Capstone Design Final Report

Diabetes Tracker: A Novel Mobile Application for Tracking Moods in Patients with Diabetes Mellitus and Comorbid Depression

Group Number:

6

Names and Student Numbers:

Nam, Geun Woo 20132915

Kim, Myeong Gyu 20134695

Sung, Ho Joon 20130895

Date:

June, 2019

Abstract

Diabetes Mellitus(DM) is a condition which is characterized by inability to control the blood glucose(BG) level. The number of patients with DM are steeply increasing. Since it is a nearly incurable disease, it needs continuous management including life styles such as diet, and exercises. Also, it is reported that they are at higher risk of developing clinical depression than healthy people. With mobile-enabled technology, they can track their blood glucose, exercise, diet, and mood to manage the diabetes and comorbid depression. In this project, we propose a novel approach to treat depression comorbid with diabetes using mHealth technology.

Keywords: diabetes mellitus, depression, mHealth, Android, blood glucose, mobile application, smartphone

Git Address : https://github.com/HJSUNG/capstone-design_2019

Table of Contents

1.0 Introduction

1.1 Motivation

Diabetes Mellitus(DM), commonly referred to as diabetes, is a group of metabolic diseases characterized by hyperglycemia due to inability to maintain normal blood glucose(BG) level. It is a result of defects either in secretion of insulin in beta-cell in pancreas, or action of insulin in target cells due to anomalies of receptors, or both. It is categorized into two main subtypes: The first one is called Type 1 diabetes(T1D), and the second one is called Type 2 diabetes(T2D). T1D is caused by the defects of beta-cell in pancreas which secrets insulin, while T2D is caused by the defects of the insulin receptor. If not treated in the long-term, it can lead to life-threatening complications: damage to organs, especially kidneys, eyes, nerves, heart and blood vessels. [1] So, it is important to keep BG on the normal level in the long -term.

According to 'Diabetes Fact Sheet 2018' published by Korean Diabetes Association, the number of patients with DM is about 5 million, which is 10% of the total national population. And it is steadily increasing. [2]

Depression and diabetes are closely correlated to each other. Patients with DM has twice the probability of having comorbid depression compared to normal people. According a study conducted by a local clinic in Korea, more than 30% of patients with T2D patients have comorbid depression. Depression, especially in diabetes patients has a negative impact. It interferes with managing BG. A research conducted in a hospital reports that T2D patients with depression has a worse performance in controlling meal and tracking BG compared to the control group in which T2D patients without comorbid depression. [3]

However, pre-existing applications only focus on just managing diabetes in physiological perspective such as BG, and diet. They do not consider psychological factors like depression into account. With this idea, we have decided to develop a mobile application to manage comorbid depression in patients with diabetes.

1.2 Background Research

In a study conducted by Kangbuk Samsung Hospital, Korea. from August, 2012 and to March, 2013, diabetes patients (glycated hemoglobin: HbA1c > 6.5%) are randomly assigned to either control group or treatment group. In control group, the patients only get typical care, whereas the treatment group gets additional care using a mobile application called HealthyNote. They can interact with nurses when they track their BG. The nurses reinforce their adaptive behaviors. The result says that there is a statistically significant difference between two groups in managing diabetes. The HbA1c level and triglyceride level have been improved in treatment group while the control group have not. This research suggests that there is a possibility of using a mobile application as an aid for managing long-term diseases like diabetes. [4]

According to T. M. Alanzi, et al., there is a proposal of a model for integrating psychological care with diabetes management system. It suggests a model that integrates self-directed Cognitive-Behavioral

Therapy(CBT) with BG tracking. CBT, developed by A. T. Beck, an American psychologist is a kind of approach to psychotherapy. In this model, it suggests that thought, behavior and feelings interact each other. And what keeps sustaining its cycle is cognitive distortions. They are irrational belief that keeps depression in patients. The typical examples are jumping into conclusion, arbitrary inference, and overgeneralization. By keeping diaries, the patients can notice their cognitive distortions, and challenge their beliefs and get better. With the Alanzi's model, the patients should no longer use separate applications for managing their diabetes and depression. They can manage them by using one application. [5]

However, there is no other commercially available software which address the needs of take care of both diabetes and its comorbid depression. There is a need for an application with integrated approach to it.

We have found out that we can develop an application even better than that. It will automate the tracking of emotions using sentiment analysis algorithm.

1.3 Problem Definition & Project Description

Patients with DM should regularly track their BG, and diet. And they should take their medication on time regularly. If they have depression, they have to take care of it. Despite the vast quantities of pre-existing solutions, they are either only focus on physiological perspective or require high cost. And the writing of CBT diary can be tedious work. With automated sentiment analysis, they should no longer classify their thoughts by thought, event, emotion.

The project aims to develop a low-cost mobile Android application for managing diabetes with depression. It will have an user interface that can be easily used by elderlies. And the users can track their moods by simply writing their diaries.

2.0 Project Goals

The goal of the project is to create an application that will help the patients with diabetes and comorbid depression.

3.0 Specifications

3.1 Requirements

Requirement	**Details**
#1 Platform	The application should be built and tested on Android.
#2 User Interface	The application should provide a convenient user interface for elderly patients.
#3 Storage	The application should store Blood glucose, diary, and sentiment values.
#4 Sentiment analysis	The application should perform sentiment analysis based on users' diary data.

#5 Graph	The application should plot graph of blood glucose by time.
#6 Search Items	The application should have a search function.
#7 Security	The passwords should be safely stored.

3.2 Constraints

Requirement	**Details**
#1 Mobile Device	The user must have an smartphone with Android operating system and have the application installed on it.
#2 Internet connection	The user must have an active Internet connection when using the application.

3.3 Objectives

Requirement	**Details**
#1 Tracking	The application will help users with keep tracks of their blood glucose and diaries.
#2 Reinforce healthy behaviors	The application will reinforce the user's desirable healthy behaviors to manage diabetes.

4.0 System Block Diagram and Description

The system consists of three components: the user application(front-end), server(back-end), and a database. The front-end interacts with users, e.g. inserting glucose, diary, or diet data. The back-end deals with the data generated by users. It interacts with the front-end. When data are inserted, it gets the data from the front-end and passes it to the database. When requests for data are passed, it retrieves the data from the database and passes them to the front-end. We have implemented the red block parts. (Figure 1)

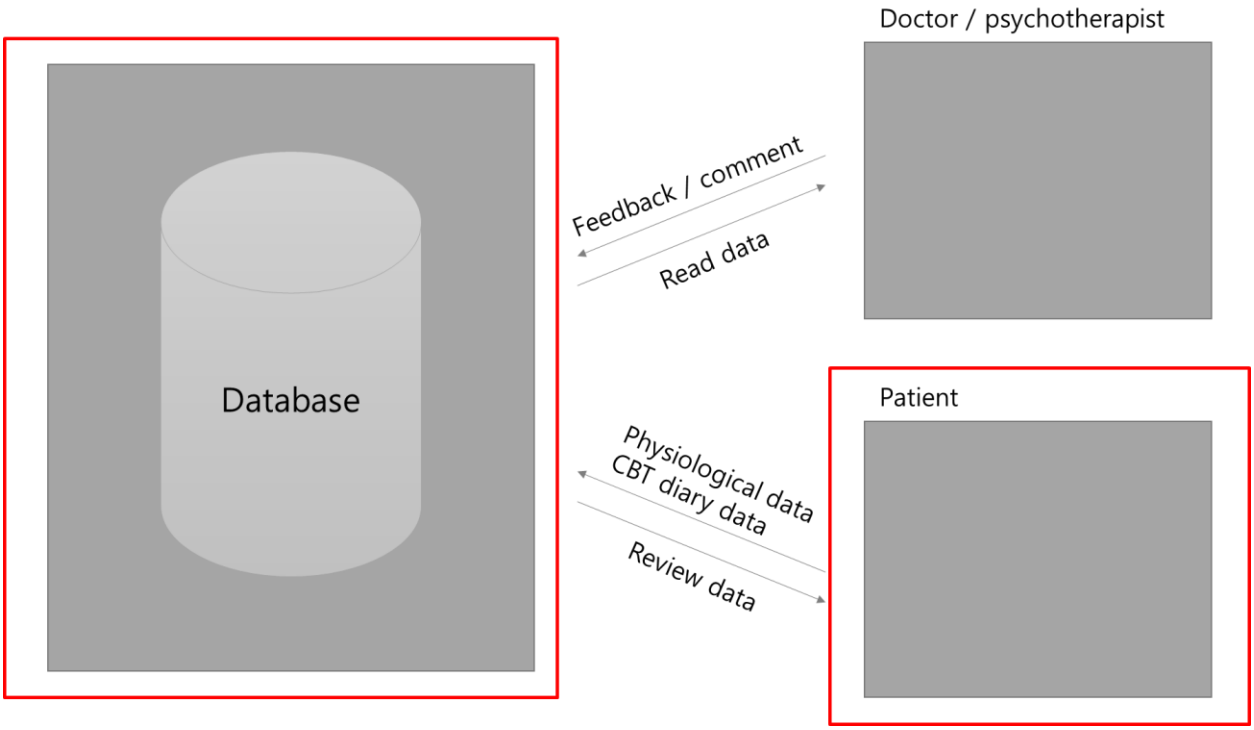
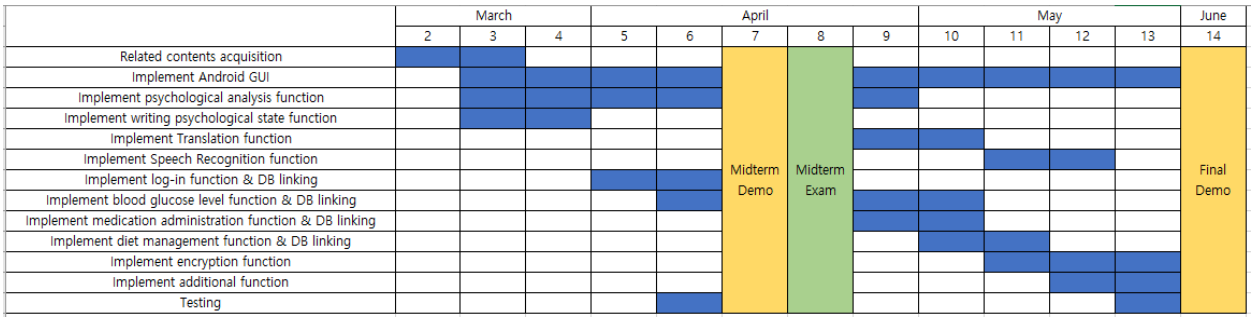


Figure 1 Overall system block diagram

5.0 Technical implementation and Project Schedule

IDE	Android Studio 3.3.2 Visual Studio Code 1.34
Target device	Oreo(8.0), min. version: Kitkat(4.4)
Server and database	PHP 5.2 MySQL 5.1.45
Programming language	Java PHP Python



Nam

- Implement client-server communication php codes
- Implement sentiment Analysis function
- Build and manage Mysql database

Sung

- Implement diary-related activities / functions
- Implement voice recognition and translation function for diary
- Implement dosing time related activities / functions

Kim

- Implement log-in , registration related activities / functions
- Implement blood glucose, meal, exercise related activities / functions
- Implement graph related functions

5.1 User Interface and Activity Description

The application is designed for smartphones with Android operating system. And it is implemented in Java and XML layout frame work in Android Studio. Each XML file is bound to an Android Java activity. The Java activity is responsible for processing user events, such as button clicks, and on.

Total activity flow is like below.

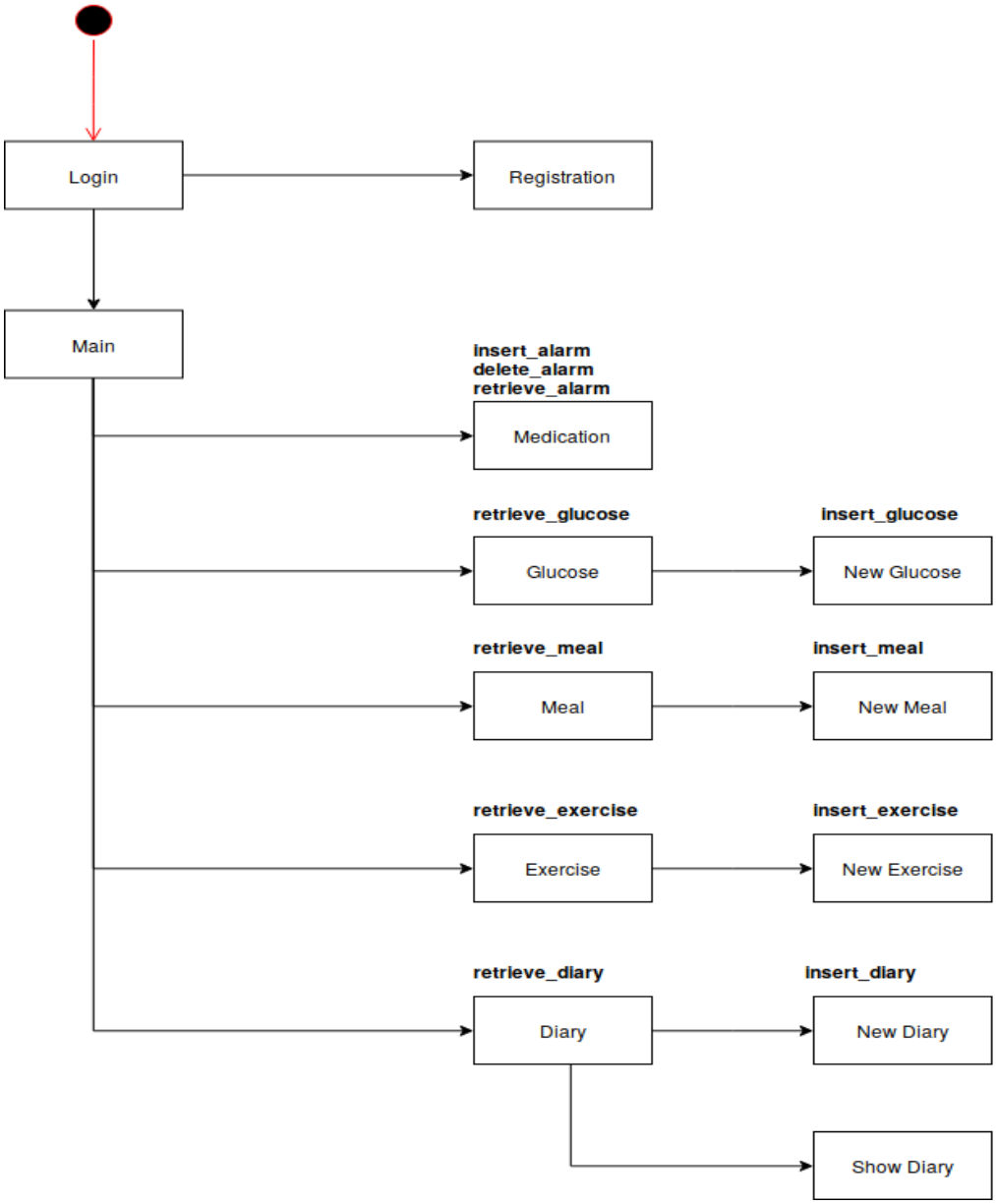
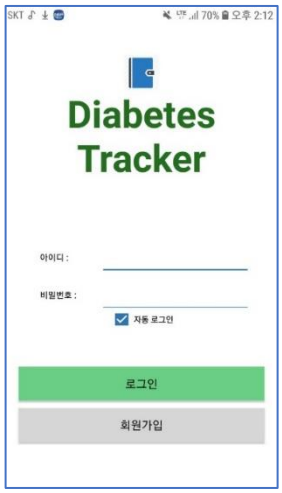


Figure 2 Activity Flow



When user runs the application, the login page as shown on the left is turned on. The login page is implemented in LoginActivity.java, and the ID and password of the user are inputted and compared with the database. If the ID and password match, the page is moved to the Main page which will be described later.

If the ID and password do not match, a login failure message will be displayed as shown on the right.

In this screen, you can move to the Main page by clicking the login button and to the Registration page by clicking the register member button.

Left screen is a registration page implemented in RegistrationActivity.java.

It is a screen for receiving user's information and adding it to the database.

User can check if the ID entered already exists in the database by clicking the duplicate check button next to ID edittext.

If user wants to complete the registration, user should fill out the content according to all the subscription forms, do the duplicate check of the ID by clicking duplicate check button, and enter the password twice to check if the password is correct.

After completing all the forms, data will be saved at the database and user will be re-directed to the login page.

Left screen is a Main page which is implemented in MainActivity.java. User can enter to the main page by inputting the correct member information on the login page and clicking the login button. The structure of the main page consists of six buttons, a dosing time display and a blood glucose graph. When user clicks the buttons, user can move to pages implemented in GlucoseActivity.java, MedicationActivity.java, ExerciseActivity.java, MealActivity.java, and DiaryActivity.java.

The last button, the logout button, is a button for logout. If user clicks logout with automatic login, the user information saved as a cookie for automatic login disappears.

The dosing time shown in the middle of page, is the closest next dosing time. Client gets all dosing time from database, search the closest dosing time and show it.

The blood glucose graph is drawn through the blood glucose values stored in the user's database. The graph shows only last three day's values. It is displayed by dividing the time zone into morning, lunch, evening on each day. If there is more than 1 value for each time zone, the average value will be plotted on the graph. For graph, MPandroidchart open source is used.

When user clicks the blood glucose management button on the main page, the blood glucose list page appears as shown on the left. The blood glucose list page is a page implemented in GlucoseActivity.java, which shows the blood glucose information of the user stored in the database in a listview format. To implement listview, glucose_listview.java and glucose_listviewadapter.java files are additionally implemented

When user clicks the new blood glucose button on the blood glucose list page, the new blood glucose page appears as shown on the right. The New Blood Glucose Page is implemented in NewGlucoseActivity.java, which is a page user can write a new blood glucose value and memo.

After inputting the blood glucose value and memo, this information will be stored in database when user clicks save button. If user tries to save with empty value, failure message is displayed.

Medication Activity is an activity to manage dosing time.

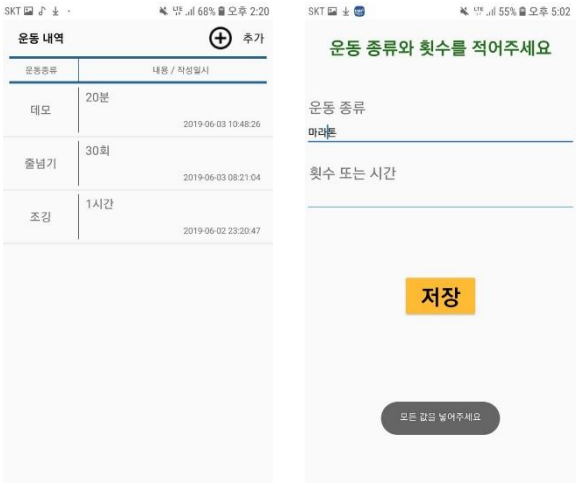
Medication Activity is divided into two parts.

First part is time picker part which is colored in red. User can set time using time picker spinner. When user clicks "Add" button, the time on the spinner is added to dosing time list.

Second part is listview part which is colored in orange. For listview, alarm_listview.java and alarm_listviewadapter.java files are implemented additionally. The client receives all the dosing time from database, and all items are added to listview adapter in the time order. Then the adapter is set to the listview. When user adds a new dosing time, new dosing time is added at the bottom of the list and sent to the database server. At this point, if the same

time already exists in the database, a toast message that notifies the existence of the same time is printed on the screen.

Also user can delete existing dosing time by clicking “Delete” button after selecting item to delete. When user clicks “Delete” button without selecting item to delete, a toast message is printed which says “Select item to delete”.



When user clicks the exercise management button on the main page, the exercise list page appears as shown on the left. The exercise list page is a page implemented in ExerciseActivity.java, which shows the exercise information of the user stored in the database in a listview format. Exercise_listview.java and exercise_listviewadapter files are additionally implemented to implement listview.

When user clicks the new exercise button on the exercise list page, the new exercise page appears as shown on the right. The New Exercise Page is implemented in NewExerciseActivity.java. User can input the type of exercise and value in this page.

After inputting the type of exercise and value, the information will be stored in database when user clicks save button. When user tries to save with empty value, failure message will be displayed.

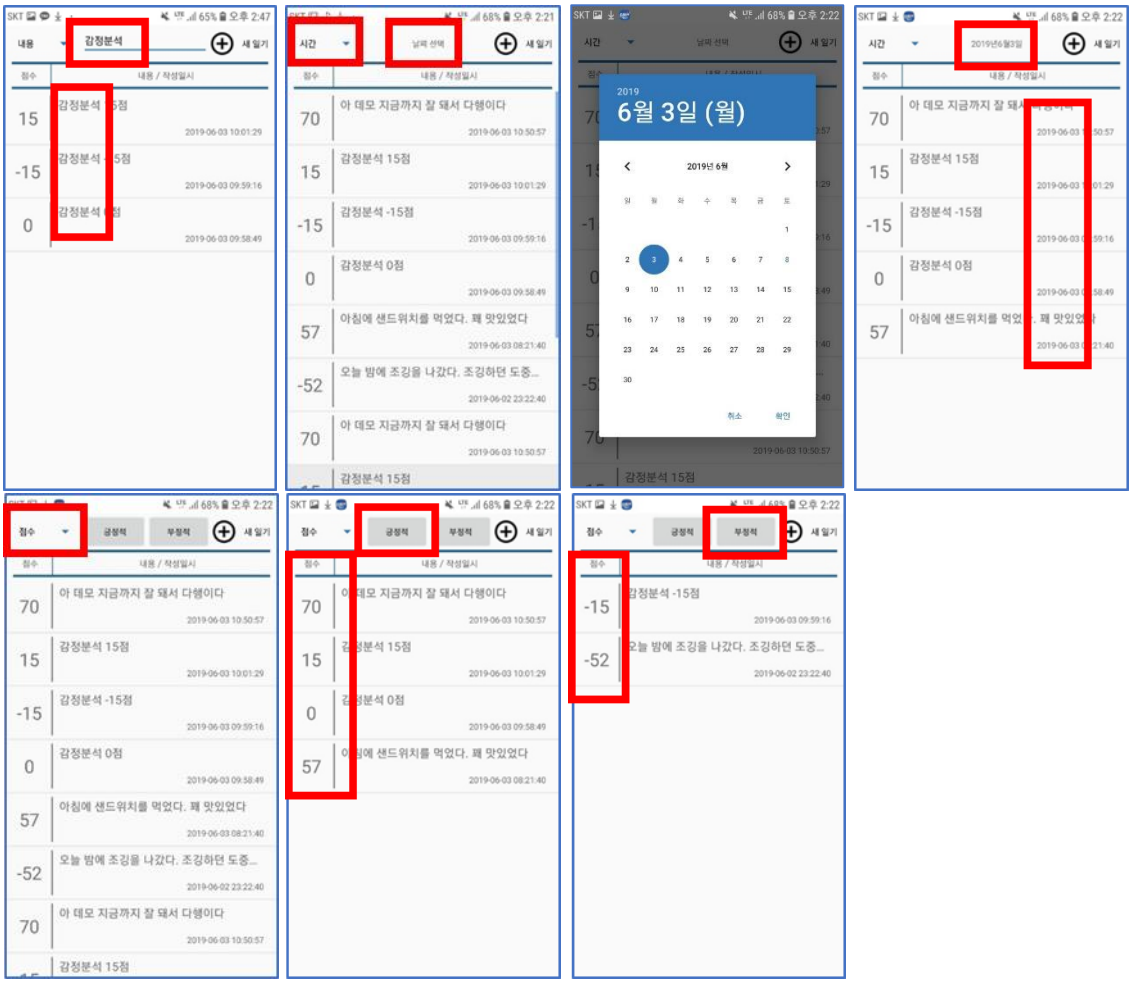
When user clicks the meal management button on the main page, the meal list page appears as shown on the left. The meal list page is a page implemented in MealActivity.java, which shows the meal information of the user stored in the database in a list format. Meal_listview and meal_listviewadapter files are additionally implemented for listview.

When user clicks the new meal button on the meal list page, the new meal page appears as shown on the right. The New Meal Page is implemented in NewMealActivity.java, which gets 5 kinds of food from user.

After inputting food, when user clicks the Save button, this information will be stored in database. When user tries to save with empty value, failure message will be displayed.

Diary Activity is an activity to show old diaries to user. Diary Activity is divided into two parts.

First part is a search bar part, which is colored in red. This part is implemented with spinner to provide user a function to search diaries with contents, written time and analysis score. In the case of searching with contents, if user input a specific text in the edittext at the top, only the diaries containing the text are searched. In the case of the searching with written time, date picker is used to select a specific date. Then the diaries which are written at the selected date are searched. In the case of searching with analysis score, user can click either two buttons, “Positive” button or “Negative” button. If user clicks “Positive” button, user can see the diaries which have 0 or more scores. Otherwise, user can see the diaries with negative score when user clicks.

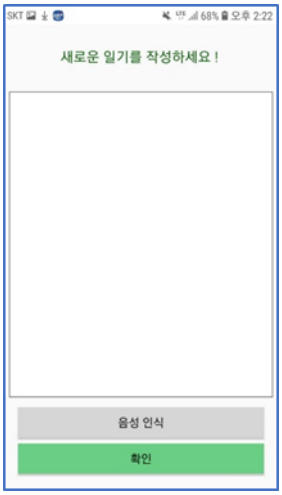


At this point, elements to show to user should be changed depending on which searching method is selected in the spinner. So, Frame layout is used to manage which elements should be visible or not.

Lastly, user can move to NewDiary Activity by clicking “New Diary” button at the top.

Second part is a listview part at the bottom, which is colored in orange color. For listview, diary_listview.java and diary_listviewadapter.java files are implemented additionally. The client receives all the diaries from database, and all items are added to listview adapter in the latest order. When user changes search method at the spinner, items which meet the condition are added to a new listview adapter and this new adapter is set to the listview.

Also, user can move to DiaryShow Activity by clicking an item in the listview. The information of the clicked diary is passed to DiaryShow Activity using intent.

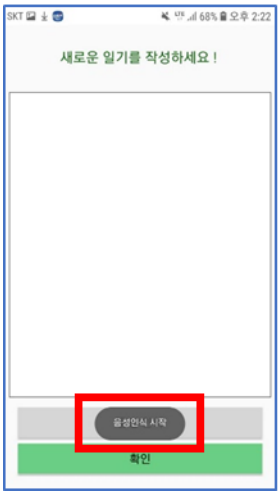


NewDiary Activity is an activity that user can write a new diary.

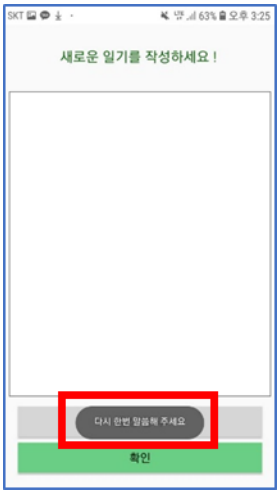
Basically, when user clicks “Confirm” button at the bottom after user inputs the diary, the diary is sent the database server and saved. After saving data, sentiment analysis is proceeded on the diary. At this moment, “Vader sentiment analysis open source” is used in server to analyze the diary. “Vader sentiment analysis open source” provides sentiment analysis function for English sentences. So, before sending diary to database server, the diary is sent to Naver Papago NMT server to be translated into English using “Naver Papago NMT API”. After translation, client can receive the result in Json format. In the process, when user tries to save an empty diary, an error occurs at parsing json object part. So, exception handling is done for this part.

To sum up, the whole process is like below.

1. Input diary
2. Translate the diary with Naver Papago NMT API
3. Send original and translated data to database server and save them
4. Proceed sentiment analysis with Vader Sentiment Analysis open source



Also, considering that target users are elder people, we implement voice recognition function using Google Speech API. When user clicks “Voice recognition” button and speak some words, these words are converted into text and added to edittext automatically. User can see a toast message which notifies the start of the recognition.



If any kinds of error occur, a toast message which request user to speak again is shown. For this function, onError method is modified to support the function.

DiaryShow Activity is an activity which shows a specific diary which is clicked at the listview in Diary Activity. When an item in listview is clicked, client gets the information of the item, and sends them to DiaryShow Activity using additional intents. After creating DiaryShow Activity, the addition intents, which are contents, written time and sentiment analysis score, are shown in the screen.

5.2 Data Storage of patients' data

The server is responsible for storing and retrieving users' data. When the users save the data, they are passed to the server. Then the server connects to the database, and executes the PHP scripts with SQL statements. When the user retrieves the data, the data to be retrieved are passed, then the server connects to the database, and executes the PHP scripts with SQL statements for the data. Then it passes the result sets to the clients. This is because it is not safe for the user application to directly be connected to the database because it has the root username, and password on the client.

The user entity has an unique ID(UserID), name, date of birth, email, phone number, user name, and password. The password is encrypted with SHA-256 for security. It is implausible to decrypt the password to original value. The user has diary, blood glucose, medication alarm, activity, and meal data. (Figure 3)

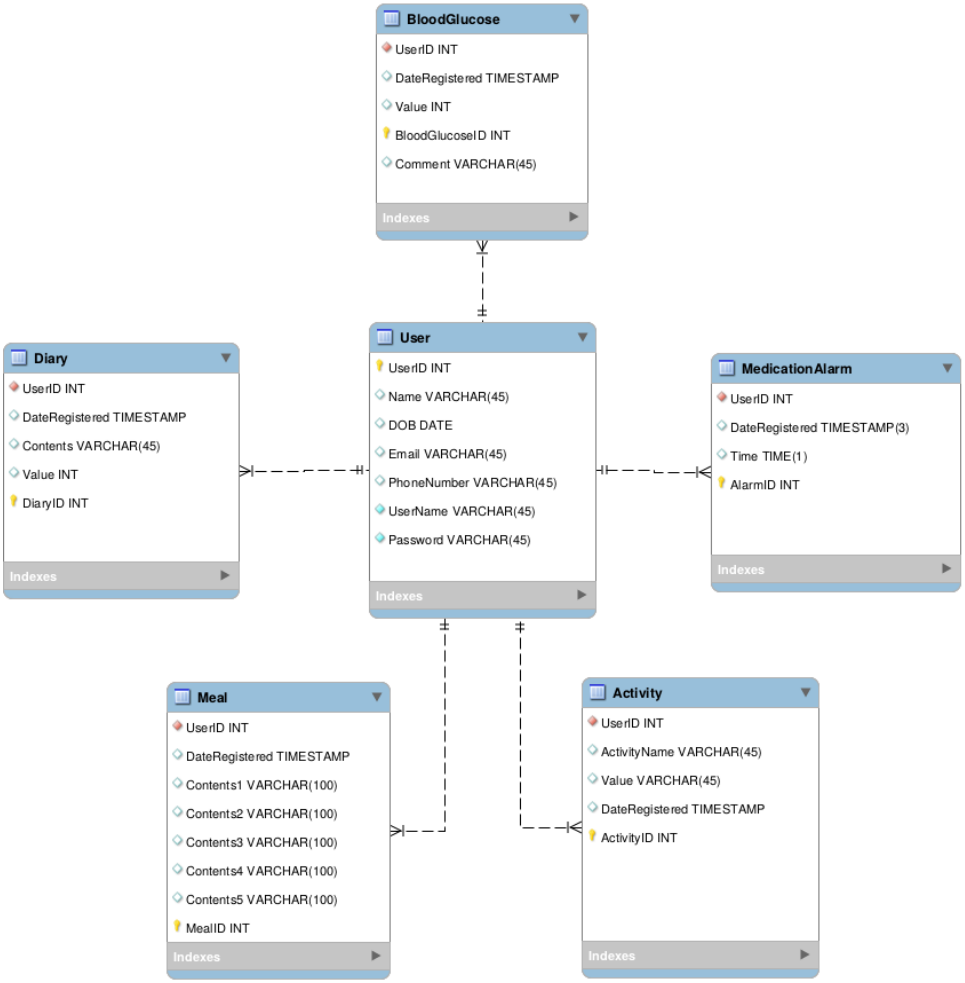


Figure 3 The ERD(Entity-Relationship Diagram) for the application

5.3 Sentiment Analysis

Before sending diary to database server, the diary is sent to Naver Papago NMT server to be translated into English using “Naver Papago NMT API”. After translation, client receives the result in Json format and parses Json object to get translated data. Then, original data and translated data are sent to database server. The process is like below. (Figure 4)

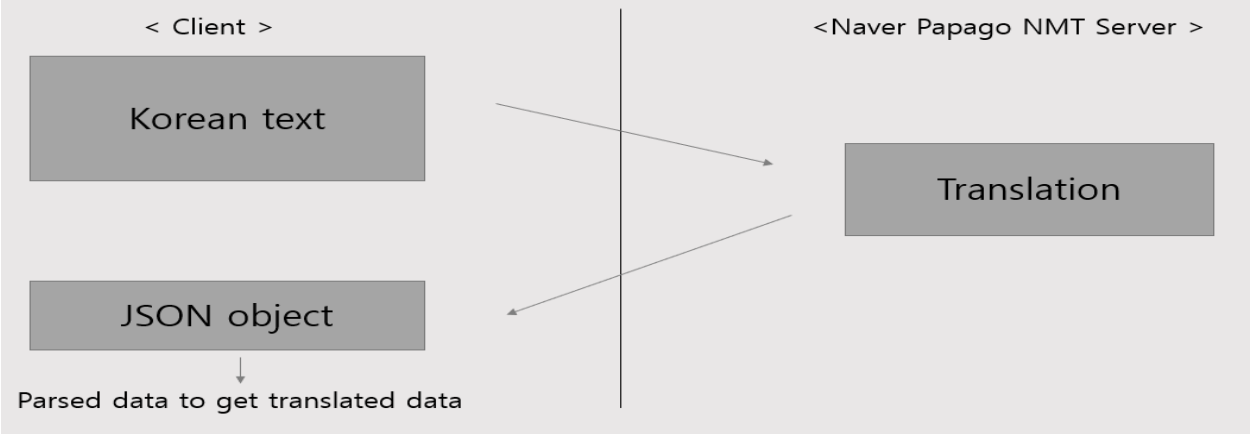


Figure 4 The process of translation

When database server receives the data from client, it passes the translated contents to the sentiment analyzer. It calculates the value (-100 ~ +100). The contents and value are stored in the database. (Figure 5)

The sentiment analysis function is implemented by using the existing VADER(Valence Aware Dictionary for Sentiment Reasoning). It is a sentiment analyzer developed by C. J. Hutto, and Eric Gilbert, Georgia Tech. It uses a combination of qualitative and quantitative methods to improve the current sentiment analysis. It gets training data from microblogs like Twitter. We have chosen it because its dataset is from there. [7] Some grammars (anonymous function, short array syntax, etc.), which are supported since PHP 5.5 are modified since the server-side PHP version is 5.2.

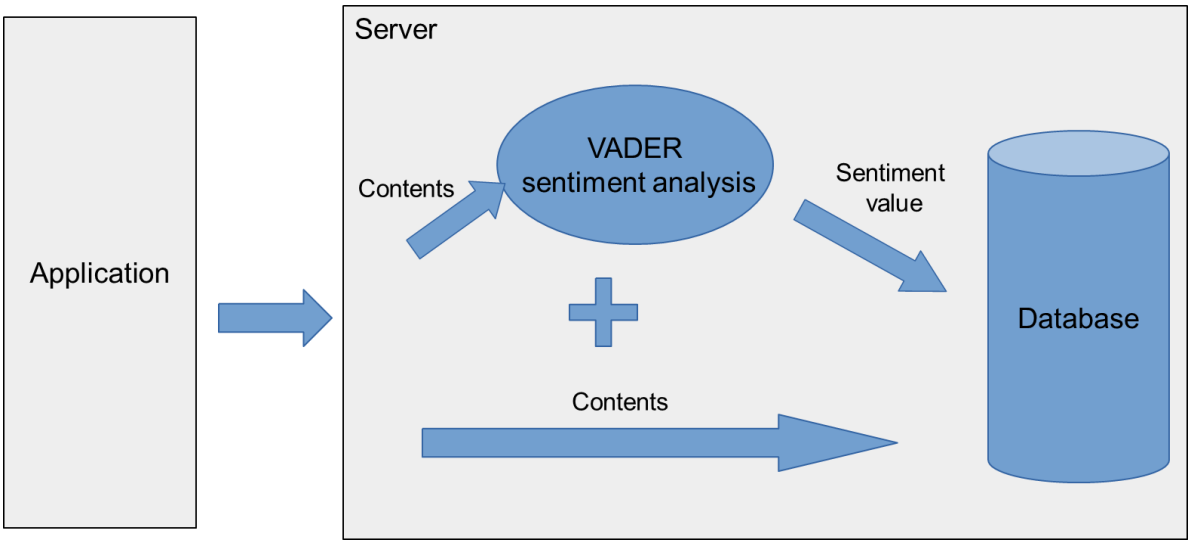


Figure 5 The process of sentiment analysis

6.0 Results

The following table describes the results.

Requirement	**Details**
#1 Platform	The application is built and tested on Android.
#2 User Interface	The application provides a convenient user interface for elderly patients.
#3 Storage	The application stores Blood glucose, diary, and sentiment values.
#4 Sentiment analysis	The application performs sentiment analysis based on users' diary data.
#5 Graph	The application plots graph of blood glucose by time.
#6 Search Items	The application has a search function.
#7 Security	The passwords is safely stored.

7.0 Budgets

Item	**Cost**	**Justification**
Android phone	KRW 0	We used the preexisting hardware.
Server	KRW 10,000	The server is used to save the data.
Android Studio	Free	It is used for developing the application
Visual Studio Code	Free	It is used for developing the server-side PHP.
VADER sentiment analysis	Free	It is used for sentiment analysis.

8.0 Task Division

Tasks are divided into mainly two parts: the application part, and the server part. Kim, and Sung is responsible for the application part. Nam is responsible for the server part. Kim mainly dealt with the user interface and graph function, and Sung's main responsibility was implementing java source codes for application. Nam managed the server and database.

All members were responsible for integrating modules, system troubleshooting, and testing.

9.0 Conclusion

9.1 Further work

This application has the fundamental features to manage diabetes and comorbid depression. It has tracking blood glucose, mood, exercise, diet.

It can be expanded into 2 aspects. Firstly, it can be connected to the hospital information system by using HL7 standard. [8] It can continuously monitor patients. Doctors, nurses, and psychologists can give them feedbacks using this system. Researchers can use those big data for their studies. And insulin pumps are connected to the application with Bluetooth for automatic BG values. Secondly, we can build a new feature that uses the users' data. They can be used as training sets for a blood glucose prediction model with moods and meals. An optimal BG prediction model can be built with those big data. (Figure 6)

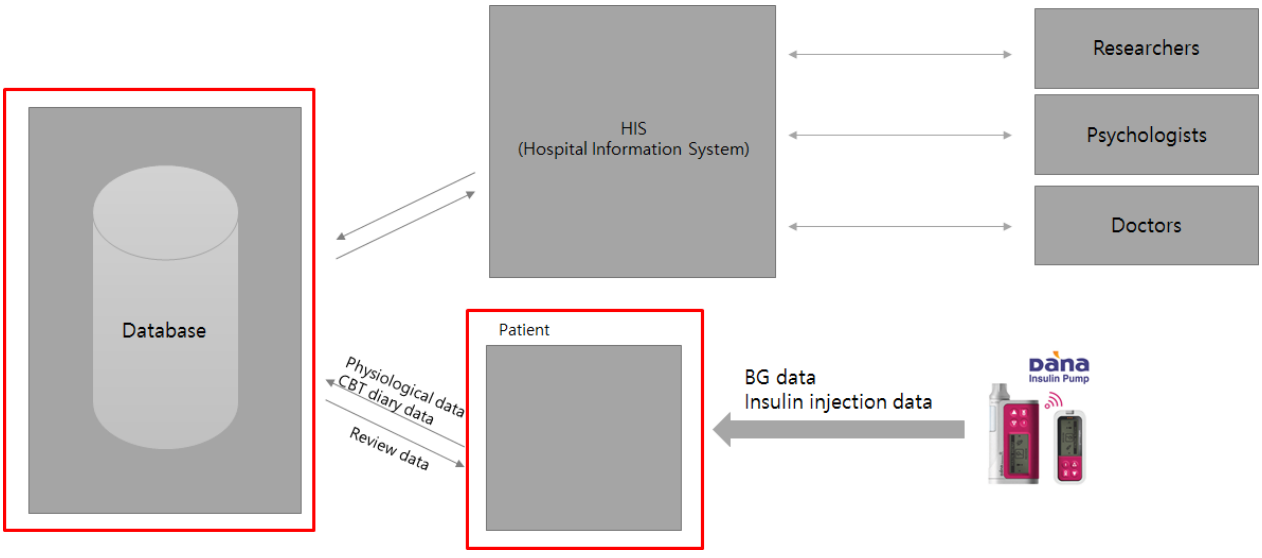


Figure 6 Suggested further works

For validating the efficacy of this application, we suggest a clinical trial. With professionals from other disciplines like medicine, nursing, and psychology, conducting a double-blinded clinical trials, where experiment group uses the application while control groups do not, we can show whether this application helps managing diabetes.

9.2 Summary & Significance

With the increase of patients with diabetes, there has been a growing demand for management of this long-term disease. The smartphones in the patients' hands can be used to continuously monitor their health. This application will benefit not only the patients themselves, and the society by reducing the costs.

9.3 Lessons Learned

Nam: The project was quite challenging, but it was a great opportunity to integrate my whole knowledge learned in school to solve the real-world problems. I have learned how to implement the solution that we proposed. Especially, I have acquired skills to manage the server, e.g. server-side script language(PHP), database system management(designing, normalization, SQL statements) etc. I also learned to utilize the open-source software using GitHub, and contribute by sharing solutions for our problems.
However, we could have done better if we managed out project better. In retrospect, we needed a project management tool like issue tracking system.

Sung: In this project, I focused on implementing diary-related functions. In this function, we implemented voice recognition function using Google Speech API. Although we expected that it would be easy to use API because it is provided by Google, but several errors occured while trying to apply API into our application. I could learn what kinds of error can occur and how to handle API-related errors. Also, this was my first time to implement android listview. I can learn the concepts of listview, adapter and other related things about listview.

Kim: In this project, I designed the UI by designing the user base of the elderly people. I was able to learn a lot because there was a lot of difficulties and considerations to design the UI considering the user layer. Using open source, it was harder to write in a format than I thought.

References

[1] American Diabetes Association. "Diagnosis and classification of diabetes mellitus." *Diabetes care* vol. 33 Suppl 1,Suppl 1 (2010): S62-9. doi:10.2337/dc10-S062

[2] 김대중, 이재혁, 원종철, 김보연, 김헌성, 박정환 (2018). Diabetes Fact Sheet in Korea 2018. 대한당뇨병학회

[3] 김수연, 이재호, 김하늘, 김동규, 나영, 김길선, ... & 송기호. (2009). 당뇨병 환자의 우울증과 자가 관리. 대한당뇨병학회지, 33, 432-438.

[4] 유승현. (2014). 당뇨병 관리를 위한 스마트폰 애플리케이션의 효과 연구. 이화여자대학교 대학원 석사학위 논문.

[5] Alanzi, T. M., Istepanian, R. S., & Philip, N. (2014, August). An integrated model for cognitive behavioural therapy for mobile diabetes self-management system. In 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (pp. 5393-5396). IEEE.

[6] Bădescu, S. V., Tătaru, C., Kobylinska, L., Georgescu, E. L., Zahiu, D. M., Zăgrean, A. M., & Zăgrean, L. (2016). The association between Diabetes mellitus and Depression. Journal of medicine and life, 9(2), 120–125.

[7] Hutto, C. J., & Gilbert, E. (2014, May). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Eighth international AAAI conference on weblogs and social media.

[8] Dolin, R. H., Alschuler, L., Boyer, S., Beebe, C., Behlen, F. M., Biron, P. V., & Shabo, A. (2006). HL7 clinical document architecture, release 2. Journal of the American Medical Informatics Association, 13(1), 30-39.