User Requirements Document

- for Diabetter, a Diabetes data dashboard

Team members

Alexandra Nikolova	- 1339311
Rik Litjens	- 1317059
Vasil Shteriyanov	- 1307282
Gabriela Zanova	- 1307509
Konstantin Velkov	- 1307436
Roel Koopman	- 1299743
Rinse Vlaswinkel	- 1312529
Nikaels Balasovs	- 1250221
Jeroen Gijsbers	- 1305832
Kevin Dirksen	- 1302191

Client & Platform Owner
M. Chaudron & P. Van Gorp

Project Managers
A. Pintea & C. Olteanu

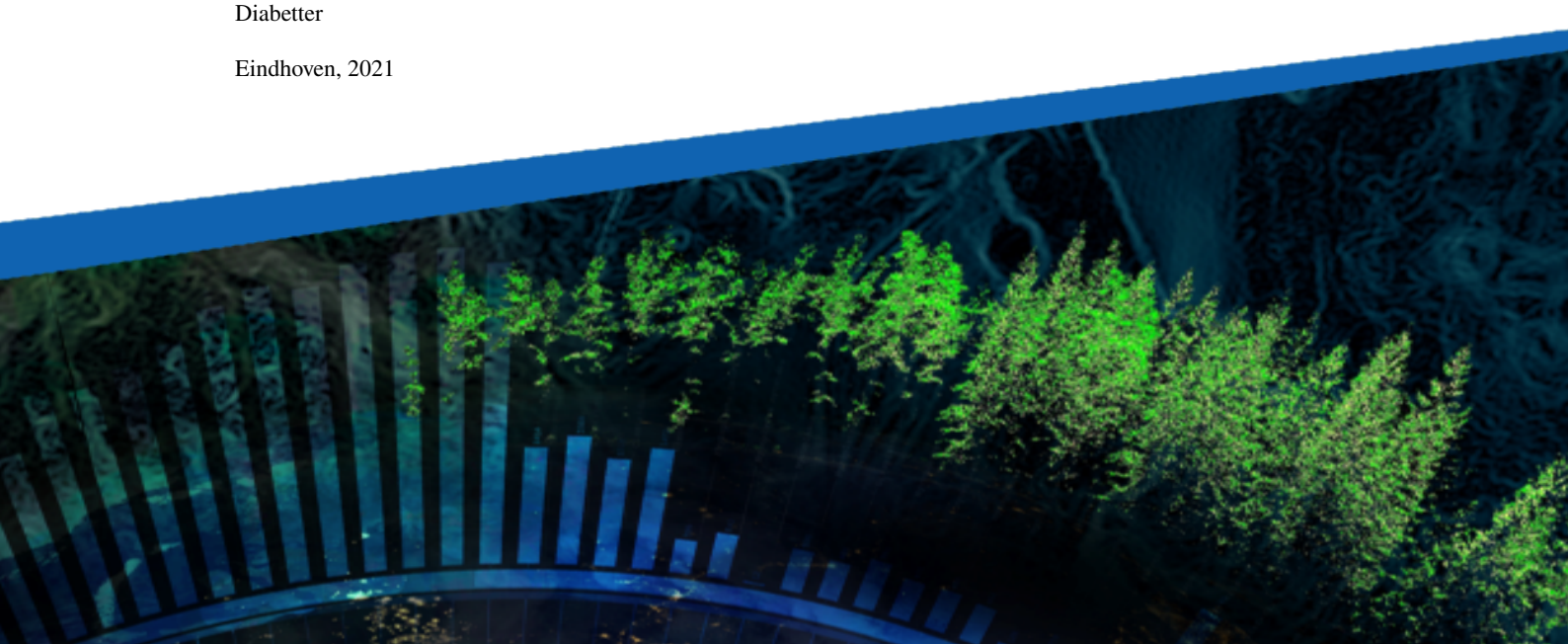
Supervisor
G. Kahraman

Eindhoven University of Technology

Department of Mathematics and Computer Science

Diabetter

Eindhoven, 2021



Abstract

This User Requirement Document (URD) describes all relevant requirements for Diabetter, a web-based data dashboard for people living with Diabetes. While using functionalities of the GameBus platform, it does not only give users insights into several aspects of their life but also enables analyses between them. This document is in line with the ESA standards [1].

Contents

1 Introduction **5**

1.1 Purpose . . . 5

1.2 Scope . . . 5

1.3 List of definitions and abbreviations . . . 5

1.3.1 Definitions . . . 5

1.3.2 Abbreviations . . . 7

1.4 List of references . . . 7

1.5 Overview . . . 8

2 General description **9**

2.1 Product perspective . . . 9

2.2 General capabilities . . . 10

2.3 General constraints . . . 12

2.4 User characteristics . . . 13

2.4.1 Normal user . . . 13

2.4.2 Supervisor . . . 14

2.5 Environment description . . . 14

2.6 Assumptions and dependencies . . . 15

2.6.1 Assumptions . . . 15

2.6.2 Dependencies . . . 15

3 Specific requirements **16**

3.1 Capability requirements . . . 16

3.1.1 General . . . 16

3.1.2 User Management . . . 16

3.1.3 Specified Time Frame . . . 18

3.1.4 Health Metrics . . . 18

3.1.5 Activity . . . 23
3.1.6 Food . . . 24
3.1.7 Emotion . . . 26
3.2 Constraint requirements . . . 27
3.2.1 Privacy & Security . . . 27
3.2.2 Performance . . . 28
3.2.3 Reliability . . . 28
3.2.4 Environment . . . 29
3.2.5 Data . . . 29

A Use cases **32**
A.1 User Management . . . 32
A.2 Specified Time Frame . . . 33
A.3 Health Metrics . . . 34
A.4 Activity . . . 48
A.5 Food . . . 57
A.6 Emotion . . . 68

B Client and Supervisor Approval **74**
B.1 Client approval . . . 74
B.2 Supervisor approval . . . 75

Document status sheet and change records

Document title	User requirements document
Document identifier	Diabetter.URD/4.0
Authors	Alexandra Nikolova Rik Litjens Vasil Shteriyanov Gabriela Zanova
Document status	Approved by clients

Document history

Version	Date	Authors	Description
0.0	26-04-2021	Vasil, Rik, Alexandra & Gabriela	Draft containing only requirements
1.0	02-05-2021	Vasil, Rik, Alexandra & Gabriela	Full draft of the URD document
2.0	06-05-2021	Vasil, Rik, Alexandra & Gabriela + comments from everyone	Ready to be signed version of the URD document
3.0	06-05-2021 3PM	Vasil, Rik, Alexandra & Gabriela + comments from everyone	Ready to be signed version of the URD document + incorporated feedback
4.0	04-06-2021	Vasil, Rik, Alexandra & Gabriela + comments from everyone	Changed document to include what the clients want

Change records

Version	Changes
0.0	Created initial set of requirements.
1.0	expanded URD to include the introduction, general description and use cases.
2.0	Added environment section, expanded general description, incorporated feedback into requirements and use cases, fixed layout.
3.0	Adaption to environment section w.r.t. Diabetter database, included GDPR compliance, added requirements about continous data, defined application.
4.0	Changes to emotion component to include 2D emotion, remove sending notifications, layout related requirements are changed to client needs and multiple clarifications

1 Introduction

1.1 Purpose

This User Requirements Document (URD) describes all user requirements of the Diabetter application, that have arisen from several meetings between the Diabetter project group, the client M. Chaudron and GameBus platform owner P. Van Gorp. Besides describing the main features and requirements in detail, all other relevant requirements are also documented and prioritized according to the MoSCoW method [2]. This document serves two important purposes. It will be used as a guideline for the development team during the project and serves as an agreement between all parties involved in the project. More specifically, all "Must-have"-requirements are to be implemented in the final product and the team will strive to add as many of the other requirements as possible, depending on available resources.

1.2 Scope

The Diabetter initiative will be designed and developed by a group of Computer Science and Engineering students from the Eindhoven University of Technology, in cooperation with the client and the platform owner. The aim of the project is to create a dashboard that integrates data of different important aspects in the life of people with Diabetes type 1 involving glucose levels, activity information, insulin intake, food intake, and emotional state. It will become a part of the GameBus platform, a platform that promotes a mentally and physically healthy lifestyle in a playful manner [3].

The dashboard shall be able to visualize data from the separate important aspects in a simple and intuitive way, but enabling analyses and showing possible interactions between these aspects will also be an important goal of the Diabetter dashboard. The users need to get insight into their glucose levels, but also into how food, insulin, activity, and emotion may influence these levels. Besides this, the integration with GameBus may enable more ways to encourage the users to live a healthier life with Diabetes.

1.3 List of definitions and abbreviations

Some of the used terms, phrases, and abbreviations might be ambiguous. Therefore we include all relevant definitions in Table 1 and 2.

1.3.1 Definitions

A1C estimate	Average blood glucose level over the past 3 months.
Activity entry	An activity that a user has done and has made a record in the web application.
Activity history	A collection of user activities.
Activity type	A subgroup of activities that includes similar activities, e.g. runs, walks.

Classification threshold of the hyperglycemia occurrences	The value above which, the application shall classify glucose level values as hyperglycemia occurrences.
Classification threshold of the hypoglycemia occurrences	The value below which, the application shall classify glucose level values as hypoglycemia occurrences.
Diabetes	When diabetes is mentioned, we refer to Diabetes type 1.
Editable data	Previously collected data for insulin, mood and profile settings that can be changed
Emotional history	A collection of users' emotional status inputs.
Emotional status entry	An emotion that a user has experienced and has made a record in the web application.
Event type	One of the following: A performed activity, an entered meal/snack (food), an entered emotion status.
Food entry	A food that a user has consumed and has made a record in the web application.
Food history	A collection of the users food entries.
Food information	Food details such as calories, carbs, name, time, date.
Food type	One of the following: breakfast, lunch, dinner, snack.
Glucose data scan	The user scans his CGM and reads the buffered glucose data, which is subsequently uploaded to the GameBus servers.
Glucose value types	Can be one of the following: "Low", "Very Low", "Normal", "High", "Very High". The application sets them by default, but the user can change them.
Health metrics	Glucose, insulin.
Hyperglycemia occurrence	The application interprets the Hyperglycemia occurrence as a glucose value above Classification threshold of the hyperglycemia occurrences.
Hypoglycemia occurrence	The application interprets the Hypoglycemia occurrence as a glucose value below Classification threshold of the hypoglycemia occurrences.
Insulin information	Indicates when the insulin was taken, whether it was rapid-acting or slow-acting, and its amount.
Insulin intake type	Can be one of the following: rapid-acting or slow-acting.
Intensity of an activity	A uniform measure of the level of physical strain during an activity.
Specified time frame	A time interval and a date interval selected by the user.
Summary statistics of activity data	Average heartbeat, sum of steps, burnt calories.
Summary statistics of food data	Sum of carbohydrates, calories.

Summary statistics of glucose data	Average, min, max.
Supervisor	A supervisor figure of a normal user with Diabetes. He can see and change the data of the normal user.
Supervise	See and change the data of the normal user.
The application	All systems designed and developed by the Diabetter team, ranging from the web page to the processing of data behind the scenes.
User	Normal user or Supervisor user.
Visualization	Graphs or diagrams.

Table 1: Definitions of terms that are used in this document

1.3.2 Abbreviations

CGM	Continuous glucose monitor
CSV	Comma Separated Values
e.g.	Exempli gratia, indicating an example
Hyper	Hyperglycemia
Hypo	Hypoglycemia

Table 2: Definitions of abbreviations that are used in this document

1.4 List of references

References

[1] E.B. for software Standardisation and Control. Esa software engineering standards. 1991.

[2] Moscow prioritization. https://www.productplan.com/glossary/moscow-prioritization/. Accessed: 04-05-2021.

[3] Gamebus. https://blog.gamebus.eu/. Accessed: 04-05-2021.

[4] Blood sugar testing: Why, when and how. https://www.mayoclinic.org/diseases-conditions/diabetes/in-depth/blood-sugar/art-20046628#:~:text=Your%20doctor%20may%20recommend%20blood,Before%20and%20after%20exercise. Accessed: 04-05-2021.

[5] Choosing the best continuous glucose monitoring (cgm) system for type 1 diabetes and type 2 diabetes. https://comprehensiveendo.com/

continuous-glucose-monitoring-systems-cgm#:~:text=There%20are%203%20common%20CGM,the%20Medtronic's%20Guardian%20Sensor%203.&text=The%20Dexcom%20G6%20is%20the,%2Dto%2Duse%20automatic%20applicator. Accessed: 04-05-2021.

[6] Why is regular exercise good for diabetes? https://www.diabetes.org.uk/guide-to-diabetes/enjoy-food/eating-with-diabetes/out-and-about/sports-nutrition-and-type-1-diabetes. Accessed: 04-05-2021.

[7] insulin and diabetes. https://www.diabetes.org.uk/guide-to-diabetes/managing-your-diabetes/treating-your-diabetes/insulin. Accessed: 04-05-2021.

[8] Abbott freestyle librelink. https://play.google.com/store/apps/details?id=com.freestylelibre.app.nl&hl=en&gl=US. Accessed: 06-05-2021.

[9] Diabetes and emotions. https://www.diabetes.co.uk/emotions/. Accessed: 05-05-2021.

[10] Mijn eetmeter. https://apps.apple.com/nl/app/mijn-eetmeter/id663382012. Accessed: 05-05-2021.

[11] David C. Rubin and Jennifer M. Talarico. A comparison of dimensional models of emotion: Evidence from emotions, prototypical events, autobiographical memories, and words. *Memory*, 17(8):802–808, 2009. PMID: 19691001.

[12] Complete guide to gdpr compliance. https://gdpr.eu/. Accessed: 05-05-2021.

[13] Stephanie Van Asbroeck and Christophe Matthys. Use of different food image recognition platforms in dietary assessment: Comparison study. *JMIR formative research*, 4(12):e15602–e15602, Dec 2020. 33284118[pmid].

[14] Calorie mama api. https://dev.caloriemama.ai/. Accessed: 05-05-2021.

1.5 Overview

The rest of the document will explain Diabetter in more detail. More specifically: Section 2 gives a general description of the Diabetter product, including its relation to other existing systems, its general capabilities and constraints, the characteristics of its different users, and the assumptions on which we base the requirements. Section 3 shows a comprehensive list of all user requirements together with their priorities. And finally, Appendix A shows all important use cases for the dashboard.

2 General description

2.1 Product perspective

Living with diabetes involves a lot of monitoring and special consideration for everyday things. Although numerous applications exist that help people with Diabetes with this, there is no fully-fledged dashboard that combines them all. This is what the Diabetter system does and what distinguishes it from other systems. This section discusses these other systems and how they are related to the Diabetter dashboard.

First and foremost, monitoring glucose levels is essential. People with Diabetes type 1 are generally recommended to test their blood sugars multiple times during the day [4]. Traditionally, this is done by spilling a drop of blood onto the test strip of a glucose meter, but later technological developments allowed people with Diabetes to wear a Continuous Glucose Meter (CGM). These devices, manufactured by Abbott, Medtronics, Dexcom [5] and others, monitor the glucose levels of users 24/7 and therefore allow enhanced insight into glucose levels. The Diabetter dashboard aims to provide these insights by processing CGM data.

Exercise or activity may also be important in the life of someone with Diabetes. Besides the usual benefits of exercise such as lowering blood pressure, cholesterol or stress levels, exercise can help people with Diabetes to lower their glucose levels as well [6]. This effect can differ between several types of activity and the situation, hence monitoring them can be beneficial. To monitor activities, one can keep track of them manually or with the use of activity applications on a phone or on a sports watch, such as Google Fit or Fitbit. However, we believe it is beneficial to the user to have both the glucose data and the activity data presented together and show their correlations. Therefore, we use activity data from several sources and include them all in the Diabetter dashboard.

Many existing Diabetes tracking applications allow the user to enter information about insulin intake. Insulin helps to reduce blood glucose levels by making the body turn it into energy. The insulin can either act rapidly or more slowly over the long term [7]. The dashboard aims to use all available information about insulin either when it is given as input directly or when it is part of exported data from other applications such as Abbott's Freestyle Librelink [8].

Another important feature to be monitored by people with Diabetes is their food intake. Which and how much food is eaten determines how much carbohydrates the body receives. These carbohydrates affect the glucose levels in their blood and therefore considering what to eat is important to prevent low or high glucose levels. One way of keeping track of what you eat is by putting it into an Excel sheet. A more advanced way of understanding what you eat, that does not involve much effort, involves uploading a picture of the food to a machine learning model, which in turn returns information about its nutrients. The Diabetter dashboard will allow the user to upload food intake in several ways and will in turn present useful overviews and correlations.

The final factor that we will be implementing into the dashboard is the emotional aspect of living with Diabetes. Feelings of displeasure, depression, tension, and fatigue have been found to be related to higher A1C levels [9], a measure for glucose level over the long term. Current mood trackers are often unrelated to Diabetes and existing Diabetes tracking applications, such as the Abbott Freestyle Librelink [8], do not offer mood tracking at all. With Diabetter we want to integrate emotion tracking in order to provide insights into how it correlates with a user's glucose level.

Ultimately, the main system to which the Diabetter dashboard will be related is GameBus. The dashboard

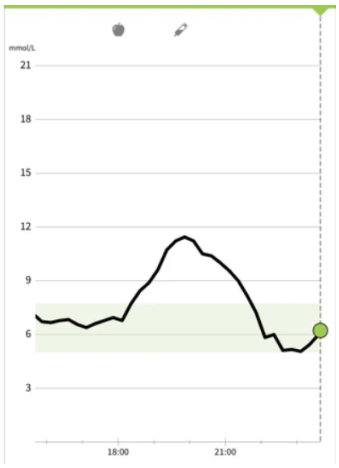
will use several functionalities of GameBus for storing and querying glucose data, food data, insulin data, and emotion data. The activity data will be directly extracted from GameBus since it is already able to process data from several previously mentioned fitness applications (e.g. Google Fit and Fitbit). Furthermore, users will be able to log in to Dashboard using a GameBus account. On the other side, a user shall also be able to view some data and events of the Diabetter dashboard in their GameBus application, such as daily aggregates for the glucose data.

2.2 General capabilities

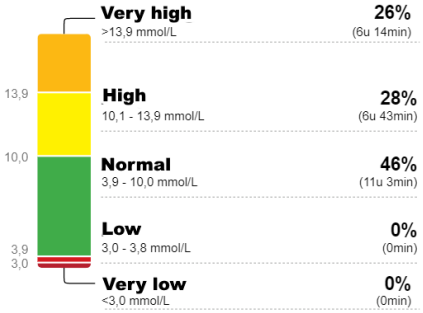
This section will describe the most important capabilities of the dashboard. User related requirements are described in section 2.4.

Glucose and other blood values

The dashboard will visualize the user's glucose level over a certain time frame that can be manually adjusted. For example, when the user wants to know more about his glucose values a few hours ago, the user can select a time frame beginning 24 hours ago until now and see his glucose levels for this time period. Filtering on other events such as meals and activities (as mentioned later) are also possible. Figure 1a shows a very basic example of such a plot taken from the Abbott Libre application. The latest glucose measurements will be accompanied by a trend line that indicates predicted future glucose levels. However, the user might also want to see what happened with his glucose levels in the evenings during the last month. For this, the dashboard contains an option to view summary data of a certain time slot within the day over a longer time period such as a month or year. The dashboard also shows the user what percentage of the time his levels were very low, low, normal, high, or very high during a time frame that the user can select, like in the concept Figure 1b.



(a) Glucose plot over a specified time frame [8].



(b) Visualization showing the percentage of time the user's glucose level fell within a certain range during a specified time frame.

Figure 1: Example plots that are similar to plots that will be on the Diabetter dashboard

In case of hyperglycemia or (more importantly) hypoglycemia, the dashboard will help the user in finding out what might have been the cause of this by visualizing the activity, food, insulin, and emotion in the

two hours preceding this hyper or hypo. The occurrences of both hypos and hypers will be logged and available to be seen by the user.

Based on this available glucose data, the dashboard will estimate A1C levels for the past three months and present them to the user. Besides this, insulin intake can be entered and visualized alongside the glucose level.

Food

The dashboard maintains a list of all meals and snacks the user has published to the dashboard, to the Abbott application, and to the Eetmeter application [10]. The user can enter food data by uploading an Excel sheet containing food data to the dashboard or a synchronized Drive. Furthermore, the user may be able to enter food data directly via the dashboard. When the user wants to know how his or her glucose levels acted after eating a certain meal, the user can select it and the dashboard shall show glucose levels around this meal. To aid the user in finding meals, he or she can filter the list and search by the name that was given to it. If a mistake was made, the user can always edit or delete entries. Furthermore, the dashboard shall be able to fill in some missing values in the Excel sheet, such as date, time (based on meal type) and total insulin (if sub-components of insulin intake are present).

Activity

As with the food intake, the dashboard maintains a list of activities. The user can select these activities in order to see his or her glucose levels around the time frame of the corresponding activity. The user will also be able to search activities in this list by name and filter by properties such as date and type.

The user will be able to plot available heartbeat data with a glucose chart to quickly see any correlations. Besides heartbeat, the user will be able to plot a histogram of the number of taken steps and the number of burnt calories over the chart as well.

Emotion

The dashboard will ask the user about their emotional status with a 2-dimensional emotion test [11]. The first dimension is the valence aspect of their emotion (i.e. positive or negative) and the second is the arousal aspect of it (i.e. calm or excited). To make this concept more user-friendly, the user shall be given two questions like the following: "How happy are you?" and "How excited are you?" as seen in the preview in Figure 2.

The user will be able to enter the emotional status when he or she accesses the dashboard. When this has not been done for several hours, a GameBus notification will be sent out asking the user to enter it. This notification will be shown to the user via the GameBus application. If the user does not enter emotion data for an even longer time period, any existing supervisor (see section 2.4) will receive an email to inform them about this.

These emotional statuses will be logged and users will be given the option to see what their glucose levels were like when they felt a certain way.

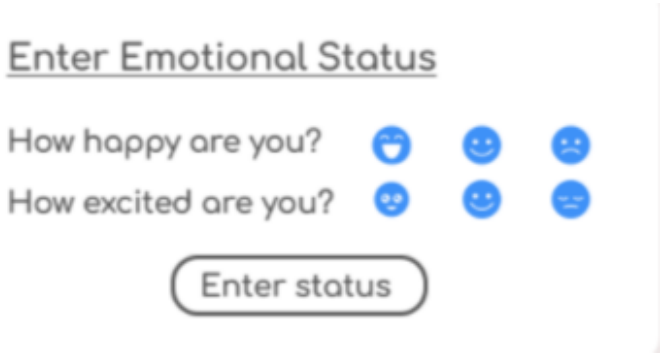


Figure 2: The questions to measure someone's emotion.

2.3 General constraints

Privacy & Security

Due to the big amount of data that is used for the dashboard and its sensitivity, maintaining the privacy of the user is an absolute must. Proper authentication mechanisms will ensure that unauthenticated users will be denied access to a dashboard and to the data behind it. To be authenticated, users have to use an existing GameBus account or create a new one and log in with their credentials.

The handling of personal data also comes with great responsibility. Therefore, the dashboard shall comply with GDPR standards [12]. Additional documents that are needed for this will also be created.

Performance

The dashboard should be interactive enough for the user to quickly check the data he is looking for. Long response times heavily impact user experience and are to be avoided. Therefore we believe most basic requests should be loaded within at least five seconds. However, we aim to bring this down to three or even one second. More specific requirements can be seen in the corresponding section 3.

Reliability

Although the dashboard does not provide advice or recommendations to the user, the presentation of data can influence the user to act a certain way. Therefore it is very important that the presented data is correct. Furthermore, it should run without crashing or showing errors in order to guarantee a consistent user experience.

Environment

The dashboard will reside on a website, that must at least work as intended on Google chrome version 90 or higher. However, we aim to include older versions of Chrome and other browsers as well. Furthermore,

the dashboard shall be connected to the GameBus platform.

Data

The dashboard has to be able to use data from a range of sources. For the glucose data, it will have to retrieve data from a .csv with data from the Abbott application that users can request. Furthermore, we aim to process data from other CGMs. The activity data shall be acquired from GameBus, as well as some food data. Other food data can be given as input via an Excel sheet, via the dashboard directly, or can be extracted from the previously mentioned .csv file. Furthermore, the user will be able to upload exports from the Eetmeter application. The dashboard will most probably be unable to automatically extract food information from a picture since any API of CalorieMama or others is either unable to do this accurately or too expensive [13] [14], although we strive to find alternatives if the available resources allow us to. The insulin data will be manually entered via the dashboard or extracted from mentioned Abbott .csv file as well. Emotional data will almost exclusively be acquired via the dashboard, although possible integration with the Wearable Tizen app and other applications will be considered.

Some of the metrics related to activity need to be available 24/7. That is why heartbeat data and data about burnt calories should be extracted at most five-minute intervals, although we aim to bring this down to one minute.

Furthermore, food data will be automatically extracted from Excel files that reside on a OneDrive if the user specifies this.

2.4 User characteristics

This section describes the intended users and their role within this web application.

2.4.1 Normal user

Normal users are people with Diabetes Type 1, who take advantage of the system to view and explore their data, which in return allows them to manage their diabetes easier. The normal users can see visualizations showing correlations between their glucose level and other metrics affecting it, including activity level, emotional level, as well as food consumption. Moreover, they can also explore these metrics individually. The following points further describe the behavioral aspects of a normal user using Diabetter.

Activity & Emotion
- View data automatically generated by external devices (for activities only)
- Enter data manually
- View manually entered data
- Edit and delete data

Food
- Enter food consumption manually
- View entered data
- Edit and delete food consumption data

Normal users can also give and revoke permission to other users to be their supervisors.

2.4.2 Supervisor

The role of the supervisor is to observe the health of the user they are supervising, as well as further provide health, activity, emotion, and food consumption data when they consider needed. This allows for parents, doctors, and other caretakers to analyze and improve the condition of their normal users. The following points describe the behavioral aspects of a supervisor using Diabetter.

- View the data of the normal user they supervise
- Enter health metrics/emotion/food information for the normal user they supervise
- Edit health metrics/emotion/food information of the normal user they supervise
- Delete health metrics/emotion/food information of the normal user they supervise
- See the normal user they supervise

2.5 Environment description

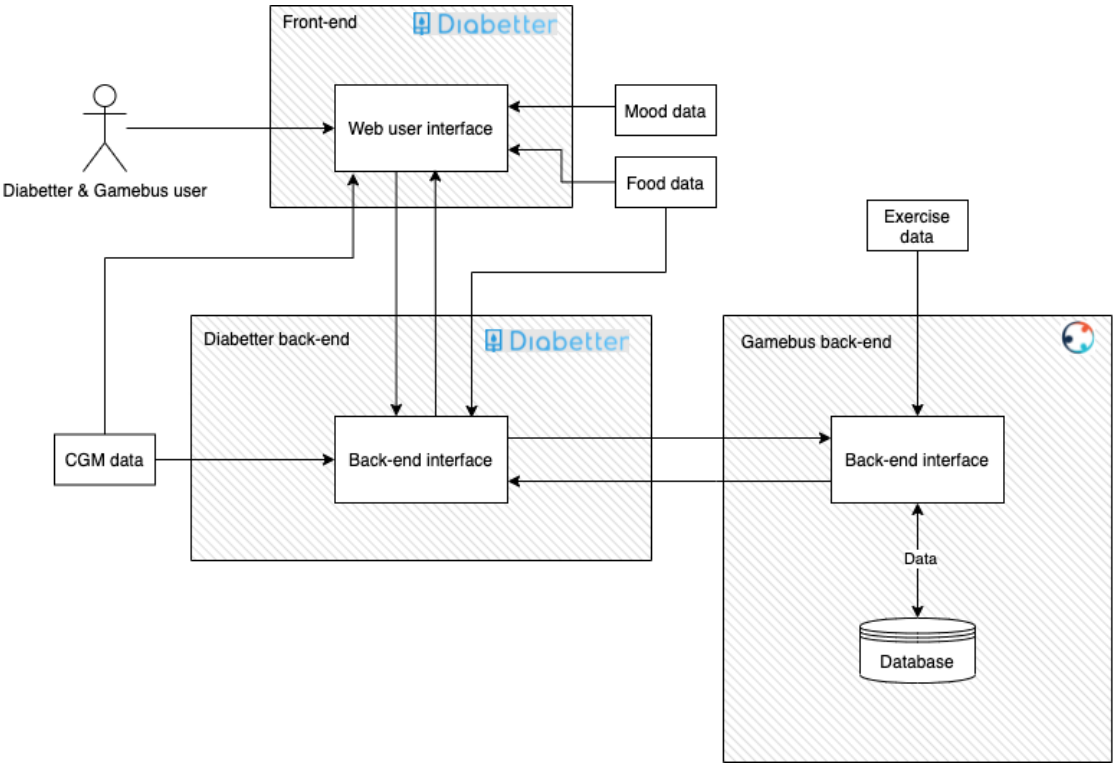


Figure 3: Environment model of Diabetter.

A model of the Diabetter environment is given in Figure 3. The main components of the system are the front-end and back-end of Diabetter as well as the back-end of GameBus. A user will interact with

Diabetter via a web application that can be run in their browser.

The front-end is used as a dashboard to view and summarize data and as an upload tool with which the user is able to add additional relevant information to his account. Authentication for Diabetter is executed through the services provided by GameBus and handled through both back-ends. The back-end of Diabetter handles parsing the uploaded data and handing it over to GameBus back-end for storage. GameBus stores information not only passed by Diabetter but also from devices that have offloaded their information through their own applications on the user's account or such that has been entered by the user directly. Hence the Diabetter back-end also extracts information from GameBus, analyzes it, and structures it for being suitably presented to the end-user.

The Diabetter database will be used as little as possible. It will only be (temporarily) used in the authentication process, while the rest should be sent to and stored by GameBus. In the future, we will explicitly ask for exemptions if we want to store additional information in the Diabetter database.

2.6 Assumptions and dependencies

In this section, we address the assumptions and dependencies of the requirements for Diabetter.

2.6.1 Assumptions

To function according to its specification, Diabetter requires the following statements to be true:

- The GameBus API is robust and acts according to its specification.
- The user has professional working proficiency in English.
- The database system is available at all times.
- The user uses the application as intended.
- The server is fully functional and accessible at all times.
- The data that comes from the Abbott Freestyle Librelink application is correct.
- The data that comes from the Eetmeter application and the user's Excel file is correct.
- The data that comes from user input is correct.

2.6.2 Dependencies

To function according to its specification, Diabetter depends upon the following:

- Fully functioning GameBus application.
- External server for hosting Diabetter.

3 Specific requirements

In this section, all requirements and their priorities are presented. The prioritization is done using the MoSCow method [2], which is shortly described in Table 3. We also added our own priority to it. The Diabetter development team will adhere to these requirements and priorities and implement all "must haves".

M	Must Have	These requirements are vital and have to be implemented in the final product
S	Should Have	These requirements are important but non-essential and have a high priority of being incorporated into the final product
C	Could Have	These requirements are nice additions to the product that can be implemented when there are available resources.
W	Won't Have	These requirements will not be included in the product, but they might be considered in the future.
TBR	To be researched	We want to do this, but we are not certain if this is possible. For this, research has to be done.

Table 3: Prioritization according to the moscow method

3.1 Capability requirements

Capability requirements state what the system must/should/can be able to do from a user perspective.

3.1.1 General

ID	Requirement	Priority
GEN 0.0	The application shall be a web application	M
GEN 0.1	The application shall support visualizations	M
GEN 0.2	The application shall be a native Android app	W
GEN 0.3	The application shall be a native iOS app	W

3.1.2 User Management

User Management - General

ID	Requirement	Priority
USER 0.0	The application shall have a user profile section.	M
USER 0.1	The application shall support two types of users: normal and supervisor.	M

User Management - Normal user

ID	Requirement	Priority
USER 1.0	A normal user can give permission to GameBus users to become his supervisor users.	M
USER 1.1	A normal user can revoke permission to GameBus user to be his supervisor user.	M

User Management - Supervisor user

ID	Requirement	Priority
USER 2.0	The supervisor can supervise at least one normal user.	M
USER 2.1	The supervisor can see the data of the user they supervise.	M
USER 2.2	The supervisor can modify the data of the user they supervise.	M
USER 2.3	The supervisor can see which normal user he supervises.	S
USER 2.4	The supervisor can supervise multiple normal users.	C

User Management - User Profile Properties

ID	Requirement	Priority
USER 3.0	The user profile shall have weight as a property.	S
USER 3.1	The user profile shall have age as a property.	S
USER 3.2	The user profile shall have height as a property.	S

User Management - User Profile Overview

ID	Requirement	Priority
USER 4.0	The application shall display the user's current weight.	S
USER 4.1	The application shall display the user's current age.	S

USER 4.2	The application shall display the user's current height.	S
USER 4.3	The application shall visualize the user's weight over time.	C
USER 4.4	The application shall display the user's weight goal.	C

User Management - User capabilities

ID	Requirement	Priority
USER 5.0	The user can change his editable data in the Diabetter application.	M
USER 5.1	The user can view blood glucose, emotion, insulin, carbs, and activity visualizations on the application.	M
USER 5.2	The user can set the weight in the user profile.	S
USER 5.3	The user can set the age in the user profile.	S
USER 5.4	The user can set the height in the user profile.	S
USER 5.5	The user can set his weight goal.	C

3.1.3 Specified Time Frame

ID	Requirement	Priority
STF 0.0	The user can select a start time to set a specified time frame for a visualization.	M
STF 0.1	The user can select an end time to set a specified time frame for a visualization.	M
STF 0.2	The user can select a start date to set a specified time frame for a visualization.	M
STF 0.3	The user can select an end date to set a specified time frame for a visualization.	M

3.1.4 Health Metrics

Health metrics - Properties

ID	Requirement	Priority
HM 0.0	A blood glucose value shall have a glucose type property.	M
HM 0.1	A blood glucose value shall have a value property.	M
HM 0.2	A blood glucose value shall have a date property.	M

HM 0.3	A blood glucose value shall have a time property.	M
HM 0.4	An insulin intake shall have an insulin intake type property.	M
HM 0.5	An insulin intake shall have a value property.	M
HM 0.6	An insulin intake shall have a date property.	M
HM 0.7	An insulin intake shall have a time property.	M

Health metrics - Overview and statistics

ID	Requirement	Priority
HM 1.0	The application shall display glucose visualizations with 5 min intervals by default.	M
HM 1.1	The application shall display glucose levels in the mmol/L unit.	M
HM 1.2	The application shall display the time of the last retrieved glucose data input.	S
HM 1.3	The application shall provide an option for the user to overlay visualizations.	S
HM 1.4	The application shall display statistics about the jitter in glucose values in a specified time frame, set by the user.	S
HM 1.5	The application shall display the A1C estimate of the user for the last 3 months.	S
HM 1.6	The application shall display the A1C estimate goal of the user.	S
HM 1.7	When the user selects a specified time frame that is longer or equal to 3 months, the application shall display the A1C level for the last 3 month of the selected time frame.	S
HM 1.8	The application shall display summary statistics on the user's glucose data over a specified time frame, set by the user.	S
HM 1.9	The application shall display the insulin intake of the user for the last 24 hours.	S
HM 1.10	The application shall provide an option for the user to display visualizations side-by-side.	C
HM 1.11	The application shall display glucose levels in the mg/dL unit.	C
HM 1.12	The application shall provide the option to change the glucose level units.	C

Health metrics - Visualizations

ID	Requirement	Priority
HM 2.0	The application shall visualize the blood glucose levels of the user over a specified time frame, set by the user.	M
HM 2.1	The application shall visualize the percentage of time the user's blood glucose values were "Very High" for a specified time frame, set by the user.	M
HM 2.2	The application shall visualize the amount of time the blood glucose values were "Very High" for a specified time frame, set by the user.	M
HM 2.3	The application shall visualize the percentage of time the user's blood glucose values were "High" for a specified time frame, set by the user.	M
HM 2.4	The application shall visualize the amount of time the blood glucose values were "High" for a specified time frame, set by the user.	M
HM 2.5	The application shall visualize the percentage of time the user's blood glucose values were "Normal" for a specified time frame, set by the user.	M
HM 2.6	The application shall visualize the amount of time the blood glucose values were "Normal" for a specified time frame, set by the user.	M
HM 2.7	The application shall visualize the percentage of time the user's blood glucose values were "Low" for a specified time frame, set by the user.	M
HM 2.8	The application shall visualize the amount of time the blood glucose values were "Low" for a specified time frame, set by the user.	M
HM 2.9	The application shall visualize the percentage of time the user's blood glucose values were "Very Low" for a specified time frame, set by the user.	M
HM 2.10	The application shall visualize the amount of time the blood glucose values were "Very Low" for a specified time frame, set by the user.	M
HM 2.11	When the user selects a specified time frame and there is no health metrics data available, the application shall display that there is no health metrics data.	M
HM 2.12	The application shall visualize the glucose level trend on the visualization of the glucose values.	S
HM 2.13	The application shall visualize an overview of hypoglycemia occurrences over a specified time frame, set by the user.	S
HM 2.14	The application shall visualize an overview of hyperglycemia occurrences over a specified time frame, set by the user.	S
HM 2.15	When the user selects a hypoglycemia or hyperglycemia occurrence, the application shall visualize the user's preceding 72 hours of activity.	S

HM 2.16	When the user selects a hypoglycemia or hyperglycemia occurrence, the application shall visualize the user's preceding 72 hours of food consumption.	S
HM 2.17	When the user selects a hypoglycemia or hyperglycemia occurrence, the application shall visualize the user's preceding 72 hours of insulin dosage.	S
HM 2.18	When the user selects a hypoglycemia or hyperglycemia occurrence, the application shall visualize the user's following 24 hours of activity.	S
HM 2.19	When the user selects a hypoglycemia or hyperglycemia occurrence, the application shall visualize the user's following 24 hours of food consumption.	S
HM 2.20	When the user selects a hypoglycemia or hyperglycemia occurrence, the application shall visualize the user's following 24 hours of insulin dosage.	S
HM 2.21	The application shall visualize the insulin intake of the user for a specified time frame.	S
HM 2.22	The application shall distinguish between the rapid-acting and the slow-acting insulin intake of the user in the insulin visualization.	S

Health metrics - Default values

ID	Requirement	Priority
HM 3.0	The application shall set the default threshold of the "Very High" blood glucose values to above 13,9 mmol/L.	M
HM 3.1	The application shall set the default range of the "High" blood glucose values in the range 10,1-13,9 mmol/L.	M
HM 3.2	The application shall set the default range of the "Normal" blood glucose values in the range 3,9-10,0 mmol/L.	M
HM 3.3	The application shall set the default range of the "Low" blood glucose values in the range 3,0-3,8 mmol/L.	M
HM 3.4	The application shall set the default threshold of the "Very Low" blood glucose values to below 3,0 mmol/L.	M
HM 3.5	The application shall set the classification threshold of the hypoglycemia occurrences to below 4.0 mmol/L.	S
HM 3.6	The application shall set the classification threshold of the hyperglycemia occurrences to above 10.0 mmol/L.	S
HM 3.7	The application shall set the default A1C estimate goal of the user to below 7 percent.	S

Health metrics - User Capabilities

ID	Requirement	Priority
HM 4.0	The user can upload glucose data using the application.	M
HM 4.1	The user can modify the threshold of the "Very High" blood glucose values.	M
HM 4.2	The user can modify the range of the "High" blood glucose values.	M
HM 4.3	The user can modify the range of the "Normal" blood glucose values.	M
HM 4.4	The user can modify the range of the "Low" blood glucose values.	M
HM 4.5	The user can modify the threshold of the "Very Low" blood glucose values.	M
HM 4.6	The user can modify the classification threshold of the hyperglycemia occurrences.	S
HM 4.7	The user can modify the classification threshold of the hypoglycemia occurrences.	S
HM 4.8	The user can select a hyperglycemia occurrence from the hyperglycemia occurrences visualization.	S
HM 4.9	The user can select a hypoglycemia occurrence from the hypoglycemia occurrences visualization.	S
HM 4.10	The users can modify their A1C estimate goal in the application.	S
HM 4.11	The user can enter insulin intake in the application.	S
HM 4.12	When the user enters an insulin intake, the user can enter the amount of insulin.	S
HM 4.13	When the user enters an insulin intake, the user can set the insulin type to slow-acting or rapid-acting.	S
HM 4.14	When the user enters an insulin intake, the user can enter the time of the intake.	S
HM 4.15	When the user enters an insulin intake, the user can enter the date of the intake.	S
HM 4.16	The user can edit an insulin intake input in the application.	S
HM 4.17	The user can delete an insulin intake input from the application.	S
HM 4.18	The user can visualize the glucose level trend on the glucose visualization.	S
HM 4.19	The user can hide the glucose level trend on the glucose visualization.	S
HM 4.20	The user can change the glucose level units.	C

3.1.5 Activity

Activity - General

ID	Requirement	Priority
ACT 0.0	The application shall have the same activity types as GameBus	S
ACT 0.1	The application shall display the user's burnt calories for the current date.	C
ACT 0.2	The application shall display the user's burnt calories goal.	C

Activity - Properties

ID	Requirement	Priority
ACT 1.0	An activity shall have a name.	M
ACT 1.1	An activity shall have a starting date.	M
ACT 1.2	An activity shall have an ending date.	M
ACT 1.3	An activity shall have a starting time.	M
ACT 1.4	An activity shall have an ending time.	M
ACT 1.5	An activity shall have a duration.	S
ACT 1.6	An activity shall have an activity type.	S
ACT 1.7	An activity shall have a description of maximum 50 words.	S
ACT 1.8	An activity shall have information about its intensity.	C
ACT 1.9	An activity shall have information about heartbeat rates during the activity.	C
ACT 1.10	An activity shall have information about steps taken during the activity.	C
ACT 1.11	An activity shall have information about burnt calories during the activity.	C

Activity - User capabilities

ID	Requirement	Priority
ACT 2.0	The user can see an overview of past activity events.	M
ACT 2.1	The user can filter the activity history on starting time.	S
ACT 2.2	The user can filter the activity history on ending time.	S
ACT 2.3	The user can filter the activity history on date.	S

ACT 2.4	The user can filter the activity history on type.	S
ACT 2.5	The user can delete an activity.	C
ACT 2.6	The user can edit an activity.	C
ACT 2.7	The user can filter the activity history on duration.	C
ACT 2.8	The user can filter the activity history on name.	C
ACT 2.9	The user can filter the activity history on description.	C
ACT 2.10	The user can modify the burnt calories goal.	C

Activity - Visualizations and Statistics

ID	Requirement	Priority
ACT 3.0	When the user selects a past activity, the application shall visualize glucose levels during this activity.	M
ACT 3.1	The application shall display summary statistics of activity data during a specified time frame, set by the user.	S
ACT 3.2	The application shall visualize the average glucose levels for an activity type during a specified time frame, set by the user.	S
ACT 3.3	The application shall indicate activities on the glucose-time visualizations.	S
ACT 3.4	The application shall indicate each activity type differently on the glucose-time visualizations.	C
ACT 3.5	The application shall visualize the heartbeat rates during a specified time frame, set by the user.	C
ACT 3.6	The application shall visualize the burnt calories during a specified time frame, set by the user.	C
ACT 3.7	The application shall visualize the steps taken during a specified time frame, set by the user.	C

3.1.6 Food

Food - General

ID	Requirement	Priority
FOOD 0.0	The application shall display food entries in a food history section.	M
FOOD 0.1	The application shall display the user's consumed calories for the current date.	M

FOOD 0.2	The application shall display the user's consumed calories goal.	S

Food - User capabilities

ID	Requirement	Priority
FOOD 1.0	The user can upload food data using the application.	M
FOOD 1.1	The user can modify the consumed calories goal.	S
FOOD 1.2	The user can filter the food entries on a specified time frame, set by the user.	S
FOOD 1.3	The user can filter the food entries on the number of carbohydrates.	S
FOOD 1.4	The user can filter the food entries on the number of calories.	S
FOOD 1.5	The user can enter food entries manually in the food history.	C
FOOD 1.6	The user can delete food entries manually from the food history.	C
FOOD 1.7	The user can edit food entries manually in the food history.	C
FOOD 1.8	The user can create a set of favorite foods.	C
FOOD 1.9	The user can enter a food item in the food history by choosing it from the set of favorite foods.	C
FOOD 1.10	The user can upload pictures of meals.	C

Food - Properties

ID	Requirement	Priority
FOOD 2.0	A food item shall have a time.	M
FOOD 2.1	A food item shall have a date.	M
FOOD 2.2	A food item shall have a type.	M
FOOD 2.3	A food item shall have carbohydrates as a property.	M
FOOD 2.4	A food item shall have calories as a property.	M
FOOD 2.5	A food item shall have a name.	S
FOOD 2.6	A food item shall have a glycemic index.	C

Food - Visualization

ID	Requirement	Priority

FOOD 3.0	When the user selects a past food entry, the application shall visualize glucose levels 2 hours before and after the time property of the food entry.	M
FOOD 3.1	The application shall visualize carbohydrates for a specified time frame, set by the user.	M
FOOD 3.2	The application shall display summary statistics of food data during a specified time frame, set by the user.	S

3.1.7 Emotion

Emotion - General

ID	Requirement	Priority
EMOT 0.0	When the user navigates to the main page of the application, the application shall remind the user to enter his emotional status by default every 24 hours.	M
EMOT 0.1	When the user has chosen that he does not want to receive reminders about entering his emotional status, the application shall skip reminding the user to enter his emotional status.	M
EMOT 0.2	When the user wants to enter an emotional status, the application shall allow the user to enter their happiness level.	M
EMOT 0.3	When the user wants to enter an emotional status, the application shall allow the user to enter their excitement level.	M
EMOT 0.4	When the user does not access the application for more than 2 days, the application shall send a GameBus notification to the supervisor.	C

Emotion - User capabilities

ID	Requirement	Priority
EMOT 1.0	The user can enter an emotional status into the Dashboard	M
EMOT 1.1	The user can turn on reminders about his emotional status.	M
EMOT 1.2	The user can turn off reminders about his emotional status.	M
EMOT 1.3	The user can delete an emotional status from the emotional history.	S
EMOT 1.4	The user can edit an emotional status in the emotional history.	S
EMOT 1.5	The user can choose whether to see his emotional status.	C

Emotion - Emotional status properties

ID	Requirement	Priority
EMOT 2.0	An emotional status shall have a time.	M
EMOT 2.1	An emotional status shall have a date.	M
EMOT 2.2	An emotional status shall have a happiness value.	M
EMOT 2.3	An emotional status shall have an excitement value.	M

Emotion - Visualization

ID	Requirement	Priority
EMOT 3.0	The application shall show the last entered emotional status of the user.	M
EMOT 3.1	When the user selects an emotional status, the application shall visualize glucose levels 2 hours before and after the time of the emotional status.	S
EMOT 3.2	The application shall visualize the emotions of the user for a specified time frame, set by the user.	S

3.2 Constraint requirements

Constraint requirements state which limiting requirements the system must/should/can meet.

3.2.1 Privacy & Security

ID	Requirement	Priority
PS 0.0	The system shall verify that the user has the correct access rights before allowing access to the application.	M
PS 0.1	When the user fails to authenticate, the system shall block access to the application.	M
PS 0.2	Only authenticated users shall have access to their user data.	M
PS 0.3	Only a supervisor user can modify data for other users.	M
PS 0.4	The user can log in using his GameBus account.	M
PS 0.5	The application shall comply to GDPR standards for handling personal data [12].	S

3.2.2 Performance

ID	Requirement	Priority
PERF 0.0	When a visualization request for a specified time frame of 1 day is received, the application shall grant it within 5 seconds.	M
PERF 0.1	When an enter request is received, the application shall grant it within 5 seconds.	M
PERF 0.2	When an edit request is received, the application shall grant it within 5 seconds.	M
PERF 0.3	When a filter request is received, the application shall grant it within 5 seconds.	M
PERF 0.4	When a delete request is received, the application shall grant it within 5 seconds.	S
PERF 0.5	When a visualization request is received, the application shall grant it within 3 seconds.	S
PERF 0.6	When an enter request is received, the application shall grant it within 3 seconds.	S
PERF 0.7	When an edit request is received, the application shall grant it within 3 seconds.	S
PERF 0.8	When a delete request is received, the application shall grant it within 3 seconds.	S
PERF 0.9	When a filter request is received, the application shall grant it within 3 seconds.	S
PERF 0.10	When a visualization request is received, the application shall grant it within 1 second.	C
PERF 0.11	When an enter request is received, the application shall grant it within 1 second.	C
PERF 0.12	When an edit request is received, the application shall grant it within 1 second.	C
PERF 0.13	When a delete request is received, the application shall grant it within 1 second.	C
PERF 0.14	When a filter request is received, the application shall grant it within 1 second.	C

3.2.3 Reliability

ID	Requirement	Priority
REL 0.0	The application shall facilitate all use cases without crashing.	M

REL 0.1	The visualized data shall be in correspondence with input data.	S

3.2.4 Environment

ID	Requirement	Priority
ENV 0.0	The application shall support Google Chrome version 90 or higher.	M
ENV 0.1	The application shall be available in English.	M
ENV 0.2	The application shall use functionalities of the GameBus platform.	M
ENV 0.3	The application shall support Google Chrome version 23 or higher.	S
ENV 0.4	The application shall support Mozilla Firefox version 80 or higher.	S
ENV 0.5	The application shall support Mozilla Firefox version 21 or higher.	C
ENV 0.6	The application shall support all Microsoft Edge versions.	C
ENV 0.7	The application shall support Internet Explorer version 10 or higher.	C
ENV 0.8	The application shall support Safari version 6 or higher.	C
ENV 0.9	The application shall support mobile browsers.	C
ENV 0.10	The application shall be available in Dutch.	C

3.2.5 Data

Data - Sending

ID	Requirement	Priority
DATA 0.0	The application shall send glucose data from an Abbott Libre .csv file to GameBus.	M
DATA 0.1	The application shall send insulin data from Abbott Libre .csv file to GameBus.	M
DATA 0.2	The application shall send food data from an Excel spreadsheet to GameBus.	M
DATA 0.3	The application shall acquire food data from an Excel spreadsheet that resides on a specified OneDrive.	S
DATA 0.4	When the application acquires food data from an Excel spreadsheet that resides on a specified OneDrive, the application shall send it to GameBus.	S
DATA 0.5	The application shall send food data coming from the Eetmeter application to GameBus.	S

DATA 0.6	The application shall send user data in the GameBus platform.	S
DATA 0.7	The application shall send food data from the FoodData Central API to GameBus.	C
DATA 0.8	The application shall send glucose data from Dexcom to GameBus.	TBR
DATA 0.9	The application shall track the emotional status of the user through the Wearable Tizen app.	C
DATA 0.10	When the application tracks the emotional status of the user through the Wearable Tizen app, the application shall send it to GameBus.	C
DATA 0.11	The application shall track the emotional status of the user through a mobile phone app.	C
DATA 0.12	When the application tracks the emotional status of the user through a mobile app, the application shall send it to GameBus.	C
DATA 0.13	The application shall send food data from food image snapshots to GameBus.	C
DATA 0.14	The application shall integrate with CalorieMama to track the user's nutrition.	W
DATA 0.15	When the application gets nutrition information from CalorieMama, the application shall send it to GameBus.	W

Data - Retrieving

ID	Requirement	Priority
DATA 1.0	The application shall acquire food data from the GameBus platform.	M
DATA 1.1	The application shall acquire activity data from the GameBus platform.	M
DATA 1.2	The application shall acquire glucose data from the GameBus platform.	M
DATA 1.3	The application shall acquire insulin data from the GameBus platform.	M
DATA 1.4	The application shall acquire emotion data from the GameBus platform.	S
DATA 1.5	The application shall acquire heartbeat data from the GameBus platform.	S
DATA 1.6	When the user is logged in, the application shall continuously acquire heartbeat data with intervals of at most 5 minutes	S
DATA 1.7	The application shall acquire data about burnt calories from the GameBus platform.	S

DATA 1.8	When the user is logged in, the application shall continuously acquire data about burnt calories with intervals of at most 5 minutes	S
DATA 1.9	The application shall acquire data about taken steps from the GameBus platform.	S
DATA 1.10	When the user is logged in, the application shall continuously acquire data about taken steps with intervals of at most 5 minutes	S
DATA 1.11	When the user is logged in, the application shall continuously acquire heartbeat data with intervals of at most 1 minute	C
DATA 1.12	When the user is logged in, the application shall continuously acquire data about burnt calories with intervals of at most 1 minute	C
DATA 1.13	When the user is logged in, the application shall continuously acquire data about taken steps with intervals of at most 1 minute	C
DATA 1.14	When a wearable device is connected, the application shall automatically acquire data about burnt calories	C
DATA 1.15	When a wearable device is connected, the application shall automatically acquire heartbeat data	C
DATA 1.16	When a wearable device is connected, the application shall automatically acquire data about steps taken	C

A Use cases

A.1 User Management

I Specified Time Frame

Goal:	The user modifies his profile information.
Actor:	Normal User/ Supervisor.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The profile information is modified by the actor.
Summary:	The actor user modifies the profile information.
Requirements:	USER 2.1, USER 2.2, USER 5.0, 5.2 to USER 5.4
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor modifies the actor profile information fields about age, height, weight, weight goal	
4.The actor selects to save the information	
	5.The system saves the actor profile information

Alternative

-

A.2 Specified Time Frame

II Set specified time frame

Goal:	The user sets a specified time frame for a visualization
Actor:	Normal User/ Supervisor user
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	A specified time frame for a visualization is set by the actor.
Summary:	The actor sets a specified time frame for a visualization.
Requirements:	USER 2.1, USER 2.2, STF 0.0 to STF 0.3
Priority:	Must have

Steps:

User action	System action
1.The actor clicks on the "Time Frame Panel"	
2.The actor fills the start time	
3.The actor fills the end time	
4.The actor fills in the start date	
5.The actor fills in the end date	
6.The actor selects optionally glucose type	
7.The actor selects optionally activities	
8.The actor selects optionally emotion type	
9.The actor selects optionally food type	
10. The actor requests the visualization	
	11. The system fetches data from the specified time frame

Alternative

11B: In case there is no data in the database for the selected time frame, or the database cannot be reached, an error message is displayed.

A.3 Health Metrics

III Visualize Glucose Data

Goal:	The user visualizes his glucose data.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The normal user has measured his glucose data and has sent it to the Abbott Libre application at least once. The actor chooses to visualize the glucose data. The user specifies a time frame.
Postcondition:	A visualization of the blood glucose data over a specified time frame is generated, with the amount/ percentage of time the blood glucose values were "Very High"/ "High"/ "Normal"/ "Low"/ "Very Low".
Summary:	The actor selects to visualize blood glucose data and a time frame, and a visualization of the blood glucose data over the specified time frame is presented.
Requirements:	USER 2.1, USER 5.1, STF 0.0 to STF 0.3, HM 1.0, HM 2.0 to HM 2.10
Priority:	Must have

Steps:

User action	System action
1.The actor selects the “Visualization Options Menu” 2.The actor chooses the "Blood Glucose" related options 3.The actor fills the “Time Frame Panel” 4.The actor requests the visualization	5. The system displays a visualization of the blood glucose data in the chosen period

Alternative

5B: In case there are no glucose data in the selected time frame, or the database cannot be reached, nothing is displayed.

IV Visualize hypoglycemia occurrences

Goal:	The user visualizes his hypoglycemia occurrences for a specified time frame.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The actor chooses to visualize the hypoglycemia occurrences. The actor specifies a time frame. There have been hypoglycemia occurrences in the specified time frame.
Postcondition:	A visualization of the hypoglycemia occurrences over a specified time frame is generated.
Summary:	The actor selects to visualize hypoglycemia occurrences and a time frame, and a visualization of the hypoglycemia occurrences over the specified time frame is presented.
Requirements:	USER 2.1, USER 5.1, STF 0.0 to STF 0.3, HM 2.13
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “Visualization Options Menu”	
	2.The system shows the actor the possible visualization options
3.The actor chooses the "Hypoglycemia Occurrences" option	
4.The actor fills the “Time Frame Panel”	
5.The actor requests the visualization	
	6. The system displays a visualization of the hypoglycemia occurrences in the chosen period

Alternative

6B: In case there are no hypoglycemia occurrences in the selected time frame, or the database cannot be reached, nothing is displayed.

V Visualize hyperglycemia occurrences

Goal:	The user visualizes his hyperglycemia occurrences for a specified time frame.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The actor chooses to visualize the hyperglycemia occurrences. The actor specifies a time frame. There have been hyperglycemia occurrences in the specified time frame.
Postcondition:	A visualization of the hyperglycemia occurrences over a specified time frame is generated.
Summary:	The actor selects to visualize hyperglycemia occurrences and a time frame, and a visualization of the hyperglycemia occurrences over the specified time frame is presented.
Requirements:	USER 2.1, USER 5.1, STF 0.0 to STF 0.3, HM 2.14
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “Visualization Options Menu”	
	2.The system shows the actor the possible visualization options
3.The actor chooses the "Hyperglycemia Occurrences" option	
4.The actor fills the “Time Frame Panel”	
5.The actor requests the visualization	
	6. The system displays a visualization of the hyperglycemia occurrences in the chosen period

Alternative

6B: In case there are no hyperglycemia occurrences in the selected time frame, or the database cannot be reached, nothing is displayed.

VI Visualize insulin intake for a specified time frame

Goal:	The user visualizes the insulin intake for a specified time frame.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The actor chooses to visualize the insulin intake.
Postcondition:	A visualization of the insulin intake for a specified time frame is generated, where there is a distinction between the rapid-acting and the slow-acting insulin intake of the normal user.
Summary:	The actor selects to visualize insulin intake and a visualization of the insulin intake for a specified time frame is presented, where there is a distinction between the rapid-acting and the slow-acting insulin intake.
Requirements:	USER 2.1, USER 5.1, STF 0.0 to STF 0.3, HM 2.21 to HM 2.22
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “Visualization Options Menu”	
	2.The system shows the actor the possible visualization options
3.The actor chooses the "Insulin Intake" option	
4.The actor fills the "Time Frame Panel"	
5.The actor requests the visualization	
	6.The system displays a visualization of the insulin intake in the chosen period

Alternative

6B: In case there is no insulin information or the database cannot be reached, nothing is displayed.

VII Upload glucose data

Goal:	The user uploads glucose data using the application.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	Glucose data is uploaded using the application.
Summary:	The actor uploads glucose data using the application.
Requirements:	USER 2.2, HM 4.0
Priority:	Must have

Steps:

User action	System action
1.The actor navigate to the "Health Metrics Section"	
	2.The system displays the "Health Metrics Section"
3.The actor selects to upload glucose data	
4.The actor chooses the glucose data he wants to upload	
5.The actor confirms the upload	
	6.The system uploads the glucose data

Alternative

5B: The actor cancels the upload and no glucose data is uploaded.

VIII Modify the ranges of the glucose level types

Goal:	The user changes the ranges of the glucose level types in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The ranges of the glucose level types are changed in the dashboard.
Summary:	The actor changes the ranges of the glucose level types in the dashboard with new values.
Requirements:	USER 2.2, HM 4.1 to HM 4.5
Priority:	Must have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the field for the range of a glucose level type	
	4.The system allows the actor to modify the field for the range of the selected glucose level type
5.The actor enters the new values in the field for the range of the selected glucose level type	
6. The actor clicks outside of the field for the range of the selected glucose level type	
	7.The system saves the new range of the selected glucose level type

Alternative

-

IX Modify the classification threshold of the hyperglycemia occurrences

Goal:	The user changes the classification threshold of the hyperglycemia occurrences in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The classification threshold of the hyperglycemia occurrences is changed in the dashboard.
Summary:	The actor changes the classification threshold of the hyperglycemia occurrences in the dashboard with a new value.
Requirements:	USER 2.2, HM 4.6
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the field for the classification threshold of the hyperglycemia occurrences	
	4.The system allows the actor to modify the field for the classification threshold of the hyperglycemia occurrences
5.The actor enters the new values in the field for the classification threshold of the hyperglycemia occurrences	
6. The actor clicks outside of the field for the threshold of the classification threshold of the hyperglycemia occurrences	
	7.The system saves the new threshold of the classification threshold of the hyperglycemia occurrences

Alternative

-

X Modify the classification threshold of the hypoglycemia occurrences

Goal:	The user changes the classification threshold of the hypoglycemia occurrences in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The classification threshold of the hypoglycemia occurrences is changed in the dashboard.
Summary:	The actor changes the classification threshold of the hypoglycemia occurrences in the dashboard with a new value.
Requirements:	USER 2.2, HM 4.7
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the field for the classification threshold of the hypoglycemia occurrences	
	4.The system allows the actor to modify the field for the classification threshold of the hypoglycemia occurrences
5.The actor enters the new values in the field for the classification threshold of the hypoglycemia occurrences	
6. The actor clicks outside of the field for the threshold of the classification threshold of the hypoglycemia occurrences	
	7.The system saves the new threshold of the classification threshold of the hypoglycemia occurrences

Alternative

-

XI Select to view information about a hyperglycemia/ hypoglycemia occurrence from the hyperglycemia/ hypoglycemia occurrences visualization

Goal:	The user selects to view information about a hyperglycemia/ hypoglycemia occurrence from the hyperglycemia/ hypoglycemia occurrences visualization.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is a visualization of the hyperglycemia/ hypoglycemia occurrences available to the actor.
Postcondition:	A hyperglycemia/ hypoglycemia occurrence from the hyperglycemia/ hypoglycemia occurrences visualization is selected by the actor and the actor can see information related to it.
Summary:	The actor selects and views information about a hyperglycemia/ hypoglycemia occurrence from the hyperglycemia/ hypoglycemia occurrences visualization.
Requirements:	USER 2.1, USER 5.1, HM 4.8, HM 4.9, HM 2.15 to HM 2.20
Priority:	Should have

Steps:

User action	System action
1.The actor selects a hyperglycemia/ hypoglycemia occurrence from the hyperglycemia/ hypoglycemia occurrences visualization	2.The system shows the actor information about the occurrence

Alternative

-

XII Modify the A1C estimate goal

Goal:	The user changes the A1C estimate goal in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The A1C estimate goal value is changed in the dashboard.
Summary:	The actor changes the current value for the A1C estimate goal in the dashboard with a new value.
Requirements:	USER 2.2, USER 5.0, HM 4.10
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the field for the A1C estimate goal	
	4.The system allows the actor to modify the A1C estimate goal field
5.The actor enters the new value in the A1C estimate goal field	
6.The actor clicks outside of the blood A1C estimate field	
	7.The system saves the new A1C estimate goal value and displays it

Alternative

-

XIII Enter insulin intake

Goal:	The user enters insulin intake in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The amount, type, date, and time of the insulin intake are logged in the application.
Summary:	The actor enters the amount, type, date, and time of the insulin intake in the application, and this information is saved.
Requirements:	USER 2.2, USER 5.0, HM 4.11 to HM 4.15
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor selects the enter insulin intake option	
	4.The system displays the “Enter Insulin Intake Section”
5.The actor sets the amount of the insulin intake	
6.The actor sets the type of the insulin intake	
7.The actor sets the time of the insulin intake	
8.The actor sets the date of the insulin intake	
9.The actor selects to save the insulin intake	
	10.The system saves the insulin intake

Alternative

-

XIV Edit insulin intake

Goal:	The user edits insulin intake in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The amount, type, date, and time of the insulin intake are edited in the application.
Summary:	The actor edits the amount, type, date, and time of the insulin intake in the application and this information is saved.
Requirements:	USER 2.2, USER 5.0, HM 4.16
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor selects to view the insulin intake inputs	
	4.The system displays the “Insulin Intake Section”
5.The actor selects to edit an insulin intake	
6.The actor sets the new amount of the insulin intake	
7.The actor sets the new type of the insulin intake	
8.The actor sets the new time of the insulin intake	
9.The actor sets the new date of the insulin intake	
10.The actor selects to save the insulin intake	
	11.The system updates the information of the insulin intake

Alternative

-

XV Delete insulin intake

Goal:	The user deletes insulin intake in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	An insulin intake input is deleted from the application.
Summary:	The actor deletes an insulin intake from the application.
Requirements:	USER 2.2, USER 5.0, HM 4.17
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor selects to view the insulin intake inputs	
	4.The system displays the “Insulin Intake Section”
5.The actor selects to delete an insulin intake	
	6.The system asks the actor for confirmation
7.The actor confirms	
	8.The system deletes the insulin intake

Alternative

7B: The actor cancels the deletion and the insulin intake is not deleted.

XVI Change the glucose level units

Goal:	The user changes the glucose level units in the dashboard
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The glucose level units are changed in the dashboard.
Summary:	The actor changes the glucose level units and the new units are displayed in the dashboard.
Requirements:	USER 2.2, USER 5.0, HM 4.20
Priority:	Could have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the “Glucose Level Units Panel”	
	4.The system shows the actor the possible glucose level unit options
5.The actor selects an option	
	6.The system displays the new glucose level unit

Alternative

-

A.4 Activity

XVII See overview of past activities

Goal:	The user sees overview of past activity events
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is activity information available in the application.
Postcondition:	An overview of past activity events is shown on the dashboard.
Summary:	The actor sees an overview of past activities on the dashboard.
Requirements:	USER 2.1, USER 5.1, ACT 2.0
Priority:	Must have

Steps:

User action	System action
1.The actor opens the main page of the dashboard	
	2.The system displays an overview of activities on the dashboard

Alternative

-

XVIII Filter the activity history on date, starting time, ending time, and type

Goal:	The user filters the activity history on date, starting time, ending time, and type.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is activity information available in the application.
Postcondition:	The activity history is filtered on the date, starting time, ending time, and type.
Summary:	The actor filters the activity history on the date, starting time, ending time, and type.
Requirements:	ACT 2.0 to ACT2.4
Priority:	Should have

Steps:

User action	System action
1.The actor navigates to the “Activity Section” 2.The actor enters information about the date, starting time, ending time, and type in the filter menu	3.The system displays the activities according to the filter information

Alternative

-

XIX Delete activity

Goal:	The user deletes an activity
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is activity information available in the application.
Postcondition:	An activity event is deleted from the dashboard.
Summary:	The actor deletes an activity from the dashboard.
Requirements:	USER 2.2, USER 5.0, ACT 2.5
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Activity Section”	
2.The actor selects the delete option on the activity	
	3.The system asks the actor for confirmation
4.The actor confirms	
	5.The system deletes the activity from the activity history

Alternative

4B: The actor does not confirm and the activity is not deleted.

XX Edit activity

Goal:	The user edits an activity
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is activity information available in the application.
Postcondition:	An activity event is edited
Summary:	The actor edits an activity
Requirements:	USER 2.2, ACT 2.6, USER 5.0
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Activity Section”	
2.The actor selects the edit option on the activity	
	3.The system displays an overview of the activity information that can be changed
4.The actor edits the fields in the overview of the activity information that can be changed	
5.The actor saves the changes	
	6.The system saves the new information of the activity in the activity history

Alternative

5B: The actor cancels the editing process and the activity is not edited.

XXI Filter the activity history on duration, name, and description

Goal:	The user filters the activity history on duration, name, and description
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is activity information available in the application.
Postcondition:	The activity history is filtered on duration, name, and description
Summary:	The actor filters the activity history on duration, name, and description
Requirements:	ACT 2.0, ACT 2.7 to ACT 2.9
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Activity Section” 2.The actor enters information about duration, name, and description in the filter menu	3.The system displays the activities according to the filter information

Alternative

-

XXII Modify the burnt calories goal

Goal:	The user changes the burnt calories goal in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The burnt calories goal value is changed in the dashboard.
Summary:	The actor changes the current value for the burnt calories goal in the dashboard with a new value.
Requirements:	USER 2.2, ACT 2.10, USER 5.0
Priority:	Could have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the field for the burnt calories goal	
	4.The system allows the actor to modify the burnt calories goal field
5.The actor enters the new value in the burnt calories goal field	
6.The actor clicks outside of the burnt calories field	
	7.The system saves the new burnt calories goal value and displays it

Alternative

-

XXIII Visualize glucose levels during an activity.

Goal:	The user sees a visualization of the glucose levels during an activity.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The normal user has measured his glucose data and has sent it to the Abbott Libre application. The normal user has tracked his activity with an application supported by the GameBus platform. There is glucose data in the time frame of the activity.
Postcondition:	A visualization of the glucose levels during an activity is displayed
Summary:	The actor sees a visualization of the glucose levels during an activity.
Requirements:	USER 2.1, USER 5.1, ACT 3.0
Priority:	Must have

Steps:

User action	System action
1.The actor navigates to the “Activity Section” 2.The actor selects an activity from the activity history	
	3.The system visualizes the glucose data in the time frame of the activity in the "Visualization Section"

Alternative

3B: In case no glucose data is available in the time frame, or the database cannot be reached, nothing is displayed.

XXIV View summary statistics of activity data during a specified time frame

Goal:	The user sees summary statistics of activity data during a specified time frame.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The normal user has tracked his activity with an application supported by the GameBus platform.
Postcondition:	Summary statistics of activity data during a specified time frame are displayed.
Summary:	The actor sees summary statistics of activity data during a specified time frame.
Requirements:	USER 2.1, USER 5.1, ACT 3.1
Priority:	Should have

Steps:

User action	System action
1.The actor navigates to the “Visualization Section”	
2.The actor fills the “Time Frame Panel”	
3.The actor selects the "Activity Option Menu"	
	4.The system shows the actor the possible activities the actor can choose between
5.The actor selects an activity type	
6.The actor requests to see summary statistics	
	7.The system displays summary statistics of activity data during the specified time frame

Alternative

7B: In case no activity data is available in the time frame, or the database cannot be reached, nothing is displayed.

XXV Visualize the average glucose levels for an activity type during a specified time frame

Goal:	The user sees a visualization of the average glucose levels for an activity type during a specified time frame.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The normal user has measured his glucose data and has sent it to the Abbott Libre application. The normal user has tracked his activity with an application supported by the GameBus platform.
Postcondition:	Visualization of the average glucose levels for an activity type during a specified time frame is shown.
Summary:	The actor sees a visualization of the average glucose levels for an activity type during a specified time frame.
Requirements:	USER 2.1, USER 5.1, ACT 3.2
Priority:	Should have

Steps:

User action	System action
1.The actor navigates to the “Visualization Section”	
2.The actor fills the “Time Frame Panel”	
3.The actor selects the "Activity Option Menu"	
	4.The system shows the actor the possible activities the actor can choose between
5.The actor selects an activity type	
6.The actor requests to see the average glucose levels	
	7.The system displays a visualization of the average glucose levels for an activity type during a specified time frame

Alternative

7B: In case no glucose data or activity data is available in the time frame, or the database cannot be reached, nothing appears.

A.5 Food

XXVI Upload food data

Goal:	The user uploads food data using the application.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	Food data is uploaded using the application.
Summary:	The actor uploads food data using the application.
Requirements:	USER 2.2, FOOD 1.0
Priority:	Must have

Steps:

User action	System action
1.The actor navigate to the "Food Section"	
	2.The system displays the "Food Section"
3.The actor selects to upload food data	
4.The actor chooses the food data he wants to upload	
5.The actor confirms the upload	
	6.The system uploads the food data

Alternative

5B: The actor cancels the upload and no food data is uploaded.

XXVII Modify the consumed calories goal

Goal:	The user changes the consumed calories goal in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The consumed calories goal value is changed in the dashboard.
Summary:	The actor changes the current value for the consumed calories goal in the dashboard with a new value.
Requirements:	FOOD 1.1, USER 2.2, USER 5.0
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor clicks on the field for the consumed calories goal	
	4.The system allows the actor to modify the consumed calories goal field
5.The actor enters the new value in the consumed calories goal field	
6.The actor clicks outside of the consumed calories field	
	7.The system saves the new consumed calories goal value and displays it

Alternative

-

XXVIII Filter the food entries on a specified time frame, amount of carbohydrates, and amount of calories

Goal:	The user filters the food entries on specified time frame, amount of carbohydrates, and amount of calories
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is food history available to the application.
Postcondition:	The food history is filtered on a specified time frame, amount of carbohydrates, and amount of calories
Summary:	The actor filters the food history on a specified time frame, amount of carbohydrates, and amount of calories
Requirements:	FOOD 1.2, FOOD 1.3, FOOD 1.4, STF 0.0 to STF 0.3
Priority:	Should have

Steps:

User action	System action
1.The actor navigates to the “Food Section” 2.The actor enters information about the specified time frame, amount of carbohydrates, and amount of calories in the filter menu	3.The system displays the food entries according to the filter information

Alternative

-

XXIX Enter food into the food history

Goal:	The user enters food in the food history.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	A food entry is added to the food history.
Summary:	The actor enters a food entry in the food history.
Requirements:	FOOD 1.5, USER 2.2, USER 5.0
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Food Section”	
2.The actor selects the option to add food	
	3.The system displays an overview of the food item information that can be entered
4.The actor edits the fields in the overview of the food item information	
5.The actor saves the changes	
	6.The system saves the food item with the provided information in the food history

Alternative

-

XXX Delete food entry from the food history

Goal:	The user deletes a food entry from the food history
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	A food entry is deleted from the food history.
Summary:	The actor deletes a food entry from the food history
Requirements:	FOOD 1.6, USER 2.2, USER 5.0
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Food Section”	
2.The actor selects the delete option on a selected food entry	
	3.The system asks the actor for confirmation
4.The actor confirms	
	5.The system deletes the food entry from the food history

Alternative

4B: The actor does not confirm and the food entry is not deleted.

XXXI Edit food entry in the food history

Goal:	The user edits food entries in the food history.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	A food entry is added to the food history.
Summary:	The actor edits a food entry in the food history.
Requirements:	FOOD 1.7, USER 2.2, USER 5.0
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Food Section”	
2.The actor selects the option to edit a food entry	
	3.The system displays an overview of the food item information that can be edited
4.The actor edits the fields in the overview of the food item information that can be edited	
5.The actor saves the changes	
	6.The system saves the new information of the food item in the food history

Alternative

-

XXXII Create a set of favourite foods

Goal:	The user creates a set of favourite foods in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The actor has created a set of favorite foods in the dashboard.
Summary:	The normal user has created a set of favorite foods in the dashboard.
Requirements:	FOOD 1.8
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the “Food Section”	
2.The actor selects the option to add a meal in the "Favourite Foods Section"	
	3.The system displays a field where the actor can enter a favorite food.
4.The actor adds a favorite food.	
	5.The system saves the favorite food in the "Favourite Foods Section"

Alternative

-

XXXIII Upload picture of meal

Goal:	The user uploads picture of a meal in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The actor has uploaded a picture of a meal in the dashboard.
Summary:	A picture of the normal users' meal is uploaded in the dashboard.
Requirements:	FOOD 1.10
Priority:	Could have

Steps:

User action	System action
1.The actor navigates to the "Food Section"	
2.The actor selects the option to add a meal picture	
	3.The system displays a field where the actor can upload a picture.
4.The actor uploads a picture of his meal	
	5.The system saves the picture of the meal in the food history

Alternative

-

XXXIV Visualize glucose levels 2 hours before and after the time of a selected food entry

Goal:	The user visualizes the glucose levels 2 hours before and after the time of a selected food entry
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is food history available to the application. There is glucose data available to the application for the time frame of the food entry.
Postcondition:	The glucose levels 2 hours before and after the time of a selected food entry are visualized to the actor.
Summary:	The actor visualizes the glucose levels 2 hours before and after the time of a selected food entry
Requirements:	FOOD 3.0, USER 5.1, USER 2.1
Priority:	Must have

Steps:

User action	System action
1.The actor navigates to the “Food Section” 2.The actor selects a food entry from the “Food Section”	 3.The system visualizes the glucose data 2 hours before and after the time of a selected food entry"

Alternative

3B: In case no glucose data is available in the time frame, or the database cannot be reached, an error message appears.

XXXV Visualize carbohydrates for a specified time frame

Goal:	The user visualizes carbohydrates for a specified time frame.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is food information available in the application. The actor specifies a time frame.
Postcondition:	A visualization of the carbohydrates for a specified time frame is generated.
Summary:	The actor selects to visualizes carbohydrates for a specified time frame.
Requirements:	FOOD 3.1, USER 2.1, USER 5.1, STF 0.0 to STF 0.3
Priority:	Must have

Steps:

User action	System action
1.The actor selects the “Visualization Options Menu”	
	2.The system shows the actor the possible visualization options
3.The actor chooses the "Food" related options	
4.The actor fills the “Time Frame Panel”	
5.The user requests the visualization	
	6. The system displays a visualization of the carbohydrates in the chosen period

Alternative

6B: In case there is no carbohydrate information in the database for the selected time frame, or the database cannot be reached, nothing is displayed.

XXXVI View summary statistics of food data during a specified time frame

Goal:	The user sees summary statistics of food data during a specified time frame.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The normal user has tracked his food intake.
Postcondition:	Summary statistics of food data during a specified time frame are displayed.
Summary:	The actor sees summary statistics of food data during a specified time frame.
Requirements:	USER 2.1, USER 5.1, FOOD 3.2
Priority:	Should have

Steps:

User action	System action
1.The actor navigates to the “Visualization Section”	
2.The actor fills the “Time Frame Panel”	
3.The actor selects the "Food Option Menu"	
	4.The system shows the actor the possible food types the actor can choose between
5.The actor selects an food type	
6.The actor requests to see summary statistics	
	7.The system displays summary statistics of food data during the specified time frame

Alternative

7B: In case no food data is available in the time frame, or the database cannot be reached, nothing is displayed.

A.6 Emotion

XXXVII Enter emotional status

Goal:	The user enters emotional status in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The actor has chosen his emotional status out of the list with 12 emotional status types.
Summary:	The actor enters his emotional status in the application and this information is saved.
Requirements:	EMOT 1.0, USER 2.2, USER 5.0
Priority:	Must have

Steps:

User action	System action
1.The actor selects the “Emotional Status Section”	
	2.The system displays the “Emotional Status Section”
3.The actor selects the "Enter Emotional Status" option	
	4.The system displays the list of 12 emotional status types
5.The actor chooses 1 type out of the 12 emotional status types in the list	
6.The actor sets the time of the emotional status	
7.The actor sets the date of the emotional status	
8.The actor selects to save the emotional status	
	9.The system saves the emotional status

Alternative

-

XXXVIII Enable/disable emotion reminders

Goal:	The user enables/ disables emotion reminders.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The actor enables/ disables emotion reminders.
Summary:	The actor enables/ disables emotion reminders.
Requirements:	USER 5.0, EMOT 1.1 to 1.2
Priority:	Must have

Steps:

User action	System action
1.The actor selects the “User Profile Section”	
	2.The system displays the “User Profile Section”
3.The actor selects the "Enable/Disable Emotion" option	
	4.The system shall enables/ disables emotion reminders

Alternative

-

XXXIX Delete emotional status

Goal:	The user deletes emotional status in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	An emotional status input is deleted from the application.
Summary:	The actor deletes an emotional status from the application.
Requirements:	EMOT 1.1, USER 2.2, USER 5.0
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “Emotional Status Section”	
	2.The system displays the “Emotional Status Section”
3.The actor selects to view the emotional status inputs	
	4.The system displays the “Emotional Status Inputs Section”
5.The actor selects to delete an emotional status input	
	6.The system asks the actor for confirmation
7.The actor confirms	
	8.The system deletes the emotional status input

Alternative

7B: The actor cancels the deletion and the emotional status input is not deleted.

XL Edit emotional status

Goal:	The user edits emotional status input in the dashboard.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application.
Postcondition:	The emotional status input is edited in the application.
Summary:	The actor edits the emotional status input in the application and this information is saved.
Requirements:	EMOT 1.2, USER 2.2, USER 5.0
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “Emotional Status Section”	
	2.The system displays the “Emotional Status Section”
3.The actor selects to view the emotional status inputs	
	4.The system displays the “Emotional Status Inputs Section”
5.The actor selects to edit an emotional status inputs	
6.The actor chooses a new emotional status type out of the list with 12 emotional status types	
7.The actor sets the new time of the emotional status	
8.The actor sets the new date of the emotional status	
9.The actor selects to save the emotional status input	
	10.The system updates the information of the emotional status input

Alternative

-

XLI Visualize glucose levels 2 hours before and after the time of a selected emotional status

Goal:	The user visualizes the glucose levels 2 hours before and after the time of a selected emotional status.
Actor:	User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. There is food history available to the application. There is glucose data available to the application for the time frame of the emotional status.
Postcondition:	The glucose levels 2 hours before and after the time of a selected emotional status are visualized to the actor.
Summary:	The actor visualizes the glucose levels 2 hours before and after the time of a selected emotional status
Requirements:	EMOT 3.1, USER 5.1, USER 2.1
Priority:	Should have

Steps:

User action	System action
1.The actor navigates to the “Emotional Status Section” 2.The actor selects an emotional status entry from the “Emotional Status Section”	3.The system visualizes the glucose data 2 hours before and after the time of a selected emotional status entry"

Alternative

3B: In case no glucose data is available in the time frame, or the database cannot be reached, nothing is displayed.

XLII Visualize Emotion Data

Goal:	The user visualizes his emotion data.
Actor:	Normal User/ Supervisor user.
Precondition:	The actor has an active account and is logged in to the application. The normal user has entered emotional data. The user specifies a time frame.
Postcondition:	A visualization of the emotion data over a specified time frame is generated.
Summary:	The actor selects to visualize emotion data and a time frame, and a visualization of the emotion data over the specified time frame is presented.
Requirements:	USER 2.1, STF 0.0 to STF 0.3, EMOT 3.2
Priority:	Should have

Steps:

User action	System action
1.The actor selects the “Visualization Options Menu” 2.The actor chooses the "Emotion" related option 3.The actor fills the “Time Frame Panel” 4.The actor requests the visualization	5. The system displays a visualization of the emotion data in the chosen period

Alternative

5B: In case there are no emotion data in the selected time frame, or the database cannot be reached, nothing is displayed.

B Client and Supervisor Approval

B.1 Client approval

Name: Michel Chaudron

Date: 09-06-2021

Approval:

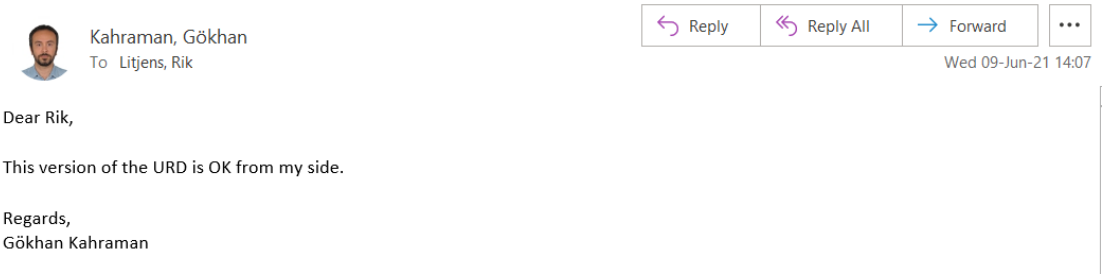
Name: Pieter van Gorp

Date: 09-06-2021

Approval:

B.2 Supervisor approval

Name: Gökhan Kahraman

Date: 09-06-2021

Approval: