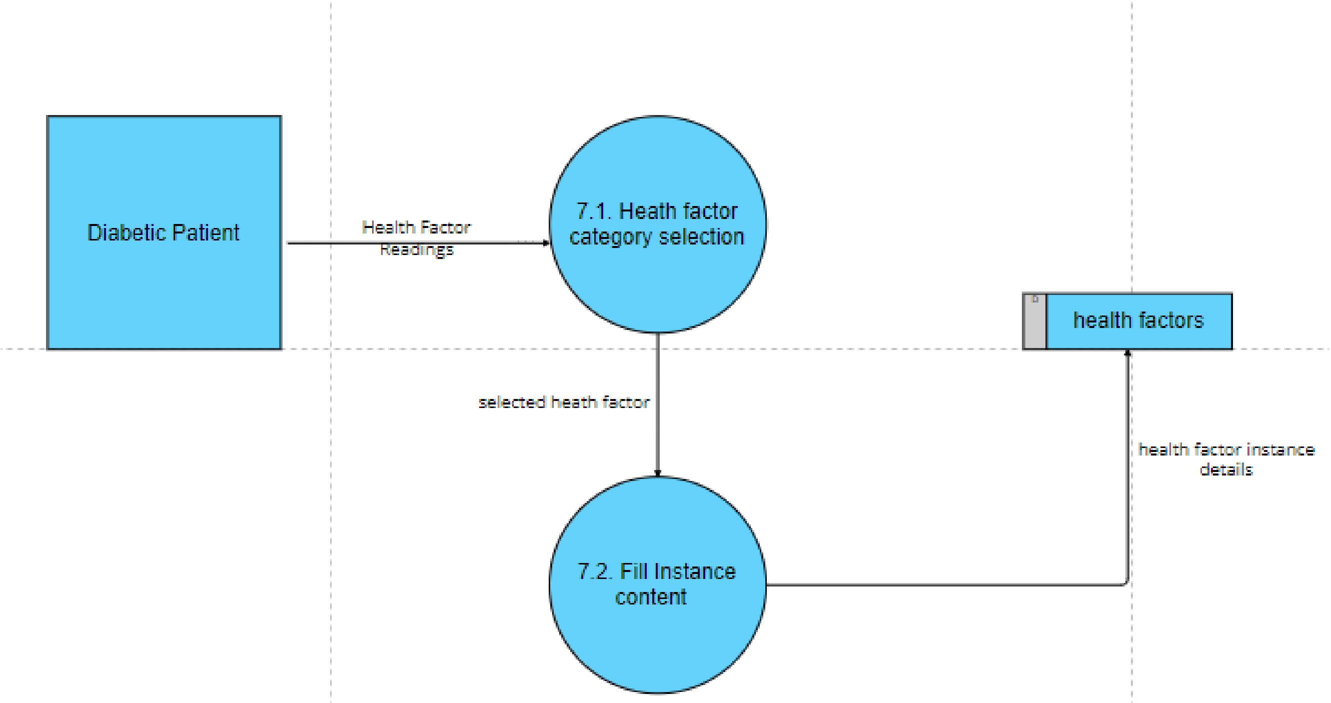

Diabetic Patient
Health Factor Readings
7.1. Heath factor category selection
health factors
selected heath factor
health factor instance details
7.2. Fill Instance content

Figure 3-47: Level 7 DFD

3.2.2.9 Level 8 DFD

Figure 3-48 shows data flow diagram for the process of viewing exercise plan. The process includes creating exercise plan by finding out the number of calories to burn according to patients' health factors. After getting these calories suitable exercise will be scrapped from external interface and customized exercise plan according to patient needs will be made and displayed to patient.

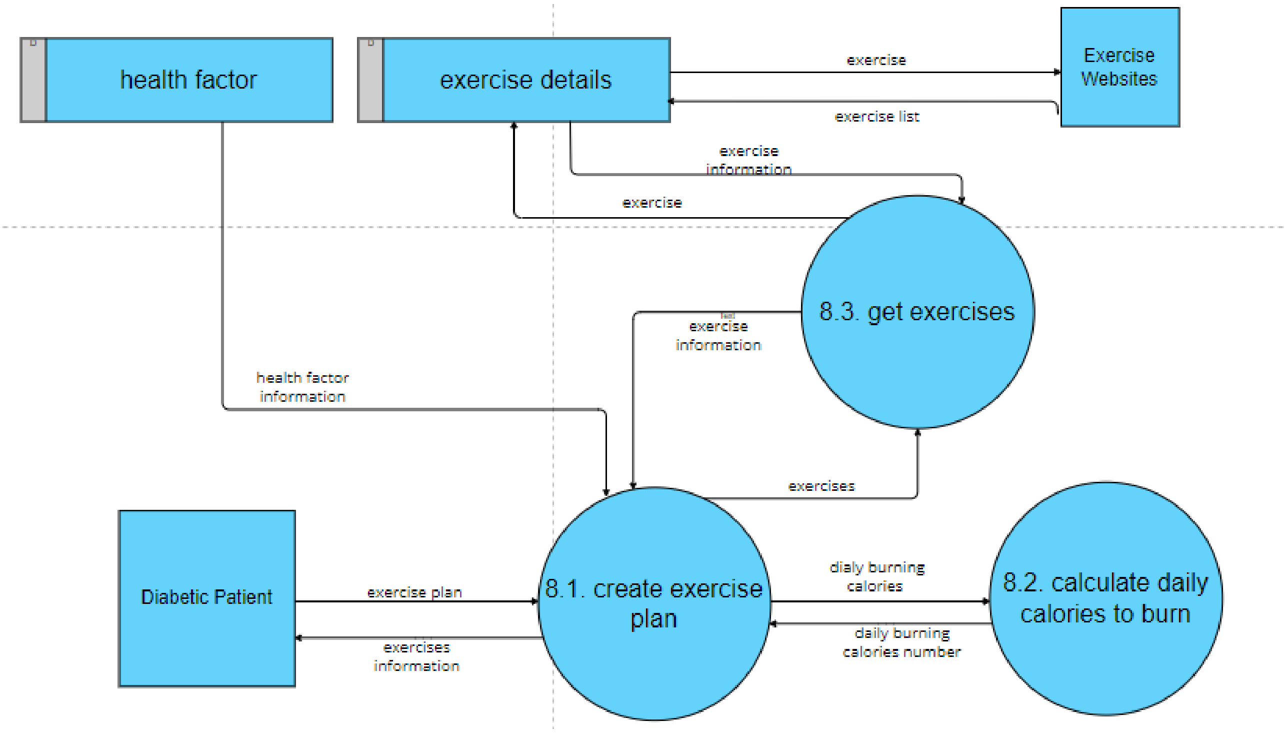


Figure 3-48: Level 8 DFD

3.2.2.10 Level 9 DFD

Figure 3-49 shows the data flow diagram for process of detecting retinopathy. The patient uploads fundus image which is processed to predict retinopathy after applying suitable model the retinopathy is detected and classified according to the image. The result is then passed back to the patient.

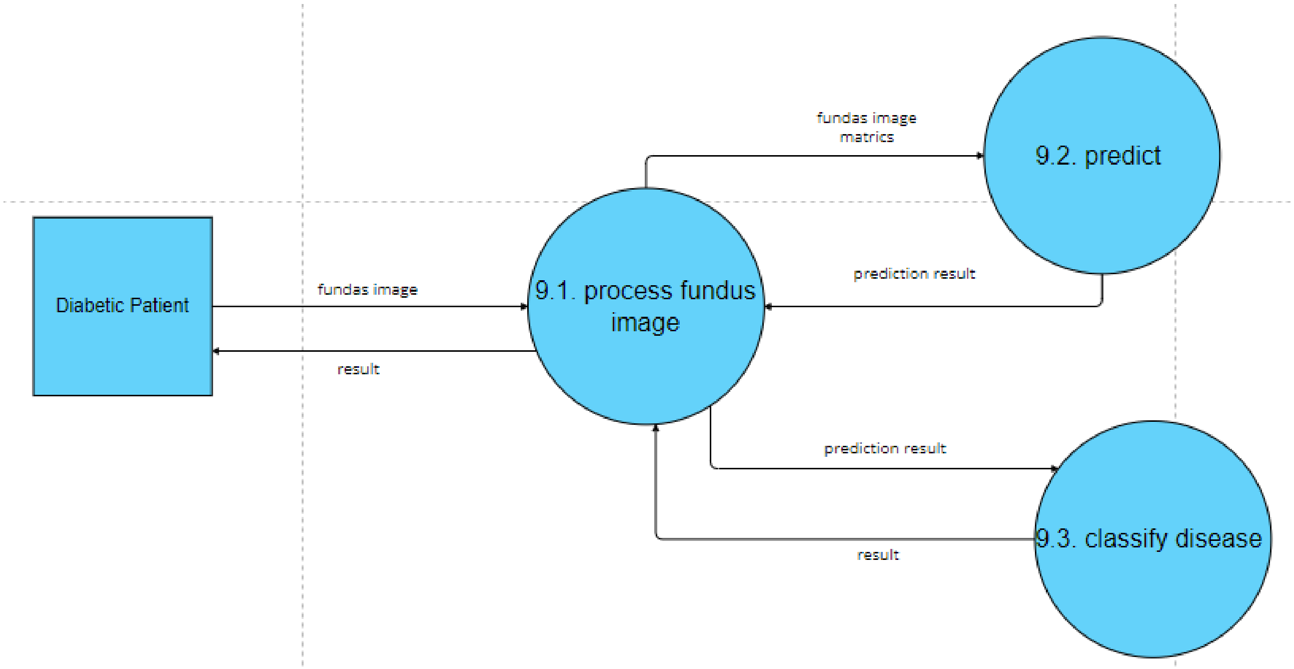


Figure 3-49: Level 9 DFD

3.2.2.11 Level 10 DFD

Figure 3-50 displays the data flow diagram for the process of image processing of the image for meal or for medication name. the meal image will be processed and then segmented to get dish name and after that predicting dish calories. While the medication image will be passed to optical character recognition to recognize name of medication from the image.

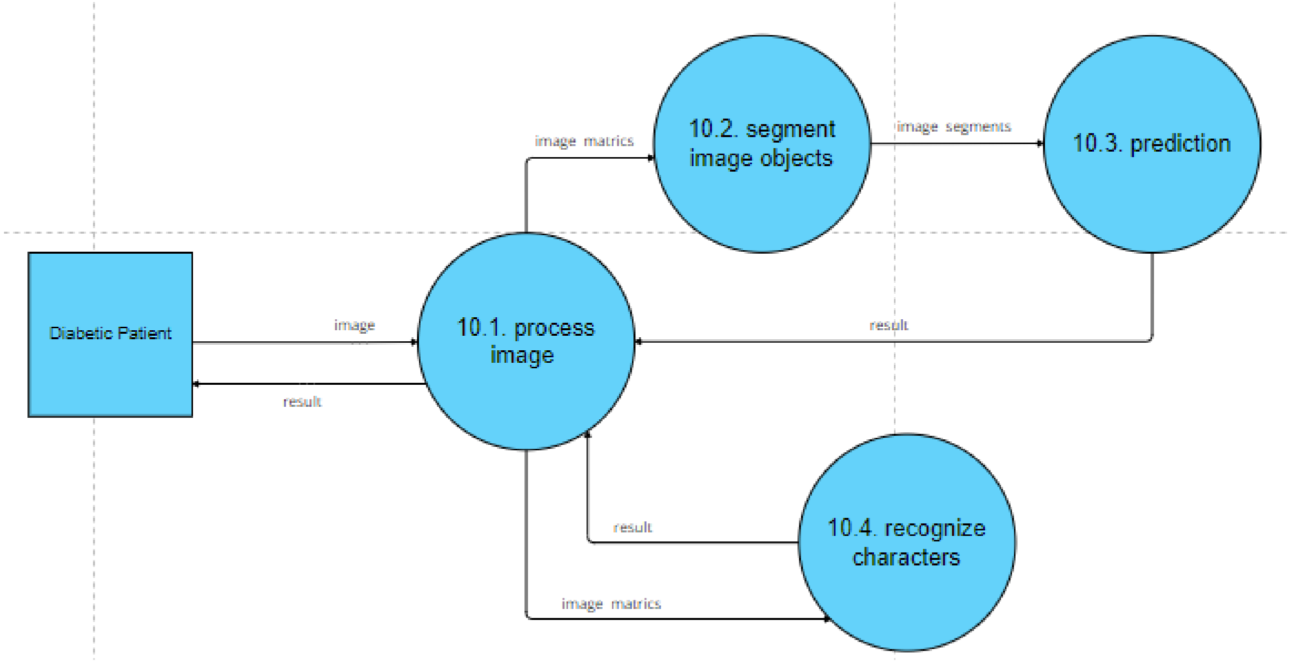


Figure 3-50: Level 10 DFD

3.2.2.12 Level 11 DFD

Figure 3-51 shows the data flow diagram for the process of posting in discussion board. The diabetic patient can post on a discussion board in two ways. The first way is by sending messages in live chat. The patient can also post a query in the query board where other patient can read that query and reply if they have answers to that query.

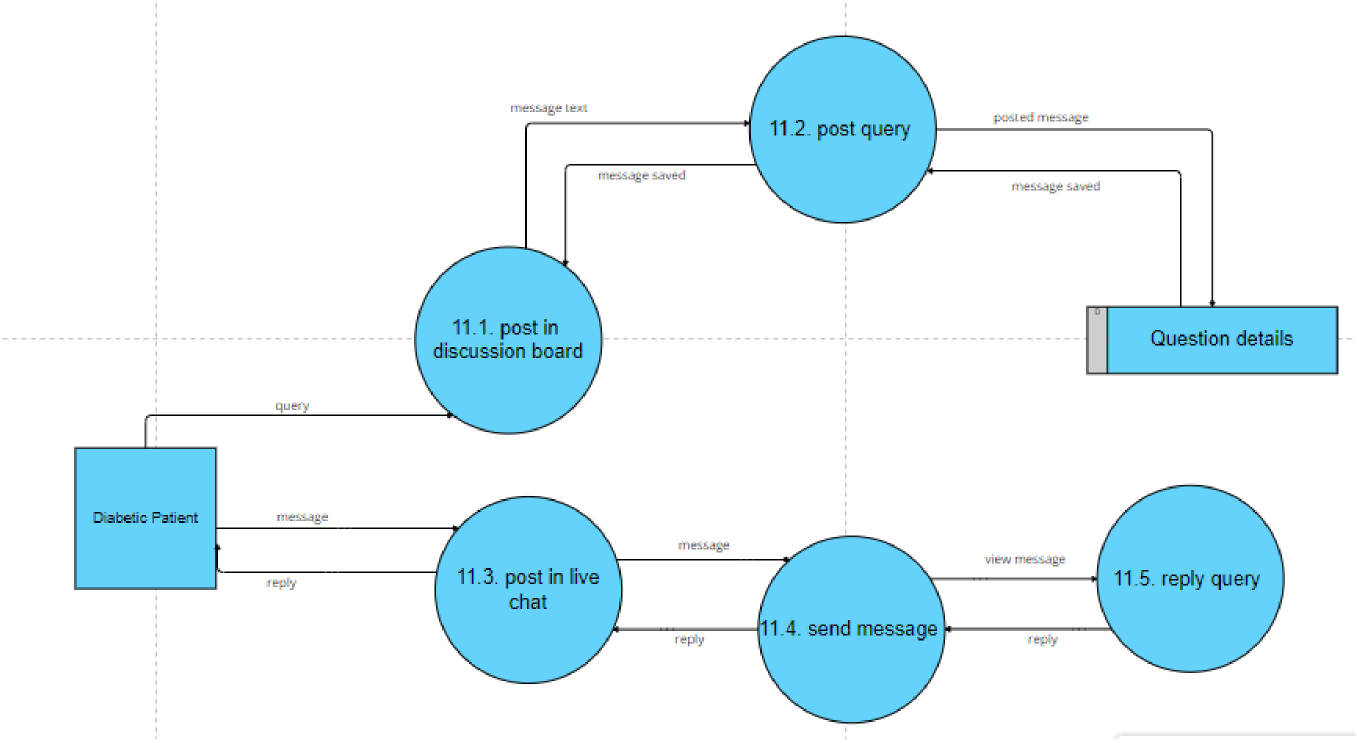


Figure 3-51: Level 11 DFD

3.2.2.13 Level 12 DFD

Figure 3-52 shows the data flow diagram for the process of viewing diabetes related information. The diabetic patient requests the information which will be scrapped from the external interfaces by the system. This information will then be displayed to the patient as a response to its request.

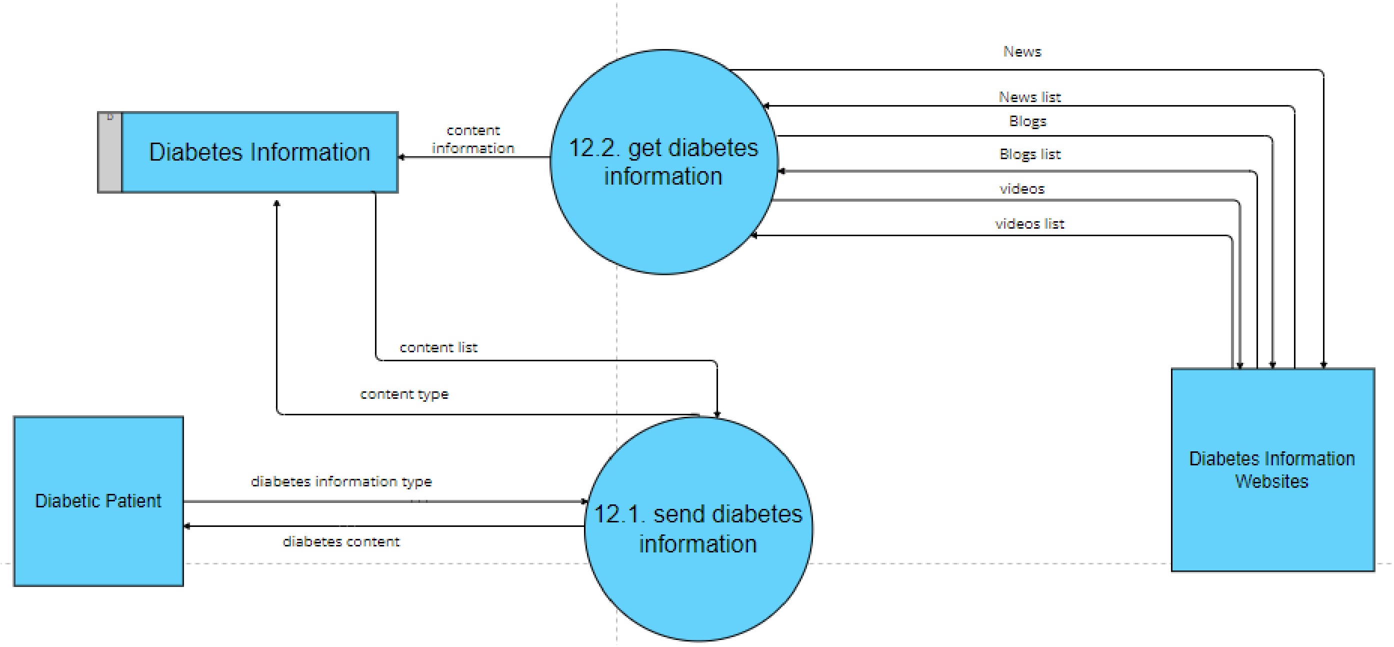


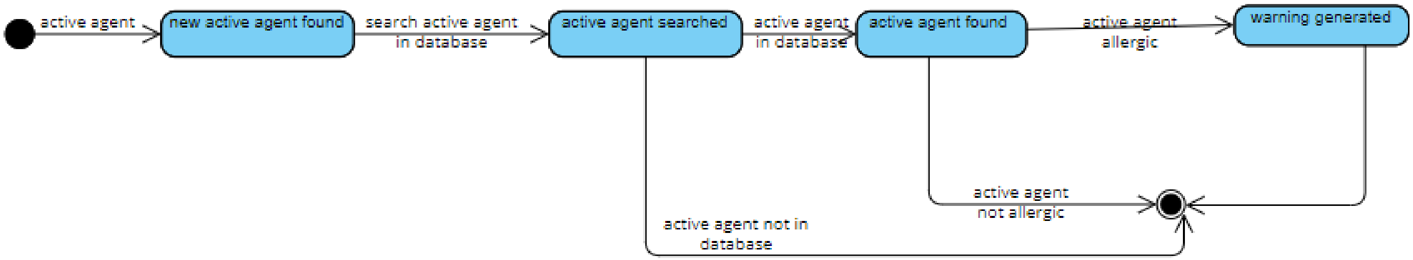
Figure 3-52: Level 12 DFD

3.2.3 Sequence Diagram

3.2.3.1 Newly added medication has active agent with allergy

Figure 3-53 contains sequence diagram for whenever a newly added medication has active agents, user have allergy with.

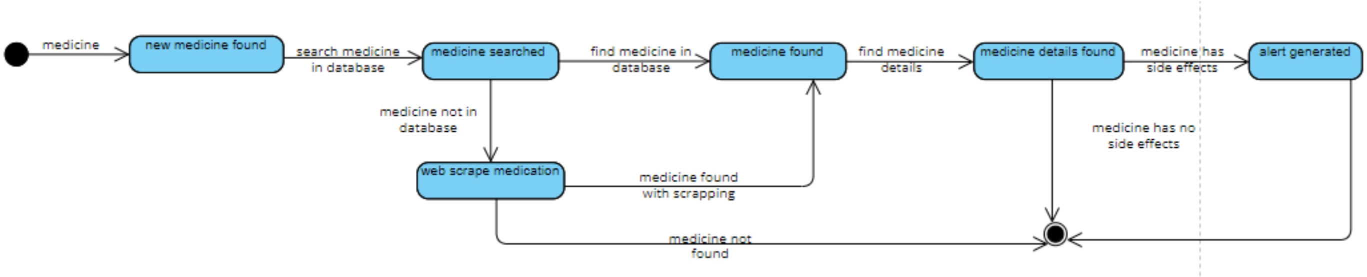
Figure 3-53: SD-1 Newly added medication has active agent with allergy



3.2.3.2 Newly added medication has the probability of side effects

Figure 3-54 contains sequence diagram for whenever a newly added medication has the probability of side effects

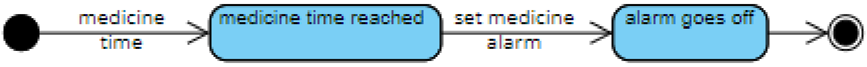
Figure 3-54: SD-2 newly added medication has the probability of side effects



3.2.3.3 medication intake time is reached

Figure 3-55 contains sequence diagram for when the medication intake time is reached.

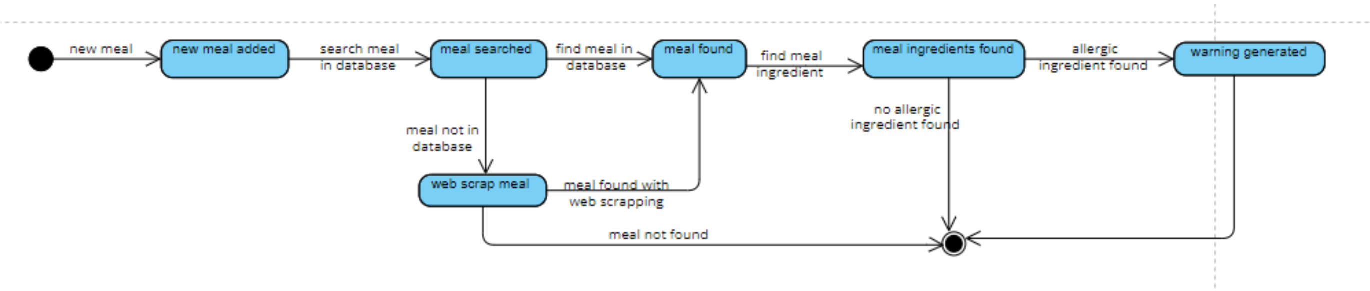
Figure 3-55: SD-3 medication intake time is reached.



3.2.3.4 *Newly added meal has ingredients, user has allergy with*

Figure 3-56 contains sequence diagram for whenever a newly added meal has ingredients, user has allergy with.

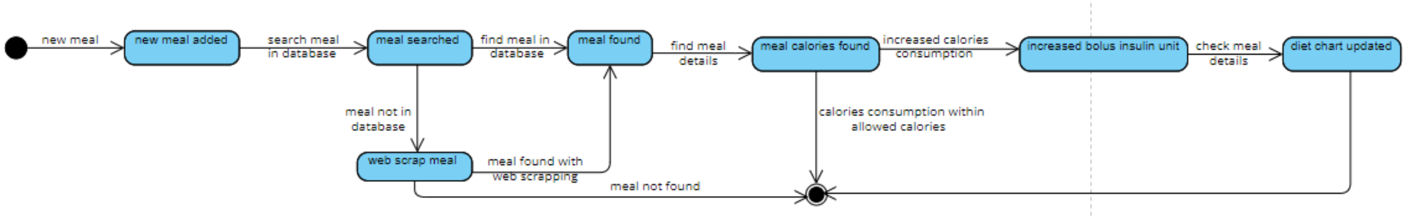
Figure 3-56: SD-4 newly added meal has ingredients; user has allergy with.



3.2.3.5 *Meal with more than allowed calories:*

Figure 3-57 contains sequence diagram for when the user intakes a meal with more than allowed calories

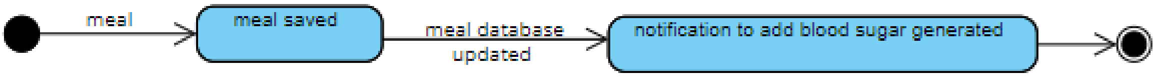
Figure 3-57: SD-5 meal with more than allowed calories



3.2.3.6 *User saves a meal intake*

Figure 3-58 contains sequence diagram for whenever the user saves a meal intake.

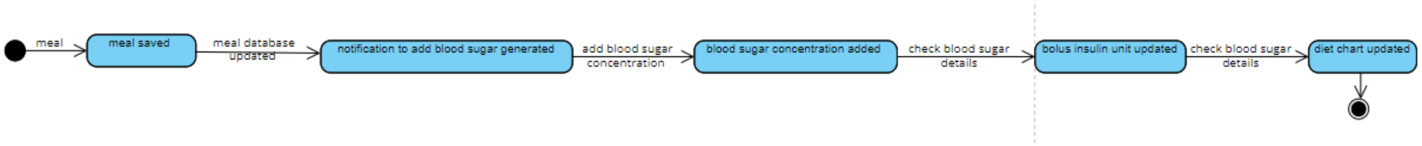
Figure 3-58: SD-6 user saves a meal intake.



3.2.3.7 *New blood sugar instance with after meal event*

Figure 3-59 contains sequence diagram for whenever the user adds a new blood sugar instance with after meal event.

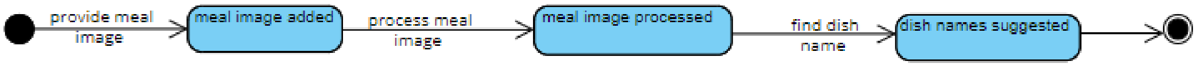
Figure 3-59: SD-7 new blood sugar instance with after meal event.


meal
meal saved
meal database updated
notification to add blood sugar generated
add blood sugar concentration
blood sugar concentration added
check blood sugar details
bolus insulin unit updated
check blood sugar details
diet chart updated

3.2.3.8 *Adds a meal consumption through image*

Figure 3-60 contains sequence diagram for whenever the user adds a meal consumption through image

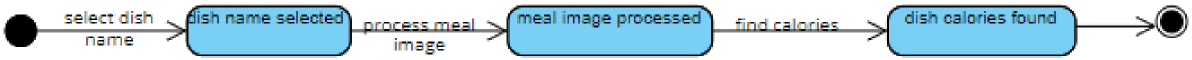
Figure 3-60: SD-8 adds a meal consumption through image



3.2.3.9 *select a dish name from the food options after taking food image*

Figure 3-61 contains sequence diagram for whenever a user selects a dish name from the food options after taking food image.

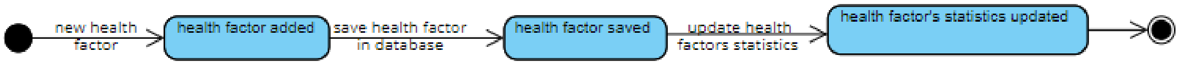
Figure 3-61: SD-9 selects a dish name from the food options after taking food image



3.2.3.10 *User adds a new health factor*

Figure 3-62 contains sequence diagram for whenever the user adds a new health factor (blood, sugar, blood pressure or cholesterol) instance.

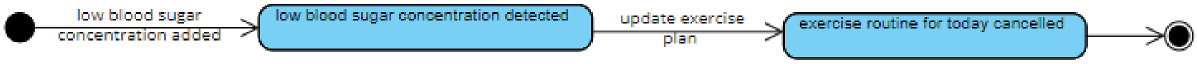
Figure 3-62: SD-10 user adds a new health factor


new health factor
health factor added
save health factor in database
health factor saved
update health factors statistics
health factor's statistics updated

3.2.3.11 blood sugar average is low for the day

Figure 3-63 contains sequence diagram for when the blood sugar average is low for the day.

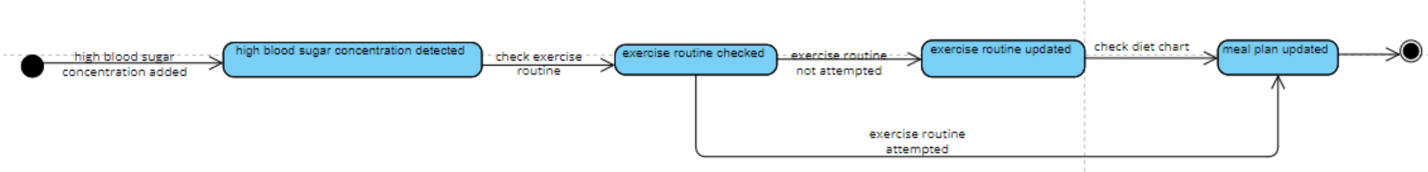
Figure 3-63: SD-11 blood sugar average is low for the day.


low blood sugar concentration added
low blood sugar concentration detected
update exercise plan
exercise routine for today cancelled

3.2.3.12 exercise routine was incomplete and blood-sugar average was also high for the day

Figure 3-3 contains sequence diagram for when the exercise routine was incomplete and blood-sugar average was also high for the day.

Table 3-3: SD-12 sugar level and exercise routine


high blood sugar concentration added
high blood sugar concentration detected
check exercise routine
exercise routine checked
exercise routine not attempted
exercise routine updated
check diet chart
meal plan updated
exercise routine attempted

3.2. Data Design

The database used for our project would be MongoDB. It is a non-relational database. MongoDB will be used to store the data with it being hosted on MongoDB Atlas.

3.2.4 Data Dictionary

3.2.4.1 Allergic Reaction

This scheme saves the allergic reaction details of the patient.

```
allergicReaction_id: String
name:String
type: String
symptoms:String[]
description:String
user_id: String
```

3.2.4.2 Blood Pressure

This schema saves the blood pressure details of the patient.

```
bloodPressure_id : String
systolic: Integer
diastolic: Integer
date: Date
description: String
user_id: String
```

3.2.4.3 Blood Sugar

This schema saves the blood glucose concentration of the patient.

```
bloodSugar_id : String
concentration: Integer
unit: Enum[mmol/L, mg/dL]
description: String
time: Time
date: Date
event: String
description: String
creationDate: Date
creationTime: Time
user_id: String
```

3.2.4.4 Cholesterol

This schema saves the cholesterol details of the patient.

```
cholesterol_id: String
cholesterol: Integer
ldl: Integer
hdl: Integer
triglycerides: Integer
description: String
user_id: String
cholesterol: String
bloodSugar: bloodSugar_id
```

3.2.4.5 FastInsulin

This schema contains the insulin medication information of a patient.

```
fastinsulin_id: String
isf: String
carb_ratio: String
user_id: String
```

3.2.4.6 LongInsulin

This schema contains the insulin medication information of a patient.

```
Longinsulin_id: String
name: String
unit: String
time: String
user_id: String
```

3.2.4.7 Oral Medication

This schema contains the oral medication information of a patient.

```
oralMedicine_id: String
name:String
dosage:Integer
unit:Integer
time: Time[]
user_id: String
```

3.2.4.8 Prescription

```
id: String
title: String
oralMedication: String
user_id: String
longInsulin:String
fastInsulin: String
```

3.2.4.9 User

This schema contains all the personal information of a patient which will be shown on his profile.

```
user_id: String
name: String
```

```
email: String
password: String
weight: Integer
height: Integer
activityLevel: String
```

3.2.4.10 Messages

This schema contains messages information send and received.

```
_id: String
text: String
user: Object{
      user_id,
      name
}
```

3.2.4.11 Dashboard

```
id: String
title: String
detail: String
userAsked: UserId
answers: object{
      user_id,
      answer
}
```

3.2.4.12 OTP schema

```
Id: String
Code: String
User_id: String
```

3.3. Human Interface Design

The following section contains human interface designs of Smart Diabetolog.

3.3.2. Screen Images

Figure 3-65 contains the login screen.

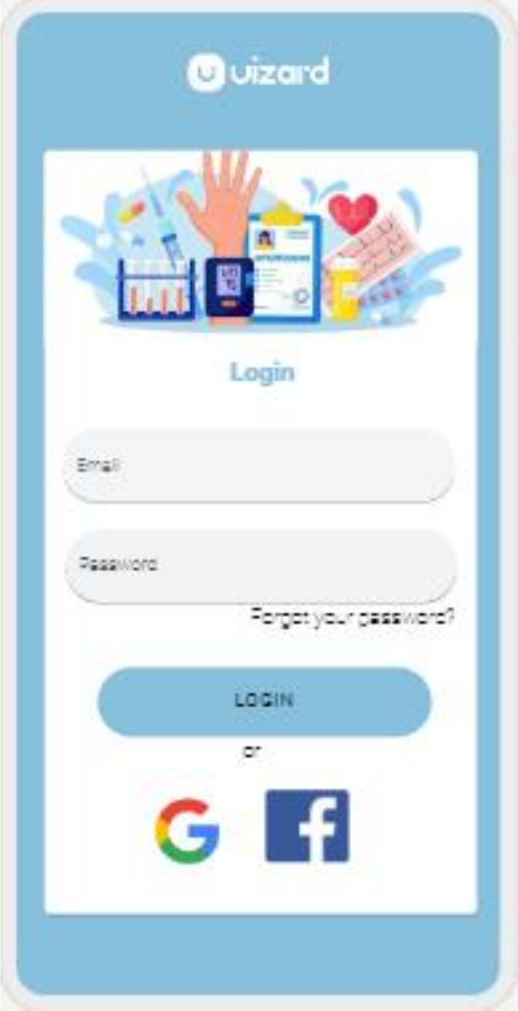


Figure 3-65: Login screen

Figure 3-64 contains the register screen.

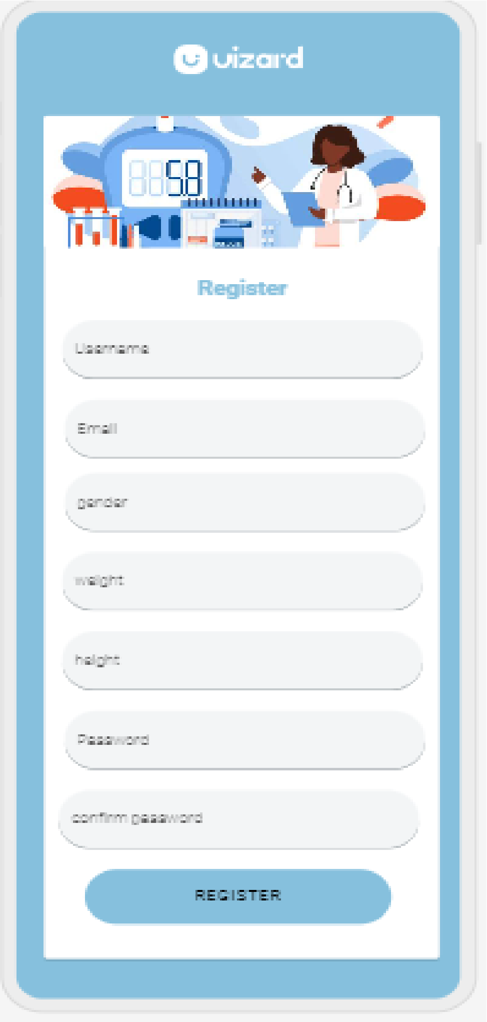


Figure 3-64: Register screen

figure 3-66 contains the home screen.

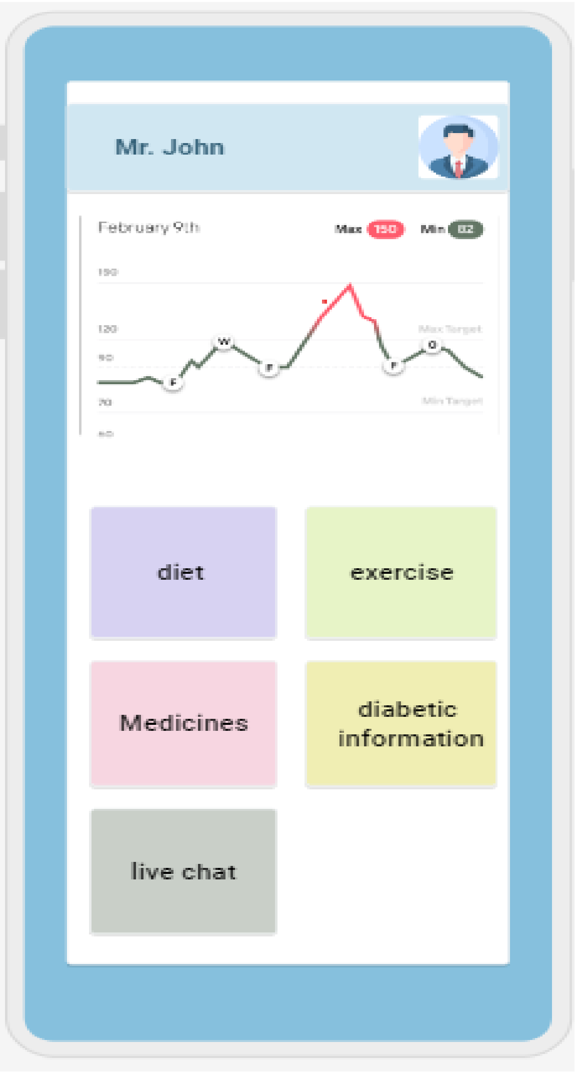


Figure 3-66: Home

Figure 3-67 contains the medication main screen.

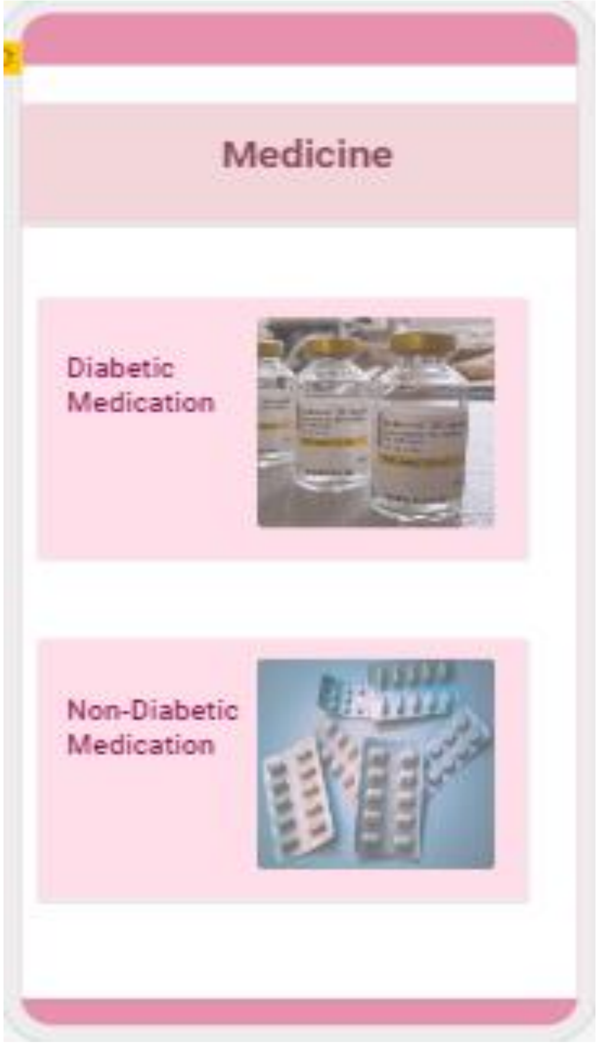


Figure 3-67: Medicine

Figure 3-69 contains the exercise main screen.

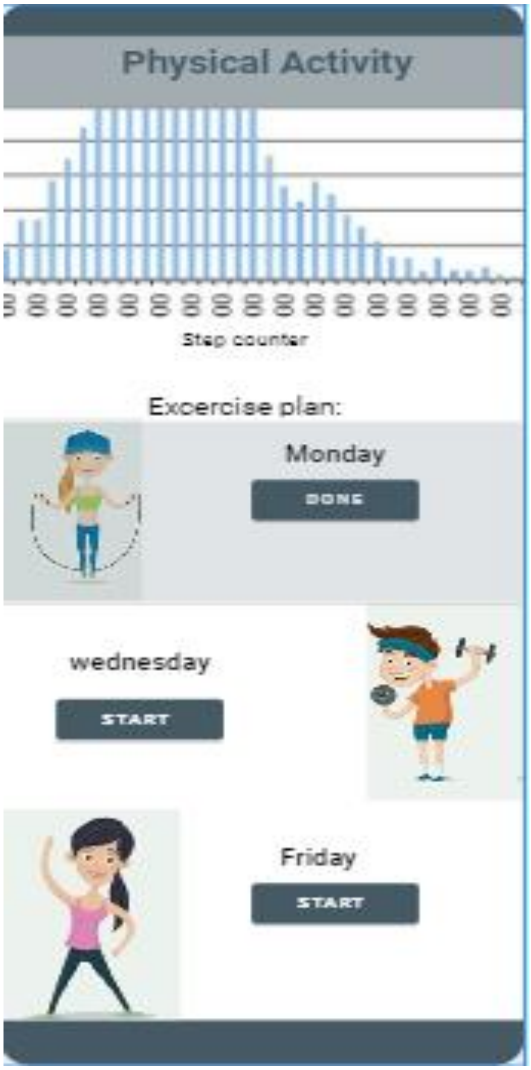
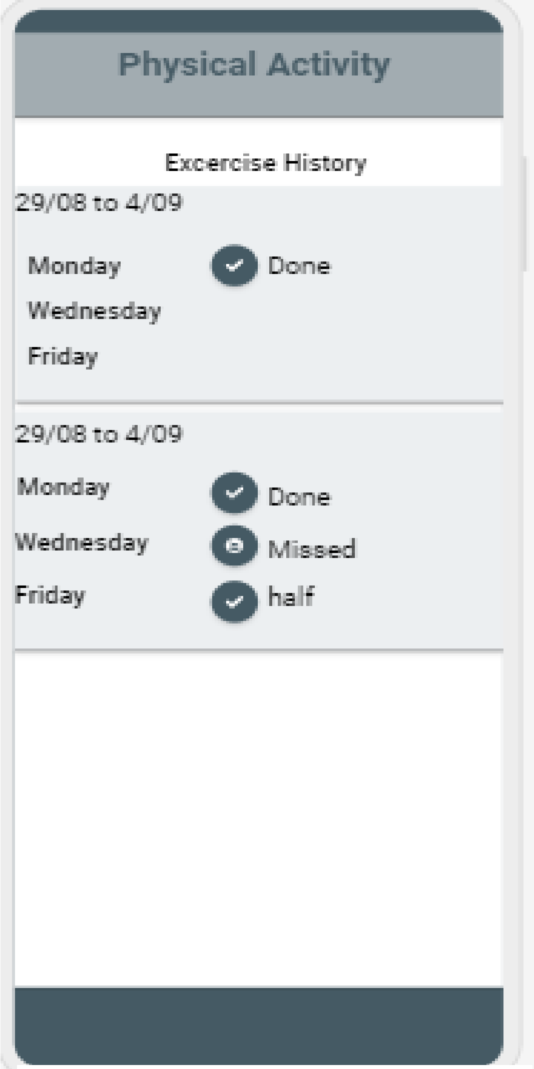


Figure 3-69: physical activity

Figure 3-68 contains the history of exercises.



Excercise History

29/08 to 4/09

Monday ✓ Done

Wednesday

Friday

29/08 to 4/09

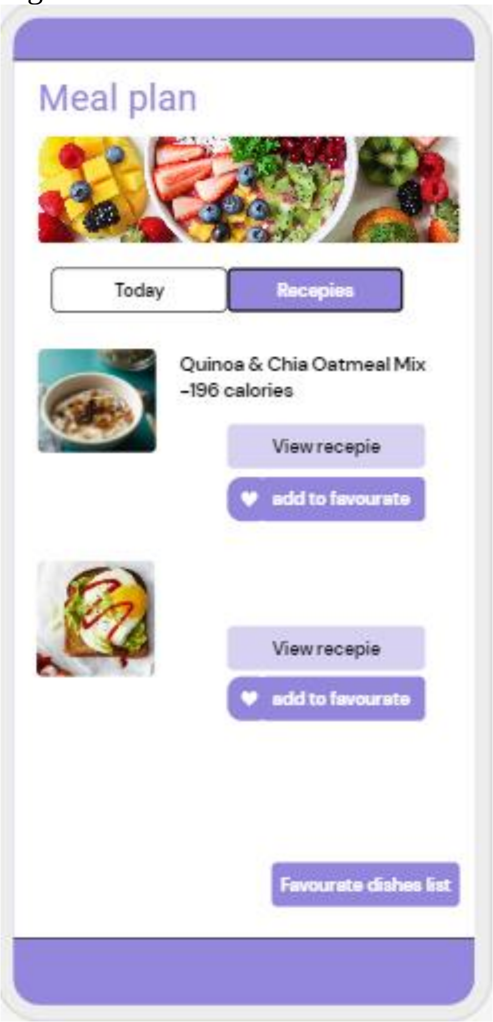
Monday ✓ Done

Wednesday ⊖ Missed

Friday ✓ half

Figure 3-68: physical activity Screen

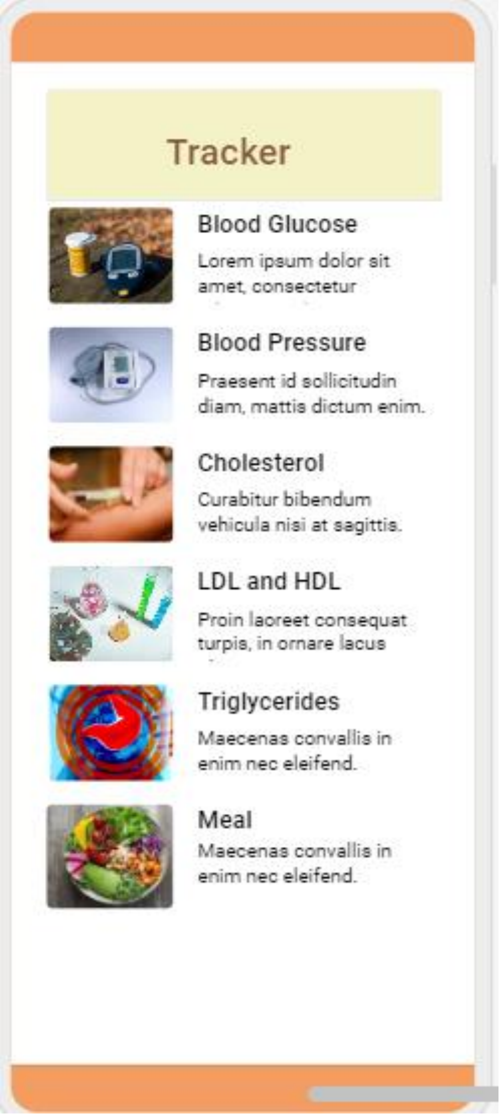
Figure 3-70 contain diet chart main screen.



Quinoa & Chia Oatmeal Mix
-196 calories

Figure 3-70: Meal Plan Screen

Figure 3-72 contains tracker main screen.



Tracker

Blood Glucose
Lorem ipsum dolor sit amet, consectetur

Blood Pressure
Praesent id sollicitudin diam, mattis dictum enim.

Cholesterol
Curabitur bibendum vehicula nisi at sagittis.

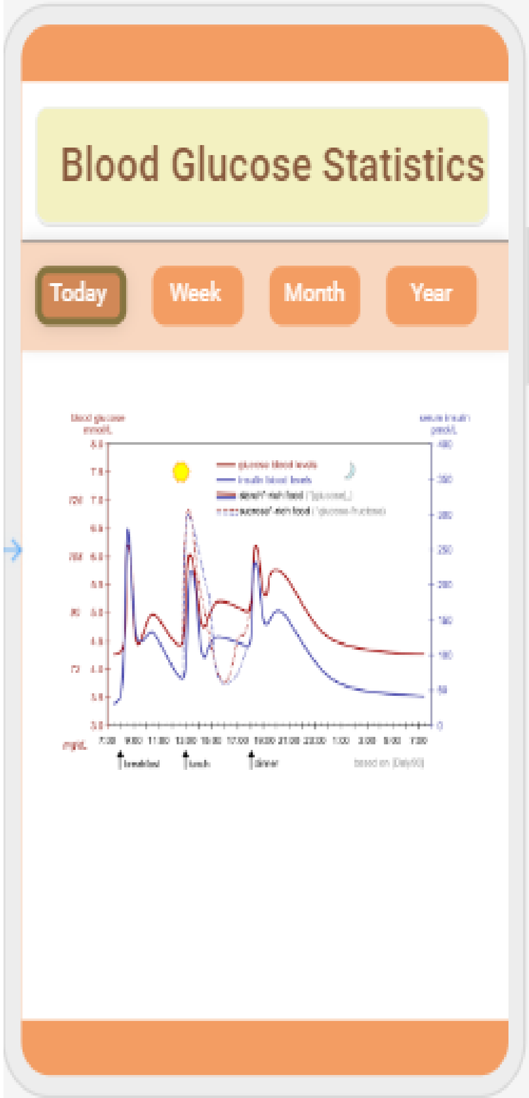
LDL and HDL
Proin laoreet consequat turpis, in ornare lacus

Triglycerides
Maecenas convallis in enim nec eleifend.

Meal
Maecenas convallis in enim nec eleifend.

Figure 3-72: Tracker Main Screen

Figure 3-71 contains statics of blood sugar.



Blood Glucose Statistics

Figure 3-71: Blood Sugar Statics Screen

3.3.3. Screen Objects and Actions

A discussion of screen objects and actions associated with those objects.

Register screen: The register button registers the user with the given details into the system.
Login screen: The login button signs the user into the system.
Home: The medications, allergic reaction, diet, exercise, chat buttons take the user to the main screen of those modules.
Medicine: The diabetic medications take the user to the diabetic medication screen.
Medicine: The non-diabetic medications take the user to the non-diabetic medication screen.
physical activity: The start button initiates the physical activity plan for the day.
Allergic Reaction Screen: The diabetic medication displays all the diabetic allergic reactions stored in the system.
Non-diabetic Medication: The non-diabetic medication displays all the non-diabetic allergic reactions stored in the system.
Allergic Reaction Screen: The insulin displays all the insulin allergic reactions stored in the system.
Allergic Reaction Screen: The food displays all the food allergic reactions stored in the system.
Meal Plan Screen: The view recipe button displays the recipe of the food.
Meal Plan Screen: The add to favorite button add the recipe to the favorite
Meal Plan Screen: The view favorite list button displays the list of the user’s favorite recipes.
Tracker Main Screen: The blood glucose displays all blood glucose instances of the day stored by user.
Tracker Main Screen: The blood pressure displays all blood pressure instances of the eek stored by user.
Tracker Main Screen: The cholesterol displays all cholesterol instances of 6 months stored by user.
Blood Sugar Statics Screen: The today button displays the statistics of all today’s blood sugar instances.
Blood Sugar Statics Screen: The week button displays the statistics of all blood sugar instances of the week.
Blood Sugar Statics Screen: The month button displays the statistics of all blood sugar instances of the month.
Blood Sugar Statics Screen: The year button displays the statistics of all blood sugar instances of the year.

4. Implementation

The following section contains the implementation information of major modules of Smart Diabetolog Application.

4.1. Algorithm

Algorithm 1
MakeDietChart

Input: calories

Output: A diet chart for the day in the form of array

```
#importing the recipes file
df=pd.read_csv('formatedRecipes.csv')
dfForRD=pd.read_csv('formatedRecipes.csv')
#transforming the name
le=LabelEncoder()
df["nameEncoded"]=le.fit_transform(df["name"])
dfForRD["nameEncoded"]=le.fit_transform(dfForRD["name"])
def dietPlan(calories):
 #get calories for each meal
    caloriesForBreakfast=calories*25/100
    caloriesForSnack1=calories*10/100
    caloriesForLunch=calories*30/100
    caloriesForSnack2=calories*10/100
    caloriesForDinner=calories*25/100
   print(caloriesForBreakfast,caloriesForSnack1,caloriesForLunch,caloriesForSnack2,caloriesForDinner)
    #dietPlan contains the arrays of all the meals
    dietPlan=[]
     i=1
     while(i<=5):
       recipe=[]
      if(i==1):
        carbs=getCarbs(caloriesForBreakfast)
        recipe=getRecipe(caloriesForBreakfast,carbs[0])
      elif(i==2):
        carbs=getCarbs(caloriesForSnack1)
        recipe=getRecipe(caloriesForSnack1,carbs[0])
      elif(i==3):
        carbs=getCarbs(caloriesForLunch)
        recipe=getRecipe(caloriesForLunch,carbs[0])
      elif(i==4):
        carbs=getCarbs(caloriesForSnack2)
        recipe=getRecipe(caloriesForSnack2,carbs[0])
      elif(i==5):
        carbs=getCarbs(caloriesForDinner)
        recipe=getRecipe(caloriesForDinner,carbs[0])
       dietPlan.append(recipe)
       print("diet plan is-----------------------")
       print(dietPlan)
       i+=1
```

Algorithm 1.1

getRecipe

Input: calories, carbs

Output: return a recipe based on carbs and calories

```
def getRecipe(calories,carbs):
  XRD=dfForRD[["kCal","carbs"]]
YRD=dfForRD["nameEncoded"]
#predict using model
model=RandomForestClassifier(n_estimators=100)
#training the model
model.fit(XRD.values,YRD.values)
#predicting
predict=model.predict([[calories,carbs]])
print(predict)
recipeDetails = dfForRD[dfForRD['nameEncoded'] == predict[0]]
#remove the predicted recipe from random forest dataset
#dfForRD=dfForRD.loc[dfForRD['nameEncoded'] != predict[0] ]
dfForRD.drop(dfForRD[dfForRD['nameEncoded'] == predict[0]].index,inplace=True)
#change 2D array to 1D
print(recipeDetails.to_numpy()[0])
return recipeDetails.to_numpy()[0]
```

Algorithm 1.2

getCarbs

Input: calories

Output: carbs

```
def getCarbs(calories):
# Define the independent variables (kCal,carbs,sugar) and dependent variable (nameEncoded)
X = df.loc[:,["kCal"]]
Y = df.loc[:,"carbs"] # Example output data
# Create an instance of the LinearRegression class and fit the model to the data
model = LinearRegression().fit(X.values, Y.values)
# Predict the output for a new set of independent variables
predicted= model.predict([[calories]])
print("Predicted carbs for given calories are: ", predicted)
# import math
# ceil = df[df['carbs'] == (math.ceil(predicted))]
# print(ceil)
# floor = df[df['carbs'] == (math.floor(predicted))]
# print(floor)
return predicted
```

Algorithm 2
Get Exercise

Input: day,calories,routine,weight

Output: An exercise plan for the day in the form of array

```
def getExercise(day,calories,routine,weight):
#routine is an array [back,arms,shoulders,waist,legs,chest,cardio,neck] containing boolean values
# 1,3,5 have activity 2,4 have exercise plan and 0 is free
if(day==1 or day==3 or day==5):
print("activity day")
activity=getActivity(weight,calories)
return(activity)
elif(day==2 or day==4 ):
print("exercise plan day")
plan=makeExercisePlan(routine,calories)
return(plan)
else:
return("free")
```

Algorithm 2.1
MakeClusters

Input: csv file

Output: exercise clusters

```
#importing the exercise file
df=pd.read_csv('fitness_exercises.csv')
activity_df=pd.read_csv("exercise_dataset.csv")
print(df)
#add a column calories burnt based on randomly generated MET later to be replaced
#"Calories Burned = (MET x Body Weight in kg x Duration in hours)"
# 0.1 means 10 minutes should be cut short 10/8=0.1
data = np.random.randint(4, 16, size=len(df))*50*0.1
df['caloriesBurned'] = data
#transforming or encoding the equipment, bodyPart and target of the dataset
le=LabelEncoder()
df["equipmentEncoded"]=le.fit_transform(df["equipment"])
df["bodyPartEncoded"]=le.fit_transform(df["bodyPart"])
df["targetEncoded"]=le.fit_transform(df["target"])
activity_df['exercise'] = le.fit_transform(activity_df['Activity, Exercise or Sport (1 hour)'])
print(df)
#applying k means clustering
km=KMeans(n_clusters=19,n_init=5)
y_predicted=km.fit_predict(df[["bodyPartEncoded","targetEncoded"]])

#adding a column in dataset naming cluster which names each cluster
df["cluster"]=y_predicted

#saperating the clusters made
```

```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
df4=df[df.cluster==3]
df5=df[df.cluster==4]
df6=df[df.cluster==5]
df7=df[df.cluster==6]
df8=df[df.cluster==7]
df9=df[df.cluster==8]
df10=df[df.cluster==9]
df11=df[df.cluster==10]
df12=df[df.cluster==11]
df13=df[df.cluster==12]
df14=df[df.cluster==13]
df15=df[df.cluster==14]
df16=df[df.cluster==15]
df17=df[df.cluster==16]
df18=df[df.cluster==17]
df19=df[df.cluster==18]
```

Algorithm 2.2
getDataFrame

Input: category

Output: dataframes in an array

```
#returns the array of dataframes related to the sent category
def getDataFrame(category):
dataframesExisting=[df1,df2,df3,df4,df5,df6,df7,df8,df9,df10,df11,df12,df13,df14,df15,df16,df17,df18,df19]
dataframes=[]
for i in dataframesExisting:
if(i.to_numpy()[0][0]==category):
dataframes.append(i)
#print("The dataframes included are "+str(dataframes)+ str(len(dataframes)) )
return dataframes
```

Algorithm 2.3
predictParticularExercise

Input: caloriestoBeBurnt,dataframe

Output: exercise

```
def predictParticularExercise(caloriestoBeBurnt,dataframe):
# applying random forest
#preprocessing and splitting X and Y
#print(df1)

#splitting
from sklearn.model_selection import train_test_split
#getting the independent and dependent variables]
X=dataframe[["caloriesBurned","equipmentEncoded"]]
Y=dataframe["name"]
```

```
#predict using model
from sklearn.ensemble import RandomForestClassifier
model=RandomForestClassifier(n_estimators=100)
#training the model
model.fit(X.values,Y.values)
#predicting
#get the encoded body weight
bodyWeightEncoded=(getDataFrame("neck")[0].to_numpy())[0][7]
#print("body wight is encoded to")
#print(bodyWeightEncoded)
predict=model.predict([[caloriestoBeBurnt,bodyWeightEncoded]])
print(predict)
a=((dataframe.loc[(dataframe['name'].isin(predict))])["caloriesBurned"])
b=dataframe[dataframe['name'] == predict[0]]
#print("b is")
#print(b.to_numpy()[0])
#return ((predict[0],a.to_numpy()[0]))
return b.to_numpy()[0]
```

Algorithm 2.4

appendExercise

Input: frames,caloriesBurntForEachCategory

Output: exercises in an array

```
#add exercises from the frame to plan
def appendExercise(frames,caloriesBurntForEachCategory):
exercises=[]
caloriesAlreadyBurnt=0
for index,k in enumerate(frames):
result=predictParticularExercise(caloriesBurntForEachCategory,k)
#print("result got is")
#print(result)
#if the newly predicted calories increases the calories to be burned limit break the loop
if (caloriesAlreadyBurnt+result[6]>caloriesBurntForEachCategory and index!=0):
#print("In break if of back")
break
caloriesAlreadyBurnt+=result[6]
#print("Calories already burnt for back are now "+str(caloriesAlreadyBurnt))
# add to exercises the name and calories burned in the exercise
exercises.append(result)
return exercises
```

Algorithm 3 Calculating calories per day

Input: weight, height, age and gender of a person

Output: Total calories a person can eat in a day

```
# Calories calculator according to a person personal information.
import math
import numpy as np

# Taking inputs from patient
weight = float(input("Enter your weight : "))
gender = (input("Enter your gender : ")).lower()

# Changing weight in kgs if given in pounds
def poundToKG(weightPound):
    return weightPound*2.20462

# Calculating the body fat percentage of patient
def bodyFatPercentage():
   height = float(input("Enter your height measurement in feet : "))
   age = float(input("enter your age : "))
   bmi = float(weight/math.pow((height*0.3048),2))
   if(gender == 'male'):
       return -44.988 + (0.503 * age) + (10.689 * 0) + (3.172 * bmi) - (0.026 * math.pow(bmi, 2)) + (0.181 * bmi * 0) - (0.02 * bmi * age) -
       (0.005 * math.pow(bmi, 2) * 0) + (0.00021 * math.pow(bmi, 2) * age)
  elif(gender == 'female'):
       return -44.988 + (0.503 * age) + (10.689 * 1) + (3.172 * bmi) - (0.026 * math.pow(bmi, 2)) + (0.181 * bmi * 1) - (0.02 * bmi * age) –
       (0.005 * math.pow(bmi, 2) * 1) + (0.00021 * math.pow(bmi, 2) * age)
# Calculating the lean fat of patient
def leanFat():
   bodyFatpercent = bodyFatPercentage()
   if(gender == 'male'):
       if(bodyFatpercent>=10 and bodyFatpercent<=14):
          return 1.0
       elif(bodyFatpercent>=15 and bodyFatpercent<=20):
           return 0.95
       elif(bodyFatpercent>=21 and bodyFatpercent<=28):
           return 0.90
       elif(bodyFatpercent>28):
          return 0.85
   elif(gender == 'female'):
       if(bodyFatpercent>=14 and bodyFatpercent<=18):
           return 1.0
       elif(bodyFatpercent>=19 and bodyFatpercent<=28):
           return 0.95
       elif(bodyFatpercent>=29 and bodyFatpercent<=38):
           return 0.90
       elif(bodyFatpercent>38):
           return 0.85
# Asking patient daily activity level
def dailyActivityLevel():
   activity = int(input("Select option from 1 to 5 for your daily activity level \n"+
   "1. (very light) typical office job (sitting,studying,little walking throughout the day)\n"+
   "2. (Light)-Any job where you mostly stand or walk (teaching,shop/lab work,some walking through the day)\n"+
   "3. (Moderate)-jobs requiring physical activity(land scaping,cleaning,maintaienance,jogging/biking/working out 2 hours/day)\n"+
   "4. (Heavy)- Heavy manual labor(construction,dancer,athlete,hard physical activity min.4 hours/day)\n"+
   "5. (Very Heavy)-Moderate to hard physical activity mim. 8hours/day : "))
   values = {1: 1.3, 2: 1.55, 3: 1.65,  4: 1.80,5: 2.00}
   return (values[activity])
# Calculating the calories by adding the above calculated values in formula
def calculateCalories():
   lean = float(leanFat())
```

```
activity = float(dailyActivityLevel())
if(gender == "male"):
    return (weight * 1.0 * 24.0 * lean * activity)
elif(gender == "female"):
    return (weight * 0.9 * 24.0* lean * activity)
```

Algorithm 4 Calculating Bolus Insulin for Carbohydrates

Input: no of carbohydrates

Output: unit of bolus insulin

```
var carbohydrates
var carbohydrateRatio = 10
var unit = carbohydrates / carbohydrateRatio
return unit
```

Algorithm 5 Calculating correction bolus unit (for Hyperglycaemia)

Input: correctionFactor, bloodSugar, normalBloodSugar

Output: unit of bolus correction insulin

```
var carbohydrates
var amountToCorrect = bloodSugar - normalBloodSugar
var unit = amountToCorrect / correctionFactor
return unit
```

Algorithm 6 Check Allergic Medication

Input: Medicine name

Output: Boolean value telling medicine is allergic for patient or not

```
var medicine = “name of medicine”
const found = AllergicReaction.Find({“name”: medicine})
if(! found.empty())
    return True
Else
    return False
```

Algorithm 7 Detecting Retinopathy and classifying it

Input: Fundus Image of patient retina

Output: Stage of diabetes patient retina which can be the following five stages:

a) Normal
b) Mild
c) Moderate
d) Severe
e) PDR

```
# Fundus Image Pre-processing
Resize fundusImage
data augmentation of the fundusImage
Normalize fundusImage
Augment fundus Image
# Passing image to VGG-NIN Model
layers.Conv2D(32, (3,3), activation='relu', input_shape =input_shape),
layers.MaxPooling2D((2,2)),
layers.Conv2D(64,kernel_size= (3,3), activation='relu'),
layers.MaxPooling2D((2,2)),
layers.Conv2D(64,kernel_size= (3,3), activation='relu'),
layers.MaxPooling2D((2,2)),
layers.Conv2D(64, (3,3), activation='relu'),
layers.MaxPooling2D((2,2)),
layers.Conv2D(64, (3,3), activation='relu'),
layers.MaxPooling2D((2,2)),
layers.Conv2D(64, (3,3), activation='relu'),
layers.MaxPooling2D((2,2)),
layers.Flatten(),
layers.Dense(64, activation ='relu'),
layers.Dense(n_classes, activation= 'softmax')
returns predicted class probability along with confidence
```

Algorithm 8 Medicine name detection using optical character recognition

Input: medicineNameImage

Output: medicine name in text

```
# Read image with medicine name using cv2 library
    med = cv2.imread(medicineNameImage)
# Setting up text configurations
    config = ('-l eng --oem 1 --psm 3')
# using pytessercat method to change image to string according to given configurations
    text = pytesseract.image_to_string(im, config=config)
    return text
```

Algorithm 9 Calculate calories from food image

Input: top food images, side food image

Output: predicted dish name and calories

```
    read foodImage
# image passed to densenet model
    dishName = predict(top_image)
# volume estimation using open CV
def midpoint(ptA, ptB):
        return ((ptA[0] + ptB[0]) * 0.5, (ptA[1] + ptB[1]) * 0.5)
def preprocess_image(path):
# load the image, convert it to grayscale, and blur it slightly
    image = cv2.imread(path)
# perform edge detection, then perform a dilation + erosion to
# close gaps in between object edges
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
blur = cv2.GaussianBlur(gray, (9, 9), 0)
 edged = cv2.Canny(blur, 50, 100)
 edged = cv2.dilate(edged, None, iterations=1)
 edged = cv2.erode(edged, None, iterations=1)
# find contours in the edge map
 cnts = cv2.findContours(edged.copy(), cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)
 cnts = imutils.grab_contours(cnts)
 (cnts, _) = contours.sort_contours(cnts)
 cntsArea = []
 for c in cnts:
     cntsArea.append(cv2.contourArea(c))
 return cnts , cntsArea

def order_points(pts):
# sort the points based on their x-coordinates
  xSorted = pts[np.argsort(pts[:, 0]), :]
# grab the left-most and right-most points from the sorted
# x-roodinate points
  leftMost = xSorted[:2, :]
  rightMost = xSorted[2:, :]
# now, sort the left-most coordinates according to their
# y-coordinates so we can grab the top-left and bottom-left
# points, respectively
  leftMost = leftMost[np.argsort(leftMost[:, 1]), :]
(tl, bl) = leftMost
# now that we have the top-left coordinate, use it as an
# anchor to calculate the Euclidean distance between the
# top-left and right-most points; by the Pythagorean
# theorem, the point with the largest distance will be
# our bottom-right point
 D = dist.cdist(tl[np.newaxis], rightMost, "euclidean")[0]
(br, tr) = rightMost[np.argsort(D)[::-1], :]
# return the coordinates in top-left, top-right,
# bottom-right, and bottom-left order
 return np.array([tl, tr, br, bl], dtype="float32")

def estimate(cnts,cntsArea, path):
 pixelsPerMetric = None
 image = cv2.imread(path)
 largeObj = np.argmax(cntsArea)
 c = cnts[largeObj]
```

```
# compute the rotated bounding box of the contour
orig = image.copy()
box = cv2.minAreaRect(c)
box = cv2.cv.BoxPoints(box) if imutils.is_cv2() else cv2.boxPoints(box)
box = np.array(box, dtype="int")
print(box)
# order the points in the contour such that they appear
# in top-left, top-right, bottom-right, and bottom-left
# order, then draw the outline of the rotated bounding
# box
box = order_points(box)
cv2.drawContours(orig, [box.astype("int")], -1, (0, 255, 0), 2)
# loop over the original points and draw them
for (x, y) in box:
  cv2.circle(orig, (int(x), int(y)), 5, (0, 0, 255), -1)
  # unpack the ordered bounding box, then compute the midpoint
# between the top-left and top-right coordinates, followed by
# the midpoint between bottom-left and bottom-right coordinates
(tl, tr, br, bl) = box
(tltrX, tltrY) = midpoint(tl, tr)
(blbrX, blbrY) = midpoint(bl, br)
# compute the midpoint between the top-left and top-right points,
# followed by the midpoint between the top-righ and bottom-right
(tlblX, tlblY) = midpoint(tl, bl)
(trbrX, trbrY) = midpoint(tr, br)
# draw the midpoints on the image
cv2.circle(orig, (int(tltrX), int(tltrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(blbrX), int(blbrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(tlblX), int(tlblY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(trbrX), int(trbrY)), 5, (255, 0, 0), -1)
# draw lines between the midpoints
cv2.line(orig, (int(tltrX), int(tltrY)), (int(blbrX), int(blbrY)),(255, 0, 255), 2)
cv2.line(orig, (int(tlblX), int(tlblY)), (int(trbrX), int(trbrY)),(255, 0, 255), 2)
 # compute the Euclidean distance between the midpoints
dA = dist.euclidean((tltrX, tltrY), (blbrX, blbrY))
dB = dist.euclidean((tlblX, tlblY), (trbrX, trbrY))
print('DA', dA, "DB", dB)
# if the pixels per metric has not been initialized, then
# compute it as the ratio of pixels to supplied metric
# (in this case, inches)
if pixelsPerMetric is None:
  pixelsPerMetric = dB / 0.955
          # compute the size of the object
dimA = dA / pixelsPerMetric
dimB = dB / pixelsPerMetric
# draw the object sizes on the image
cv2.putText(orig, "{:.1f}in".format(dimA),(int(tltrX - 15), int(tltrY - 10)), cv2.FONT_HERSHEY_SIMPLEX,0.65, (255, 255, 255), 2)
cv2.putText(orig, "{:.1f}in".format(dimB),           (int(trbrX + 10), int(trbrY)), cv2.FONT_HERSHEY_SIMPLEX,0.65, (255, 255, 255),
2)
return dimA, dimB
Volume = estimateVolume(top_image, side_image )
 query = str(Volume)+' cm3 '+foodName
 print("**********************************************",query)
  api_url = 'https://api.api-ninjas.com/v1/nutrition?query={}'.format(query)
  response = requests.get(api_url, headers={'X-Api-Key': 'avnKxya254Me1ijNIJ6uIw==JA9ffe9fNotaycST'})
  if response.status_code == requests.codes.ok:
      information = response.text
  else:
      information = 'Cant find your required food'
  return information
```

10.1 External APIs/SDKs

Table 4-1: External APIs/SDKs

Name of API and version	Description of API	Purpose of usage	List down the API endpoint/function/class in which it is used
Microsoft computer vision API	This API is used to detect text from images.	The API will be used to detect medication name from image of medicine.	Medicine module.

10.2 User Interface

Following images represents the user interface of our application:

10.2.1 Login Screen

Figure 4-2 contains login screen of our app login.

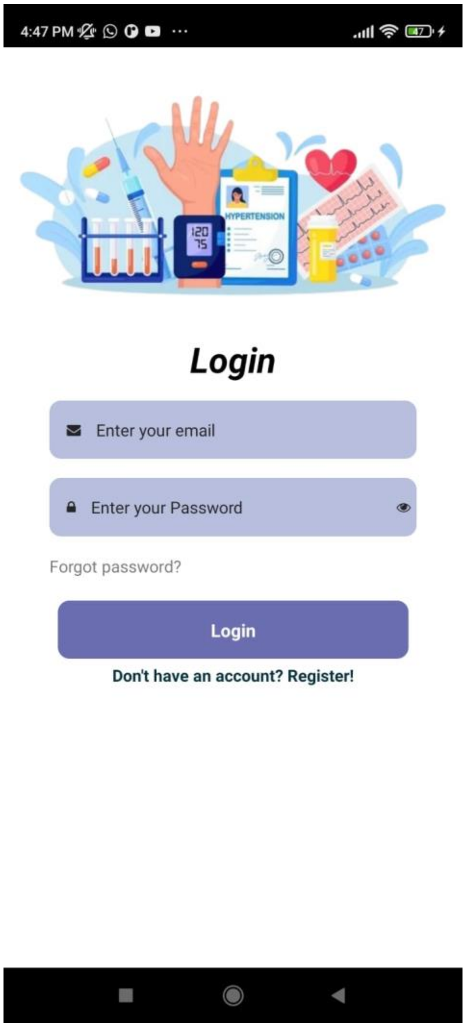


Figure 4-2: Login screen

10.2.2. Home Screen

Figure 4-1 contains first screen where you can where user enters his email and password.

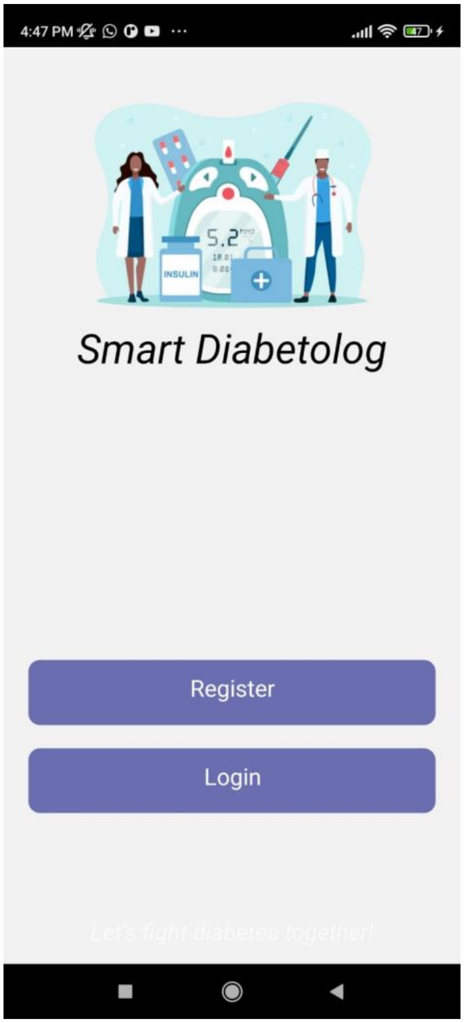


Figure 4-1: Start screen

10.2.3 Home screen

Figure 4-3 contains home screen of application where you can select different pages.

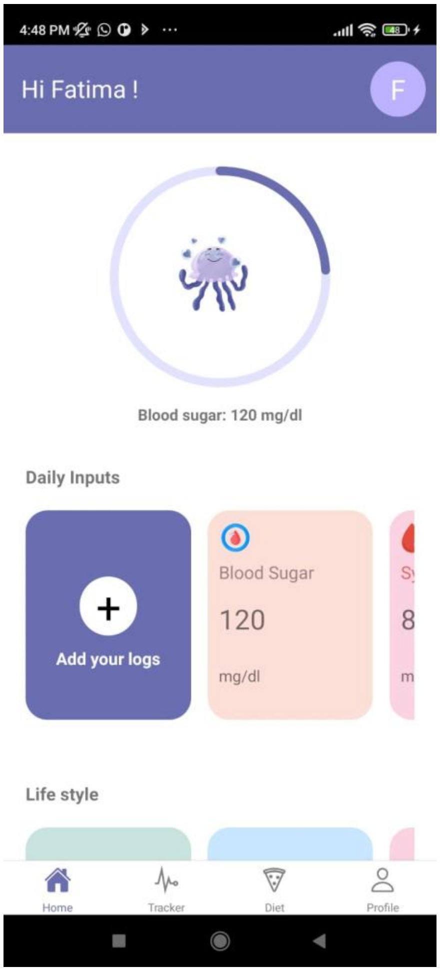


Figure 4-3: Home screen

10.2.4 Tracker main screen

Figure 4-4 shows the tracker screen where you can select which thing you want to track or view.

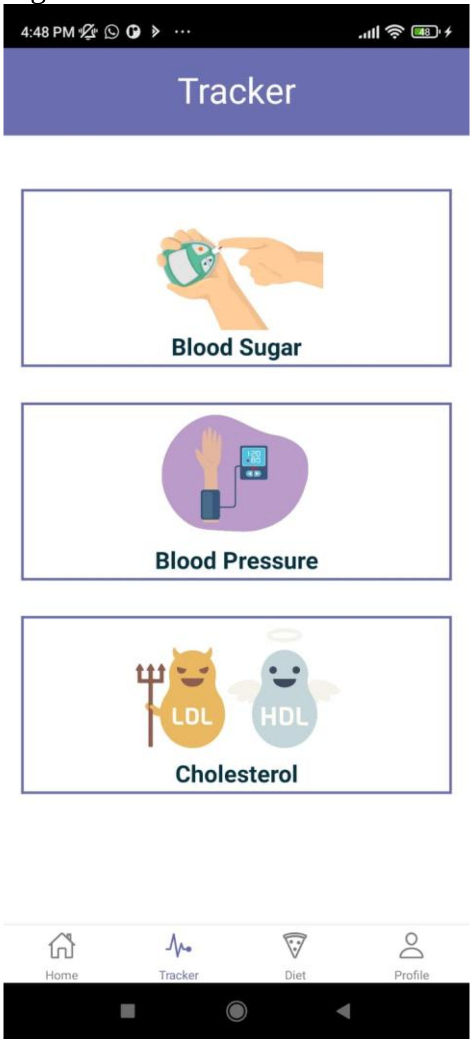
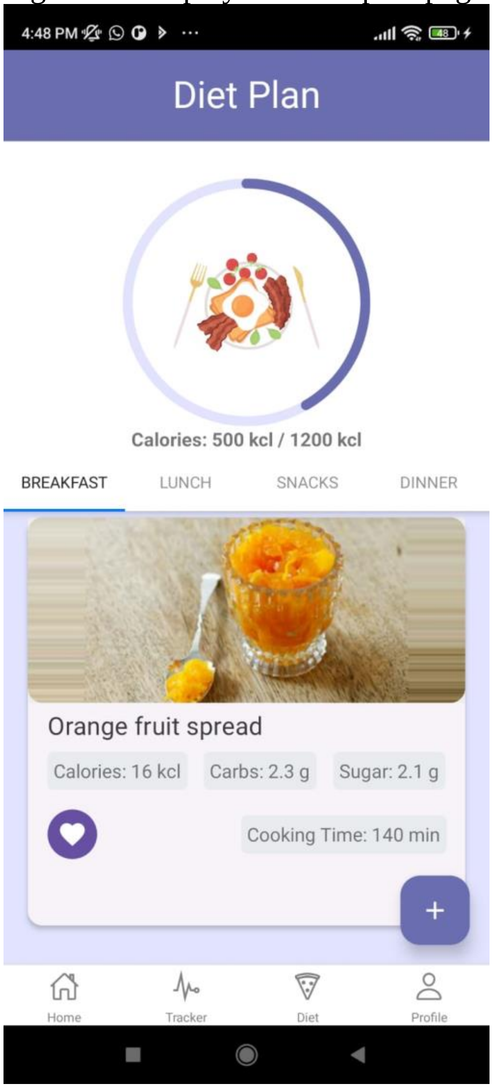


Figure 4-4: Tracker Screen

10.2.5 Diet plan view page

Figure 4-5 displays the diet plan page shows the diet plan according to your information.



Diet Plan

Calories: 500 kcl / 1200 kcl

Orange fruit spread

Calories: 16 kcl

Carbs: 2.3 g

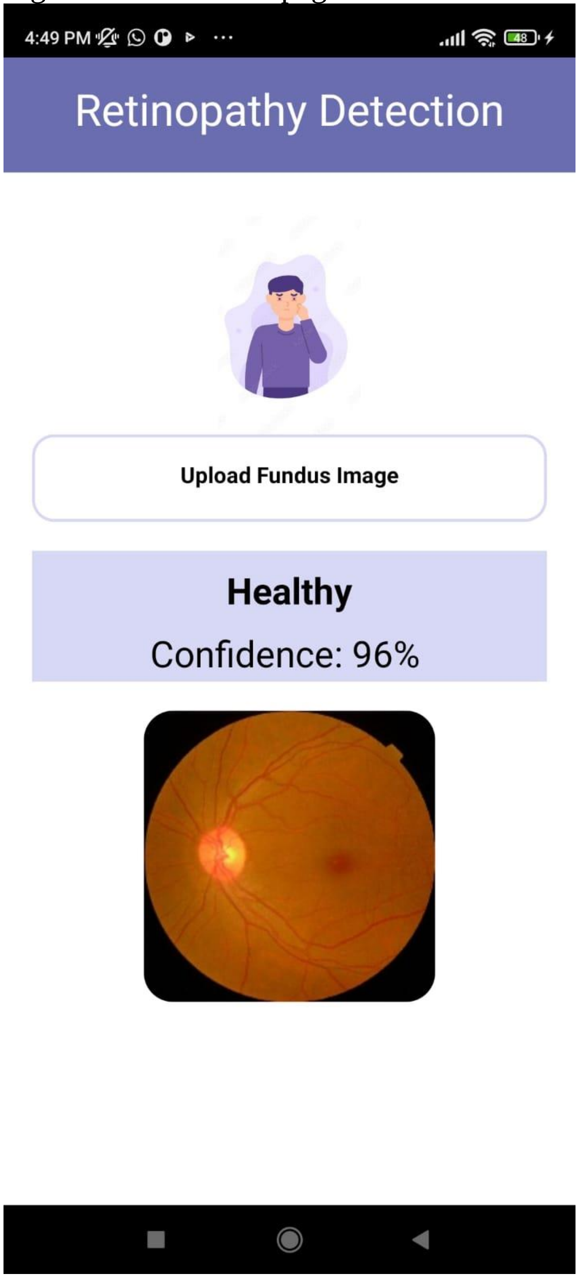
Sugar: 2.1 g

Cooking Time: 140 min

Figure 4-5: Diet Plan Screen

10.2.6 Retinopathy detection page

Figure 4-6 shows the page that will take fundus image and predicts retinopathy on it.



Retinopathy Detection

Healthy

Confidence: 96%

Figure 4-6: Retniopathy Screen

11 Deployment

The application is under implementation phase and not yet deployed.

11.1 Testing and Evaluation

The following section contains test cases of the implemented application.

11.1.1 Unit Testing

11.1.1.1 Unit Testing 1: Register as Patient with valid and invalid credentials

Testing Objective: To ensure that the user registers with the correct data

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check the email field of registration page to validate that it takes proper email	Email: abc@gmail.com	Validates email without warning	Pass
2	To check the email field of registration to validate that it displays error message on wrong email.	Email: abc.com	Highlights field box in red along with a wrong email warning.	Pass
3	To check that the user entered password contains at least one Uppercase letter, 1 digit and special character	password: P@ss123	Validates password without any warning.	Pass
4	To check that the user entered password does not contains Uppercase letter, digit and special character	password: passooo	Highlights the field box with a message that password must contain at least one uppercase letter, digit and a special character	
5	To check that the user entered password and confirm password is same.	password: P@ss123 confirm password: P@ss123	Validate the password and confirm password without warning	Pass
6	To check that the user entered password and	password: P@ss123 confirm password: password123	Highlights both password and confirm password field box in red	Pass

	confirm password is different.		along with a warning message.	

11.1.1.2 *Unit Testing 2: Login as Patient with valid and invalid credentials*

Testing Objective: To ensure that the user logins with the correct data

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that both entered email and password are correct	Email:abc@gmail.com password: P@ss123	Allows to login without warning	Pass
2	To check that both entered email and password are incorrect	Email:abc.gmail.com password: passwer	Highlights field box in red along with a message “Enter correct email and password”	Pass

11.1.1.3 *Unit Testing 3: Add a blood sugar instance with valid and invalid credentials*

Testing Objective: To ensure that the user adds blood sugar with the correct data

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that the user selected time is updated immediately.	Time: 7:30 PM	User selected time is updated on the screen	Pass
2	To check that blood sugar concentration is added in numbers.	Blood sugar concentration = 200	Validates the blood sugar concentration without warning	Pass
3	To check that the selected radio button for blood sugar concentration unit is highlighted	User selects the mmol/l	Mmol/l radio button is highlighted with dark blue color.	Pass
4	To check that the user selected event from event dropdown is updated immediately in the event field box.	User select Before lunch” event	The event box field contains “Before Lunch”	pass
5	To check that the user successfully writes a note	Notes: abc	The added text is successfully written in the notes field box.	Pass

11.1.1.4 Unit Testing 4: Add a blood pressure instance with valid and invalid credentials

Testing Objective: To ensure that the user adds blood pressure with the correct data

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that disystolic pressure reading is added in numbers.	disystolic pressure concentration = 120	Validates the blood pressure concentration without warning	Pass
2	To check that systolic pressure reading is added in numbers.	systolic pressure concentration = 100	Validates the blood pressure concentration without warning	Pass

11.1.1.5 Unit Testing 5: Add a cholesterol instance with valid and invalid credentials

Testing Objective: To ensure that the user adds cholesterol readings with the correct data

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that LDL concentration is added in numbers.	LDL concentration = 120	Validates the LDL concentration successfully.	Pass
2	To check that HDL concentration is added in numbers.	HDL concentration = 130	Validates the HDL concentration successfully.	Pass
3	To check that Triglycerides concentration is added in numbers.	Triglycerides concentration = 120	Validates the Triglycerides concentration successfully.	Pass

11.1.1.6 Unit Testing 6: Update a user profile credentials.

Testing Objective: To ensure that the user updates user profile information with the correct data

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that user successfully changes its username.	Username=Fatima Shahzad	Validates username without any warning.	Pass
	To check that user changes its username to empty.	Username=	Highlights the field box in red with message “Username cannot be empty”	Pass

2	To check that user successfully changes its email with new valid email.	Email=abc@gmail.com	Validates the email without any warning.	Pass
3	To check that user changes its email with invalid email.	Email=abc.gmail.com	Highlights the field box in red with message “Email is not valid”	Pass
4	To check that height feets is added in numbers.	Feet=5	Validates the feet without any warning.	Pass
5	To check that height inches is added in numbers below between 0 and 12.	Inches=10	Validates the inches without any warning.	Pass
6	To check that height inches is added in numbers above 12.	Inches=15	Highlights the field box in red with message “Inches cannot be greater than or equal to 12”	Pass
7	To check that weight is added in numbers.	Weight=15	Validates the weight without any warning.	Pass

11.1.1.7 Unit Testing 7: Add a medication instance with valid and invalid credentials

Testing Objective: To ensure that that a medication may be oral or insulin are added with the correct data.

No.	**Test case/Test script**	**Attribute and value**	**Expected result**	**Result**
1	To check that medication name is successfully added	Name=Panadol	Validates the medication name without any warning.	Pass
2	To check that the oral medication unit is entered in numbers	unit = 500	Validates the medication unit successfully.	Pass
3	To check oral medication intake time with more than 1 timing instances are successfully added	Timing: 2:00 PM, 7:00 PM, 11:00 PM	Successfully adds the timings for the medication intake.	Pass
4	To check that the selection of long acting insulin displays related fields only	Selection of long acting insulin radio button	2 fields naming time and insulin unit appears on the screen.	Pass

5	To check that the selection of fast acting insulin displays related fields only	Selection of fast acting insulin radio button	2 fields naming glycemic index and insulin sensitivity factor appears on the screen.	Pass
6	To check that the time for the long acting insulin is successfully selected	Time=11:00 PM	Successfully adds the timing for the long acting insulin intake.	Pass
7	To check that if the time for the long acting insulin is not selected, its selected as 10:00 PM by default.	Time is not selected	Successfully adds the timing for the long acting insulin intake as 10:00.	Pass
8	To check that the long acting insulin units are entered in numbers	units = 2	Validates the insulin units successfully.	Pass
9	To check that the fast acting insulin glycemic index is entered in numbers	GI=20	Validates the glycemic index successfully.	Pass
10	To check that the fast acting insulin sensitivity factor is entered in numbers	Insulin Sensitivity factor =10	Validates the insulin sensitivity factor successfully.	Pass

11.1.1.8 Unit Testing 8: Add a allergic reactions.

Testing Objective: To ensure that the allergic reactions are successfully added.

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that the selected allergic reaction radio button is highlighted with dark blue color.	User a food radio button.	The radio button with food is highlighted with a dark blue color.	Pass
2	To check that user successfully adds the name of the allergic reaction medication.	Name = Arinac	Validates the medication name successfully.	Pass
3	To check that user successfully adds the name of the allergic reaction food.	Name = Apple	Validates the food name successfully.	Pass

11.1.1.9 Unit Testing 9: To check states.

Testing Objective: To ensure that all the states are updated successfully after updation.

No.	Test case/Test script	Attribute and value	Expected result	Result
1	To check that the todayExerciseDone status changed to true after exercise is done.	todayExerciseDone = false	todayExerciseDone changed to true	Pass
2	To check that the back exercise status changed after setting is turn ON.	back = false	Back status changed to true	Pass
3	To check that todayBreakfastDone changed to true when user initiates the button.	todayBreakfastDone = false	todayBreakfastDone changed to true.	Pass

11.1.2 Functional Testing

11.1.2.1 Functional Testing 1: Register an account

Objective: To ensure that the user is successfully registered for an account.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Register an account	User initiates the register button.	User is directed to confirmation code screen	Confirmation Code screen appears.	Pass
2.	Register sends registration confirmation message to user email.	User initiates the register button.	Registration confirmation message received at email.	Confirmation email received.	Pass
3.	User writes correct confirmation code.	User writes the confirmation code.	User is directed to its account Home Screen.	Directed to Home Screen	Pass
4.	User writes incorrect confirmation code.	User writes the confirmation code.	User is displayed with a message “Incorrect Code”.	“Incorrect Code” message appears on the screen.	Pass

11.1.2.2 Functional Testing 2: Login into the account

Objective: To ensure that the user account with the correct user information is loaded.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Login with correct information	Email: fshahzad2123@gmail.com Password: Pass@123	Main page with the user's account information appears.	Logged in and redirected to the user account.	Pass
2.	Login With incorrect information.	Email: abc@gmail.com Password: Pass@123	A message appears on the login page "Incorrect email or password"	Message saying incorrect email or password appears.	Pass

11.1.2.3 Functional Testing 3: Tracker tracks the health factors

Objective: To ensure that the user successfully views, adds or updates its health factors.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the tracker.	User initiates the tracker button from home screen.	User directed to tracker screen.	Redirected to Tracker Screen.	Pass
2.	User views the blood sugar.	User initiates the blood sugar button from tracker screen.	User is redirected to screen where all the user's blood sugar instances for the day appears.	Redirected to screen where all blood sugar instances for the day appears.	Pass
3.	User views the blood pressure.	User initiates the blood pressure button from tracker screen.	User is redirected to screen where all the user's blood pressure instances for the week or month appears.	Redirected to screen where all blood pressure instances appears.	Pass
4.	User views the cholesterol.	User initiates the cholesterol button from tracker screen.	User is redirected to screen where all the user's cholesterol instances for the 6 month appears.	Redirected to screen where all 6 months cholesterol instances appears.	Pass

5.	User initiates to add new blood sugar instance.	User initiates the add blood sugar button.	User is redirected to add blood sugar page.	Redirected to add blood sugar page.	Pass
6.	User initiates to add new blood pressure instance.	User initiates the add blood pressure button.	User is redirected to add blood pressure page.	Redirected to add blood pressure page.	Pass
7.	User initiates to add new cholesterol instance.	User initiates the add cholesterol button.	User is redirected to add cholesterol page.	Redirected to add cholesterol page.	Pass
8.	User selects a blood sugar instance.	User presses the blood sugar instance.	User is redirected to the page with blood sugar instance details.	Redirected to page having blood sugar instance details.	Pass
9.	User selects a blood pressure instance.	User presses the blood pressure instance.	User is redirected to the page with blood pressure instance details.	Redirected to page having blood pressure instance details.	Pass
10.	User selects a cholesterol instance.	User presses the cholesterol instance.	User is redirected to the page with cholesterol instance details.	Redirected to page having cholesterol instance details.	Pass
11.	User saves the changes in the blood sugar instance.	User initiates the save changes button.	Blood Sugar instance gets updated.	Blood Sugar instance updated.	Pass
12.	User saves the changes in the blood pressure instance.	User initiates the save changes button.	Blood Pressure instance gets updated.	Blood Pressure instance updated.	Pass
13.	User saves the changes in the cholesterol instance.	User initiates the save changes button.	Cholesterol instance gets updated.	Cholesterol instance updated.	Pass

11.1.2.4 Functional Testing 4: Edit user profile details

Objective: To ensure that the user successfully views or updates the profile information.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the profile information.	User initiates the profile button.	User directed to profile Screen.	Redirected to Profile.	Pass
2.	User saves the changes in the profile.	User initiates the save changes button from profile Screen.	User information gets updated.	User information updated.	Pass

11.1.2.5 Functional Testing 5: User maintains Medication history

Objective: To ensure that the user successfully views medication prescription adds, updates or view oral and insulin medication.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the Medication.	User initiates the Medication button from home screen.	User directed to medication screen.	Redirected to Medication Screen.	Pass
2.	User views the oral medication.	User initiates the oral medication button from medication screen.	User is redirected to screen where all the user's oral medication instances of current prescription appears.	Redirected to screen where all oral medication instances for the current prescription appears.	Pass
3.	User views the insulin medication.	User initiates the insulin medication button from medication screen.	User is redirected to screen where all the user's insulin instances of current prescription appears.	Redirected to screen where all insulin medication instances for the current prescription appears.	Pass
4.	User views the Prescriptions.	User initiates the Prescription button from medication screen.	User is redirected to screen where all the prescription instances appears.	Redirected to screen where all prescription instances appears.	Pass
5.	User initiates to add new oral	User initiates the add oral medication button.	User is redirected to add oral medication page.	Redirected to add oral medication page.	Pass

	medication instance.				
6.	User initiates to add new insulin medication instance.	User initiates the add insulin medication button.	User is redirected to add insulin medication page.	Redirected to add insulin medication page.	Pass
7.	User selects a oral medication instance.	User presses the oral medication instance.	User is redirected to the page with oral medication instance details.	Redirected to page having oral medication instance details.	Pass
8.	User selects an insulin medication instance.	User presses the insulin medication instance.	User is redirected to the page with insulin medication instance details.	Redirected to page having insulin medication instance details.	Pass
9.	User selects a prescription instance.	User presses the prescription instance.	User is redirected to the page with prescription instance details.	Redirected to page having prescription instance details.	Pass
10.	User saves the changes in the oral medication.	User initiates the save changes button.	Oral medication gets updated.	Oral medication updated.	Pass
11.	User saves the changes in the insulin medication.	User initiates the save changes button.	Insulin medication gets updated.	Insulin medication updated.	Pass
12.	User saves the changes in the prescription title.	User initiates the save changes button.	Prescription Title gets updated.	Prescription Title updated.	Pass

11.1.2.6 Functional Testing 6: User maintains Allergic Reaction

Objective: To ensure that the user successfully views, adds, or updates food and medication allergic reactions.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the Allergic Reactions.	User initiates the Allergic Reaction button from home screen.	User directed to Allergic Reaction screen.	Redirected to Allergic Reaction Screen.	Pass
2.	User views the food allergic reactions.	User initiates the food allergic reactions button from Allergic Reaction screen.	User is redirected to screen where all the user's food allergic reactions instances appears.	Redirected to screen where all food allergic reactions instances appears.	Pass
3.	User views the medication allergic reactions.	User initiates the medication allergic reactions button from Allergic Reaction screen.	User is redirected to screen where all the user's medication allergic reactions instances appears.	Redirected to screen where all medication allergic reactions instances appears.	Pass
4.	User initiates to add new allergic reaction instance.	User initiates the add allergic reaction button.	User is redirected to add allergic reaction page.	Redirected to add allergic reactions page.	Pass
5.	User selects a allergic reaction instance.	User presses the allergic reaction instance.	User is redirected to the page with allergic reaction instance details.	Redirected to page having allergic reaction instance details.	Pass
6.	User saves the changes in the allergic reaction.	User initiates the save changes button.	allergic reaction gets updated.	allergic reaction updated.	Pass

11.1.2.7 Functional Testing 7: User view the diet plan

Objective: To ensure that the user successfully views diet plan and saves meal.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the diet plan.	User initiates the Diet button from home screen.	User directed to diet screen.	Redirected to diet screen.	Pass
2.	User open saves meal	User initiates the add button on screen from diet screen	User is redirected to save meal screen.	Redirected to save meal screen	Pass
3.	User adds meal information.	Users enter meal information ad select save	User meal information is updated and redirected to diet screen.	User meal information is updated and redirected to diet screen.	Pass
4	User adds the meal to favorite meal dish	User initiate the heart button on recipe card on the diet screen.	Meal added to favorite meal list	Meal added to favorite meal list	Pass

11.1.2.8 Functional Testing 8: User view the exercise plan

Objective: To ensure that the user successfully views exercise plan and start exercise.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the exercise plan.	User initiates the exercise button from home screen.	User directed to exercise screen.	Redirected to exercise screen.	Pass
2.	User starts exercise routine	User initiates the start exercise button	Exercise start page appears	Redirected to save exercise page	Pass
3.	User views exercises plan settings	Users inities exercise plan settings	Exercise plan settings appear on page	Exercise plan settings appear on page	Pass
4	User set exercise plan setting	User selects the desired setting and selects ok	Exercise setting set and redirected to exercise plan page	Exercise setting set and redirected to exercise plan page	Pass

11.1.2.9 Functional Testing 9: User view retinopathy page and checks it

Objective: To ensure that the user successfully views exercise plan and start exercise.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	User views the retinopathy page	User initiates the retinopathy button from home screen.	User directed to retinopathy screen.	Redirected to retinopathy screen.	Pass
2.	User selects a fundus image	User initiates upload an image button	Images uploaded and result received	Images uploaded and result received	Pass

11.1.3 Integration Testing

11.1.3.1 Integration Testing 1:

Objective: To ensure that the diet plan made contains no food user is allergic with.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	To check that the diet chart for the day contains no allergic food ingredient.	Diet chart for the day, allergies=[beef, egg]	Diet chart should not contain any recipe contain beef and egg as its ingredient.	Diet chart contains no food having allergic ingredients.	Pass

11.1.3.2 Integration Testing 2:

Objective: To ensure that the user added medication in a prescription is safe for the user's medication allergies.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	To check that the added medication in prescription contains any active ingredient user is allergic with.	Medication=Panadol, allergies=[paracetamol]	Warning is shown that you are allergic to the medicine.	Warning for the allergy shown.	Pass

11.1.3.3 Integration Testing 3:

Objective: To ensure that the total calories for the diet chart are calculated on users personal details.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	To check that the calories for diet chart are calculated on the basis of weight, height, activityLevel, and gender.	Weight = 50 Height = 5 feet 3 inches activityLevel = Light gender=Female	Calories for the diet chart calculated.	Calories calculated on the basis of user factors.	Pass

11.1.3.4 Integration Testing 4:

Objective: To ensure that the total calories to burn in the exercise plan are calculated on users personal details.

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	To check that the burning calories for exercise plan are calculated on the basis of weight, height, activityLevel, and gender.	Weight = 50 Height = 5 feet 3 inches activityLevel = Light gender=Female	Burning Calories for the exercise plan calculated.	Burning Calories calculated for exercise plan based on factors.	Pass

11.1.4 Business Rules Testing

Business Rule 1: The history of only a year would be available for the user.

Condition	Rule 1	Rule 2
History Available?	Yes	No
Last 1 Year History Shown	Yes	No

Business Rule 2: The user must register for getting an access of the system.

Condition	Rule 1	Rule 2
User Registered?	Yes	No
Have System Access	Yes	No

Business Rule 3: If the user has diabetes type-1 then they would have no connection with diabetic oral medications.

Condition	Rule 1	Rule 2
Diabetes Type?	Type 1	Type 2
Diabetes Oral Medication	No connection	Have connection

Business Rule 4: The system must use basal-bolus regimen for insulin management.

Condition	Rule 1	Rule 2
Insulin?	Yes	No
Basal Bolus Regimen	Yes	No

Business Rule 5: If the user tracks cholesterol, then the system must show the cholesterol record of 6 months to the user.

Condition	Rule 1	Rule 2
Track Cholesterol?	Yes	No
Show 6 month record	Yes	No

Business Rule 6: If the user tracks blood sugar then the system must show today’s blood sugar record to the user.

Condition	Rule 1	Rule 2
Track Blood Sugar?	Yes	No
Show today’s Blood Sugar Record	Yes	No

Business Rule 7: If the user tracks blood pressure then the system must show blood pressure record of a week to the user.

Condition	Rule 1	Rule 2
Track Blood Pressure?	Yes	No
Show week record	Yes	No

Business Rule 8: The user must add only 2 current insulin medication (Fast-Acting Insulin and Long-Acting Insulin).

Condition	Rule 1	Rule 2
Use Insulin?	Yes	No

Fast-Acting Insulin	Yes	No
Long-Acting Insulin	Yes	No

12 Conclusion and Future Work

This section contains the conclusion of our project and future work that can be done to enhance our system.

12.1 Conclusion

To conclude we can say Smart Diabetolog application helps diabetes patient maintain their lifestyle in a balanced way by tracking their health factors and providing customized diet and exercise plans. It also maintains their medication and allergic reaction history to make the plans more efficient and customizable. It also keeps a check on diabetes patient eyes disease by checking his fundus images for retinopathy. Moreover, it also provides awareness to patients about diabetes, so, we can say that our application Smart Diabetolog is a full package for maintaining a diabetes patients lifestyle.

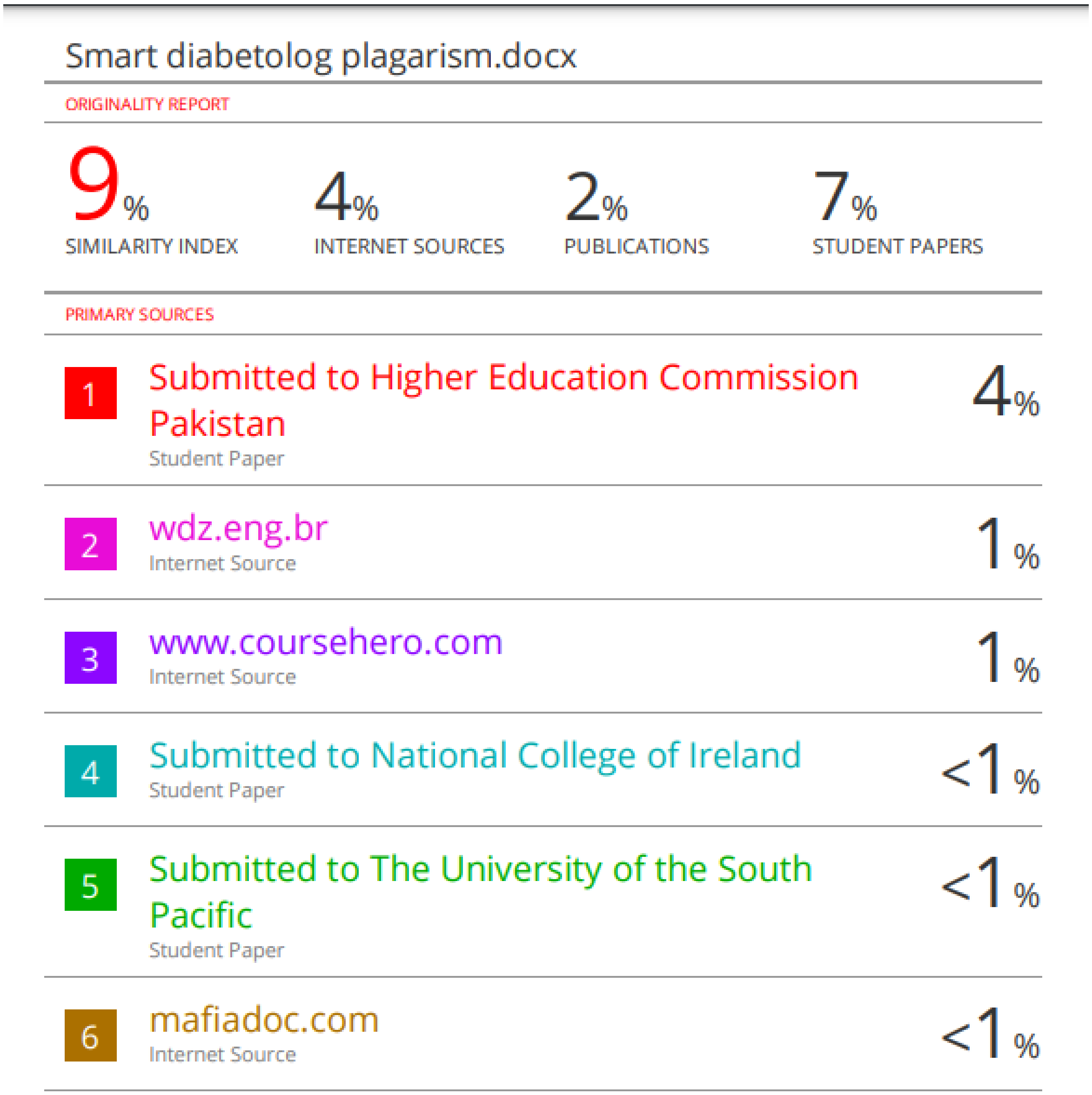
12.2 Future Work

In future work we can scrape more websites for recipes and exercises to make our system more efficient and can also train our systems with better and new datasets. In this way we can also enhance our system accuracy and efficiency.

13 References:

1. https://www.ibm.com/cloud/learn/supervised-learning
2. https://myhealth.alberta.ca/Health/Pages/conditions.aspx?hwid=ta3112
3. https://www.telerik.com/blogs/functional-programming-javascript#:~:text=Functional%20programming%20is%20a%20really,also%20improves%20how%20you%20code.
4. https://www.toptal.com/javascript/es6-class-chaos-keeps-js-developer-up

14 Plagiarism



Smart diabetolog plagarism.docx

ORIGINALITY REPORT

9%	4%	2%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	4%
2	wdz.eng.br Internet Source	1 %
3	www.coursehero.com Internet Source	1 %
4	Submitted to National College of Ireland Student Paper	<1 %
5	Submitted to The University of the South Pacific Student Paper	<1 %
6	mafiadoc.com Internet Source	<1 %