

DIABETEC.

Entwicklungsprojekte interaktiver Systeme

im Wintersemester 2018/2019

bei Prof. Dr. Gerhard Hartmann und Prof. Dr. Kristian
Fischer

von Sami Hassini

betreut von Robert Gabriel

Inhaltsverzeichnis

1. Installationsdokument	3
2. Prozessassessment	4
3. Fazit.....	5

1. Installationsdokument

Das Installationsdokument dient zur Erläuterung der Einweisung und der Anforderungen des Systems. Sämtliche Implementationen und benötigten Softwaredateien befinden sich im GitHub Repository unter der URL:

https://github.com/sami1905/EISWS1819Hassini/tree/master/Abgabe_MS3/Implementation/Diabetec

Um das gesamte GitHub-Repository zu erhalten, muss dieses zunächst geklont werden. Der Befehl dafür lautet:

```
❯ git clone https://github.com/sami1905/EISWS1819Hassini.git
```

Folglich wird der Installationsvorgang des Servers und Clients beschrieben.

1.1 Server

Anforderungen

Für den Server ist die Installation von Node JS im File-Ordner notwendig.

Installation

1. Starte Shell und öffne den File-Ordner „Implementation“ unter dem Pfad *EISWS1819Hassini/Abgabe_MS3/Implementation/Diabetec*.
2. Wechsle in den File-Ordner „Server“.
3. Führe den Befehl *npm install all* aus um alles notwendige zu installieren.
4. Starte den Server mit *node app.js*.
5. Der Server läuft, sobald die Nachricht „Server läuft auf Port 3000.“ erscheint.

1.2 Client

Anforderungen

Für den Client wird die aktuelle Version von Android Studio benötigt.

Installation

1. Android Studio starten und öffne dort den Pfad *EISWS1819Hassini/Abgabe_MS3/Implementation*.
2. Ein Android Virtual Device installieren und starten.
3. Projekt starten.

2. Prozessassessment

In der ersten Phase des Entwicklungsprozesses galt es ein Konzept mit entsprechender Domänenrecherche, Zielsetzung und Stakeholder-Analyse anzufertigen. In dieser Phase konnte sich gut mit der Domäne beschäftigt und diese erläutert werden, da bereits eigene Erfahrungen im Umgang mit Diabetes mellitus gemacht wurden. Es wurde geschafft, alle wichtigen und relevanten Aspekte des Diabetes in einem großen Umfang zu erläutern. Mit der Konzipierung des zu entwickelnden Systems ist man alles im allem gut zu recht gekommen.

In der zweiten Phase der Entwicklung wurden die Mensch-Computer-Interaktion und die Web-basierte-Anwendung berücksichtigt. Diese Phase wies die größten Probleme auf. In der Prozessmodellierung fehlte es meist an Input, Beispielen oder Hilfestellungen. Hatte man keinen Zugang zu der benötigten Literatur, war man leicht aufgeschmissen. Dies verursachte eine große Ungewissheit bezüglich des Umfangs und der Richtigkeit der Modellierung.

Auch die Systemmodellierung fiel dem Entwicklungsteam schwer. Hier gelang es erst sehr spät, die erlangten Kenntnisse aus dem Modul „Web-basierte-Anwendungen II“ mit den Erwartungen aus dem Modul „Entwicklung interaktiver Systeme“ zu verknüpfen. Besonders die Systemarchitektur schien nicht klar zu sein. Hier wurde sich bis zuletzt zu sehr an die erlernten Grundlagen der „Web-basierte Anwendungen“ gehalten. Sehr selten wurde „über den Tellerrand“ geschaut.

In der dritten und letzten Phase wurde ein funktionaler Prototyp anhand des Konzeptes und der Modellierung implementiert. In der Android-Programmierung gab es im Entwicklerteam keine Erfahrungen, wodurch viel Zeit in das Recherchieren und Erlernen von Android-Programmierung investiert wurde. Dies hat sich auf die kurze Projektzeit ausgewirkt. Zudem wurde sich zu Beginn der Implementierung sehr an der Client-Oberfläche festgehalten. Einige Funktionalitäten wurden implementiert und durch die Erfahrung mit der Diabetes und von Funktionen anderer Applikationen, konnte in jeder Implementierungsphase kritisch Stellung genommen und iterative Verbesserungen vollzogen werden. Auch in dieser Phase gab es zeitliche Einschränkung. Nicht alles Geplante, konnte umgesetzt werden, da der Zeitrahmen des Entwicklungsprozesses zu klein war.

3. Fazit

Im Fazit wird auf die Zielerreichung anhand der festgelegten Zielhierarchie eingegangen.

Die operativen Ziele wurden alle bis auf eines erreicht.

Der Benutzer erhält Zugang zu einem System, welches kontinuierliche Blutzuckermessung ermöglicht. Dies wurde mit der Schnittstelle zur Dexcom-API, welche im Prototyp implementiert wurde, umgesetzt. Der implementierte Prototyp erhält bis zu 288 Blutzuckerwerte am Tag. Dies entspricht allen fünf Minuten einen Blutzuckerwert. Hiermit wurde das Ziel erreicht und eine kontinuierliche Blutzuckermessung implementiert.

Dem Benutzer wurde es zudem ermöglicht, manuelle Erfassung der Blutzuckerwerte durch Ereignis-Einträge in ein Tagebuch durchzuführen.

Der funktionale Prototyp besitzt sowohl einen Insulineinheiten-, Korrektoreinheiten- und BE-Rechner, als auch einen HbA1c-Rechner.

Lediglich das Benachrichtigen des Benutzers beim Erfassen eines schlechten Blutzuckerwertes konnte nicht umgesetzt werden. Dies liegt daran, dass die Blutzuckerwerte der Dexcom-API nicht in Echtzeit sondern mit vier bis sechs Stunden Verzögerung an den Prototypen übermittelt werden. Somit ist eine Benachrichtigung bei schlechten Blutzuckerwerten aktuell nicht umsetzbar.

Auch die taktischen Ziele wurden meist erreicht. Durch das Einsehen des berechneten HbA1c-Wertes erlangt der Benutzer die Möglichkeit, seinen HbA1c-Wert zu kontrollieren und in einem

gesunden Bereich zu halten. Zu dem wurde durch die simple Erstellung von Ereignissen mit BE- Insulin- und Korrektur-Rechner eine einfache und zeitgewinnende Dokumentation ermöglicht, das dem Benutzer sämtliche Berechnungen genommen werden. Hier benötigt der Benutzer kein analoges Tagebuch und auch kein Stift zum festhalten seiner Diabetes-Ereignisse.

Die Individualität des Benutzers wurde berücksichtigt. Durch Benutzerdaten wurden individuelle Insulin- und BE-Rechner angewendet, welche einen benutzerspezifischen Insulinbedarf berechnen.

Von den festgelegten Alleinstellungsmerkmalen wurden zwei von vier umgesetzt.

Mit dem Insulin-, Korrektur, Be- und HbA1c-Rechner hat der funktionale Prototyp zwei alleinstellungsmerkmale. Die Erkennung von Mustern wurde begonnen zu implementieren, jedoch zeitlich nicht vervollständigt.

Lediglich mit der Druckvorlage aus dem gesammelten Daten wurde sich nicht beschäftigt. Die zwei nicht erreichten Alleinstellungsmerkmale sollten in zukünftiger Implementierung umsetzbar sein.