

Diplomarbeit

Diabetes Helfer für Kinder



Projektteam: Tamara Holzer

Petra Enengl

Patrick Mittermayr

Projektbetreuer: Prof. Dipl.-Ing. Dr. Michael Buchberger

Bearbeitungszeitraum: 18.08.2015 – 08.04.2016

Eidesstattliche Erklärung

Hiermit versichern wir, die vorliegende Arbeit selbstständig, ohne fremde Hilfe und ohne Benutzung anderer als der von uns angegebenen Quellen angefertigt zu haben. Alle Stellen, die wörtlich oder sinngemäß aus fremden Quellen direkt oder indirekt übernommen wurden, sind als solche gekennzeichnet.

Perg, _____ Unterschrift _____

(Tamara Holzer)

Perg, _____ Unterschrift _____

(Petra Enengl)

Perg, _____ Unterschrift _____

(Patrick Mittermayr)

Danksagung

In erster Linie bedanken wir uns bei unserem Betreuungslehrer Dipl.-Ing. Dr. Michael Buchberger, welcher uns mit seiner Fachkompetenz sowie Ratschlägen bei der gesamten Durchführung zur Verfügung stand.

Weiters bedanken wir uns bei Karoline Holzer, welche uns bei dieser Diplomarbeit als Beraterin für die Thematik *Diabetes* zur Verfügung stand.

Ein großer Dank gilt auch unserer Auftraggeberin, die uns mit Listen von Lebensmittel und deren Kohlenhydrate versorgte.

Im weiteren Sinne bedanken wir uns ganz herzlich bei allen, welche uns bei der Durchführung der Diplomarbeit unterstützt haben, vor allem Familien, Freunden und Klassenkollegen.

Herzlichen Dank!

Inhaltsverzeichnis

1.	Einführung	6
1.1.	Kurzfassung.....	6
1.2.	Abstract	7
2.	Projektdefinition	8
2.1.	Auftraggeber.....	8
2.2.	Projektbetreuer	8
3.	Entstehung.....	9
3.1.	Ist – Zustand	9
3.2.	Motivation	9
4.	Diabetes mellitus.....	10
4.1.	Allgemein.....	10
4.1.1.	Was ist Diabetes?.....	10
4.1.2.	Diabetes Typen	11
4.1.3.	Insulin	12
4.1.4.	Broteinheiten.....	13
4.1.5.	Glukose	13
4.1.6.	Kohlenhydrate.....	14
4.1.7.	Hypoglykämie.....	15
4.1.8.	Hyperglykämie.....	16
4.2.	Entstehung.....	17
4.2.1.	Ursachen bei Diabetes mellitus Typ 1	17
4.2.2.	Ursachen bei Diabetes mellitus Typ 2	18
4.3.	Berechnung.....	19
4.4.	Behandlung.....	20
4.4.1.	Insulintherapie.....	20
4.4.2.	Pumpe	22
4.4.3.	Pen.....	23
5.	technische Grundlagen.....	24
5.1.	verwendete Technologien	24
5.1.1.	HTML/CSS3.....	24
5.1.2.	MySql.....	25
5.1.3.	Windows Server 2012 R2	26
5.1.4.	Visual Studio.....	26
5.1.5.	ASP.NET	27

5.1.6.	Entity Framework	28
5.1.7.	Linq	29
5.1.8.	jQuery	30
5.1.9.	JSON	31
5.1.10.	SmartGit	34
6.	Benutzung	35
6.1.	Programmablauf	35
6.1.1.	Startseite	35
6.1.2.	Login	36
6.1.3.	Über mich	38
6.1.4.	Berechnung	39
6.1.5.	Informationen	40
6.1.6.	Lebensmittelliste	41
6.1.7.	Statistik	43
6.2.	Use – Case Diagramm	44
6.3.	Ergebnis	48
7.	Realisierung	49
7.1.	Projektstrukturplan	49
7.2.	Aufwandsschätzung	50
7.3.	Schnittstellen und Datenfluss (P)	51
7.3.1.	MySql Connector	51
7.3.2.	MySql	52
7.3.3.	MySql Workbench	53
7.3.4.	IIS Server Konfiguration	54
8.	Implementierung	55
8.1.	Model-View-Controller	55
8.2.	Allgemein	57
8.3.	Funktionen	59
8.3.1.	Login	59
8.3.2.	Stammdaten eingeben	61
8.3.3.	Lebensmittelliste anzeigen	62
8.3.4.	Lebensmittel zur Lebensmittelliste hinzufügen	65
8.3.5.	Lebensmittel auswählen	67
8.3.6.	Informationen einholen	68
8.3.7.	Statistik anzeigen	68
8.3.8.	Berechnungen	69
8.4.	Datenbankmodell	70

9.	UML Klassendiagramm.....	74
10.	Qualitätssicherung.....	77
10.1.	Aufgetretene Probleme und Lösungen	77
10.1.1.	E-Mail Adresse statt Benutzername.....	77
10.1.2.	MySql Connector erstellen	77
10.1.3.	Verbindung zum Server	78
10.1.4.	Bilder anzeigen	78
10.2.	Qualitätsmerkmale.....	79
10.2.1.	Funktionalität	79
10.2.2.	Bedienbarkeit	79
10.2.3.	Portabilität.....	80
10.2.4.	Installierbarkeit	80
10.2.5.	Korrektheit	80
10.2.6.	Erlernbarkeit	81
10.2.7.	Ressourcen	81
10.3.	Testfälle	82
10.3.1.	Testfall: 1	83
10.3.2.	Testfall: 2	84
10.3.3.	Testfall: 3	85
11.	Evaluierung.....	86
11.1.	Kompatibilität.....	86
11.2.	Voraussetzungen.....	86
11.2.1.	internetfähiges Gerät.....	86
11.2.2.	aktive Internetverbindung	86
11.2.3.	Blutzuckermessgerät	87
11.3.	Resümee.....	88
12.	Impressum.....	89
12.1.	Diplomanden	90
3.3.1	Tamara Holzer	90
3.3.2	Petra Enengl	90
3.3.3	Patrick Mittermayr	90
13.	Anhänge	91
13.1.	Quellenverzeichnis	91
13.2.	Abbildungsverzeichnis	93

1. Einführung

1.1. Kurzfassung

Die Diplomarbeit *Diabetes Helfer für Kinder* ist während des fünften Jahrgangs von Holzer Tamara, Enengl Petra und Mittermayr Patrick im Zuge der Reife- und Diplomprüfung an der Technischen Bundeslehranstalt Perg erstellt worden.

Als Auftraggeber tritt eine externe Person ein, deren Kind an Diabetes Typ 1 leidet.

Ein Diabetiker muss einige Berechnungen durchführen, um Lebensmittel zu sich nehmen zu können. Die einzunehmenden Kohlenhydrate werden in Broteinheiten umgewandelt, um das zu verabreichende Insulin zu ermitteln. Weiters muss der aktuelle Blutzuckerwert gemessen und interpretiert werden, das heißt ob dieser mittels des Insulins korrigiert werden muss. Aufgrund dieser zwei Werte ergibt sich der zu verabreichende Insulinwert.

Diese vorwissenschaftliche Arbeit beschäftigt sich mit der genauen Erfassung der Kohlenhydrate pro Lebensmittel. Weiters wird die verabreichte Insulinmenge für eine Mahlzeit berechnet. Dabei wird insbesondere auf die Berechnung eingegangen und alle verfügbaren Lebensmittel, welche sich bereits in einer Liste befinden, grafisch dargestellt. Ebenfalls werden mittels einer Statistik der Blutzuckerwert und der Insulinwert grafisch dargestellt. Aus dieser Grafik kann man zu einem bestimmten Zeitpunkt diese zwei Werte betrachten.

Das Ziel unserer Diplomarbeit besteht darin, dass bereits im Kleinkindalter der Umgang mit einer lebenslangen Krankheit zu leben, erlernt wird. Mit der grafischen Darstellung der Lebensmittel kann ein Kind das gewünschte Produkt auswählen und den Insulinwert berechnen lassen. Das Programm hilft dabei dem Kind eigenständig zu lernen und bereits im frühen Alter die Krankheit zu kennen und den Umgang damit zu erforschen.

1.2. Abstract

The diploma project *Diabetes Helfer für Kinder* has been made during the fifth grade by Holzer Tamara, Enengl Petra and Mittermayr Patrick in the course of the matura and diploma exam in the Federal Higher Technical Institute Perg.

The client is an external person, whose child is affected with type 1 diabetes.

A diabetic has to do some calculations in order to eat the food they want without risking serious harm to their health and even lives. The person affected must convert the carbohydrates taken during the meal into bread units and by doing this he or she can calculate how much insulin has to be injected.

Additionally, the current blood sugar value must be measured and interpreted, to determine if it has to be changed with insulin or not. Based on these two values, the amount of the necessary insulin is calculated.

This project deals with the exact calculation of the carbohydrates per food item. Moreover, the insulin dosage is calculated for a meal. Thereby, there will be a particular regard to the calculation and all available groceries, which can already be found in a list where they are shown graphically. Furthermore, the blood sugar value and the insulin value will be shown in a statistic. In this statistic, it is possible to compare these two values at a certain period of time.

The aim of our project is to teach the children how to deal with a lifelong illness. Due to the graphical illustration of the groceries, a child can select the desired food and calculate the appropriate insulin value. The program helps kids to learn independently and gets them to understand the illness already at an early age.

2. Projektdefinition

2.1. Auftraggeber

Auftraggeber unserer vorwissenschaftlichen Arbeit ist eine private Person, deren Kind an Diabetes Typ 1 leidet. Die Diagnose ist bereits im Kleinkindalter festgestellt worden. Ab diesem Zeitpunkt hat sich die Lebensweise der ganzen Familie von Grund auf geändert. Das Kind darf nur zu definierten Zeitpunkten eine Mahlzeit einnehmen. Bei den Mahlzeiten müssen die Kohlenhydrate von den zu verzehrenden Lebensmitteln ermittelt und auf Broteinheiten umgerechnet werden. Weiters muss die Insulinmenge berechnet werden, welche dem Kind für diese Mahlzeit verabreicht werden muss, da es sonst zu einer Überzuckerung kommen kann.

Das Kind besucht derzeit den Kindergarten. Damit das Kind überhaupt einen Kindergarten besuchen kann, müssen die Mitarbeiter dieses Kindergartens eine spezielle Schulung absolvieren. Da das Kind im Kindergarten eine Mahlzeit einnehmen muss, sind die Mitarbeiter des Kindergartens dafür verantwortlich, dass die verabreichende Insulinmenge korrekt bestimmt wird. Wenn sich der verantwortliche Mitarbeiter bezüglich der Insulinmenge unsicher ist, wird die Mutter des Kindes kontaktiert und gemeinsam eine Insulinmenge berechnet.

2.2. Projektbetreuer

Name:	Dipl.Ing. Dr. Michael Buchberger
Funktion:	Betreuung der Arbeit
E-Mail:	m.buchberger@htl-perg.ac.at

3. Entstehung

3.1. Ist – Zustand

Derzeit ist es ein großer Aufwand, die Kohlenhydrate der benötigten Lebensmittel abzulesen und diese in Listen einzutragen. Da die Kohlenhydrate je nach Produkt unterschiedlich zu finden sind, ist dies sehr aufwendig und zeitintensiv. Die Kohlenhydrate der zu verspeisenden Produkte werden dann in Broteinheiten umgerechnet. Danach werden mittels des Umrechnungsfaktors, welcher im Krankenhaus ermittelt wird, die Broteinheiten in Insulineinheiten umgerechnet. Der betroffene Diabetiker erfährt dadurch, wie viel Insulin er sich selbst spritzen muss, damit er diese Mahlzeit zu sich nehmen kann.

3.2. Motivation

Durch unseren externen Auftraggeber erfuhren wir, dass sich nach Feststellung der Krankheit die Berechnungen für die zu verabreichende Insulineinheit recht schwierig gestalten. Dadurch angespornt, wollten wir ein Programm entwickeln, welches diese Phase etwas erleichtert.

Ein Hauptaugenmerk lag dabei auf der Benutzerfreundlichkeit, da es auch ermöglicht werden soll, dass Kinder selbst das Programm bedienen. Dies zeichnet sich durch kinderfreundliche Optik der Benutzeroberfläche aus, indem der Aufbau der Webseite einfach geregelt ist und auch viele Bilder eingebaut sind.

Der Benutzer hat die Möglichkeit in der Lebensmittelliste die Kohlenhydratwerte der Nahrungsmittel einzusehen und gegebenenfalls neue Lebensmittel mit deren Informationen einzutragen. Mittels unseres Programms wird die Berechnung der Insulinwerte vereinfacht und die Ergebnisse können in einer Statistik angezeigt werden, wo sie über einen gewissen Zeitraum miteinander vergleichbar sind. Somit kann man sehen, wie sich die Ernährung auf die Gesundheit auswirkt.

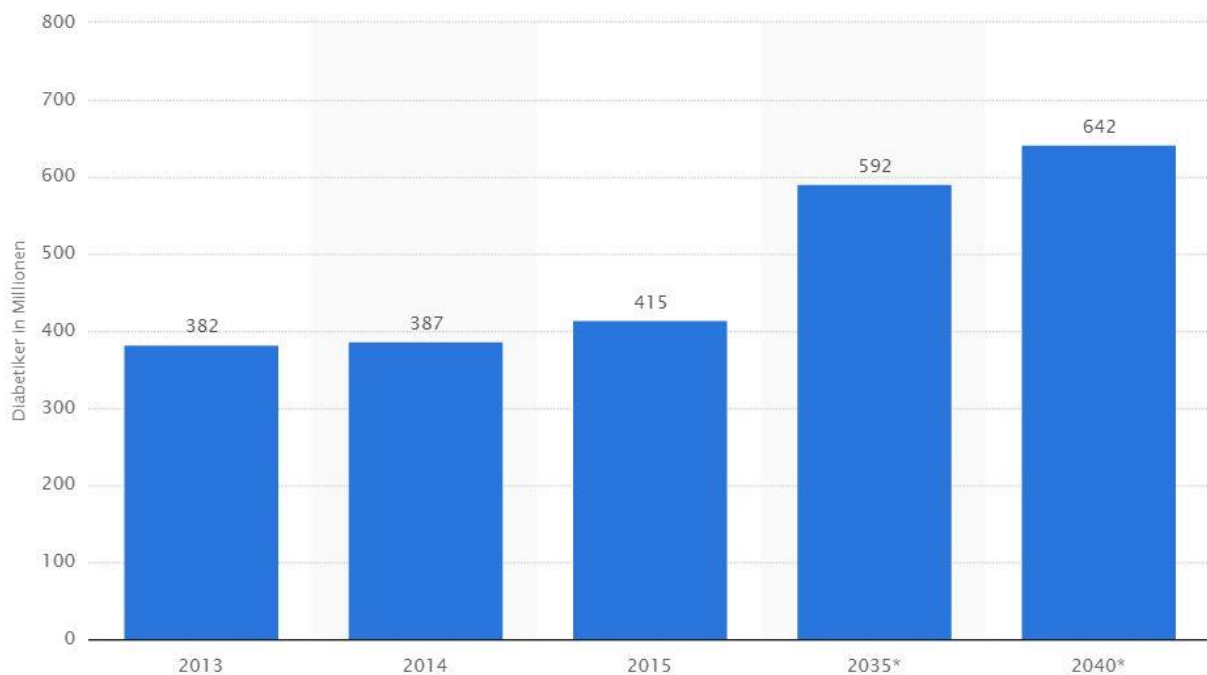
Wir möchten mit dieser Arbeit Informationen über das Thema Diabetes zur Verfügung stellen, weil die Zahl der Erkrankungen in den letzten Jahren anstieg und somit vermehrt Bedarf an solch einer Technologie gegeben ist.

4. Diabetes mellitus

4.1. Allgemein

4.1.1. Was ist Diabetes?

Diabetes mellitus ist eine Stoffwechselkrankheit, bei derer die Werte des Blutzuckers nicht der Norm entsprechen, sondern zu hoch sind. Ein Grund, für diese Krankheit, kann die Insulinresistenz sein. Dies bedeutet, dass die Wirksamkeit des Insulins vermindert ist. Ein anderer Grund, der zum Ausbruch dieser Krankheit führen kann, ist der Insulinmangel im Körper. Diabetes mellitus hat sich in den letzten Jahren zu einer „Volkskrankheit“ entwickelt. Eine Statistik [de.statista.com, 2016], zeigt, dass im Jahre 2015 415 Millionen Menschen weltweit an Diabetes mellitus erkrankt sind. 2013 waren es noch 382 Millionen und für das Jahr 2040 wird eine Zahl von 642 Millionen Diabetikern weltweit vorausgesagt.



© Statista 2016

Abbildung 1: Statistik Diabetes weltweit

4.1.2. Diabetes Typen

Es werden prinzipiell zwischen drei Diabetestypen unterschieden:

- **DIABETES MELLITUS TYP 1** [Pflegeassistenz, 2011]

Diese Art des Diabetes wurde früher auch Jugenddiabetes genannt. Dieser Typ ist eher selten und tritt meist nur bei Kindern und Jugendlichen auf.

- **DIABETES MELLITUS TYP 2** [Pflegeassistenz, 2011]

Diese Art des Diabetes wurde früher auch Altersdiabetes genannt. Dieser Typ tritt sehr häufig auf, meist bei Erwachsenen ab dem 40. Lebensjahr.

- **SCHWANGERSCHAFTSDIABETES** [Diabetes Ratgeber, 2015]

Diese Art des Diabetes wird auch Gestationsdiabetes genannt. Während der Schwangerschaft entwickelt sich eine diabetische Stoffwechsellage. In den meisten Fällen verschwindet dieser Diabetes wieder.

95% der Diabetesbetroffenen sind an Typ 1 oder Typ 2 erkrankt, dennoch gibt es neben den obig genannten Formen andere, welche aber eher selten auftreten.

Beispiele können sein:

- Diabetes durch Erkrankungen (z.B.: Bauchspeicheldrüsenentzündung)
- Diabetes durch Hormonstörungen
- Diabetes, welches durch Medikamente ausgelöst wird

4.1.3. Insulin

Insulin ist ein Hormon, welches in der Bauchspeicheldrüse gebildet wird. Es ist als einziges in der Lage, den Blutzuckerwert zu senken. Der Gegenspieler von Insulin ist die Glukose.

WIRKUNG VON INSULIN

Eine Insulinausschüttung kann zu einer Senkung des Blutzuckers führen und fördert den Aufbau von Energiedepots. Das Hormon hilft, den Blutzucker in die Zellen zu transportieren.

Als Folge davon, sinkt der Blutzuckerspiegel, was innerhalb von ein paar Minuten passieren kann. In der Leber und den Muskelzellen wird der Aufbau des Zuckerspeichers (Glykogen) gefördert. Außerdem hemmt Insulin auch den Umbau von Aminosäuren in Glukose und fördert den Fettaufbau.

ÜBERWACHUNG DER INSULINFREISETZUNG

Die Bauchspeicheldrüse gibt jeweils nach Bedarf Insulin ins Blut ab. Dabei ist ein wichtiger Wert der Blutzuckerspiegel. Ist dieser niedrig, wird kaum Insulin ins Blut abgegeben. Dies tritt zum Beispiel auf, wenn keine Glukose über die Nahrung zugeführt wird.

Gelangt Glukose ins Blut, wird mehr Insulin durch die Bauchspeicheldrüse abgegeben und der Zucker wird vom Blut in die Körperzellen transportiert. Dies führt dazu, dass der Blutzuckerspiegel kurze Zeit nach der Nahrungsaufnahme wieder im Normalbereich liegt.

Außerdem sorgt das Hormon Insulin auch dafür, dass die Eiweißbestandteile in die Zelle aufgenommen und zum Eiweißaufbau verwendet werden. Dadurch regt natürlich auch der Eiweißbestand die Insulinausschüttung an. [MERCK GESELLSCHAFT MBH, 2016]

INSULINPRÄPARATE

Insulin wird in drei Gruppen unterteilt:

- **Kurz wirksame Insuline**
Dieses Insulin wird direkt zu den Mahlzeiten gespritzt und senkt den Blutzucker nach zirka 10-15 Minuten.
- **Verzögerungsinsuline**
Dies ist eine Kombination von Insulin und Verzögerungsstoffen. Sie werden ein- bis zweimal täglich als Vorsorge gespritzt, da hierbei die Insulinwirkung verzögert wird. Die Wirkung hält länger an und entfaltet sich erst nach zirka drei Stunden.
- **Insulinmischungen**
Dies wird meist nach einem gewissen Schema mehrmals täglich gespritzt. Es enthält Konzentrationen von kurz wirksamen Insulin und Verzögerungsinsulin.

[Pflegeassistenz, 2011]

4.1.4. Broteinheiten

Eine Broteinheit definiert eine Menge an Nahrungsmittel, welche 12g Kohlenhydrate enthält. Sie werden im Rahmen einer Insulintherapie ermittelt, damit der jeweilige Insulinbedarf abgeschätzt wird. [DR. EIGLER, Beate, 2013]

4.1.5. Glukose

Glukose ist der Blutzucker und der Gegenspieler vom Insulin. Die Messung des Blutzuckerspiegels ist der wichtigste Labortest zur Diagnose von Diabetes mellitus.

[BUNDESMINISTERIUM FÜR GESUNDHEIT, 2015]

Die Glukose ist für die Körperzellen ein wichtiger Energielieferant, insbesondere für das Gehirn. Damit Glukose in die Zellen gelangt, wird das in der Bauchspeicheldrüse gebildete Hormon Insulin benötigt.

4.1.6. Kohlenhydrate

Kohlenhydrate sind neben Fett und Eiweiß die Hauptnährstoffe in der Ernährung. Beim Abbau von Kohlenhydrate entsteht im Körper Energie, welche in Kalorien oder Joule gemessen wird. Die täglich aufgenommene Energie sollte zu 55% aus Kohlenhydrate, höchstens 30% aus Fett und 15% Eiweiß bestehen, damit alle Stoffwechselfunktionen im Körper optimal ablaufen.

Bei einer Durchschnittsperson liegt der Gesamtenergiebedarf am Tag bei 250 bis 360 Gramm Kohlenhydrate. Die Mindestmenge an Kohlenhydrate, die Menschen brauchen, liegt bei 2 Gramm je Kilogramm Körpergewicht.

Kohlenhydrate haben im Körper viele unterschiedliche Funktionen. Zum einen sind sie Energiequellen zum anderen, haben sie die Aufgabe als Stützfunktion, da komplexe Kohlenhydrate Bestandteile von Sehnen, Knochen und Bindegeweben sind. Sie sind auch Energiereserven, da Kohlenhydrate als Glykogen in der Leber und der Muskulatur gespeichert werden und bei Bedarf Energie sehr schnell bereitgestellt werden kann. Außerdem sind diese Nährstoffe auch für den Wasser- und Elektrolythaushalt wichtig, da sie notwendig sind, um diesen Haushalt im Körper aufrecht zu erhalten.

Man unterscheidet drei Arten von Kohlenhydraten:

- Einfachzucker (Monosaccharide)
 - Glukose (Süßigkeiten und Traubenzucker im Obst)
 - Fruktose (Fruchtzucker)
 - Galaktose (Milchzucker)
- Zweifachzucker (Disaccharide)
 - Saccharose (Rübenzucker und Haushaltszucker)
 - Laktose (Milchzucker)
 - Maltose (Malzzucker im Bier)
- Vielfachzucker (Polysaccharide)
 - Stärke (in Getreide und Gemüse)
 - Glykogen im Muskelfleisch
 - Ballaststoffe (in Vollkorngetreide, Gemüse und Obst)

[MAG. TIEFENBÖCK, Waltraut, 2014]

4.1.7. Hypoglykämie

Hypoglykämie ist der Fachbegriff für den Unterzucker. Dies bedeutet, dass zu wenig Zucker im Blut vorhanden ist.

SYMPTOME EINER HYPOGLYKÄMIE

- starkes Zittern
- man schwitzt sehr stark
- man fühlt sich schwach
- die Konzentration wird gestört
- man hat plötzliche Heißhungerattacken

Die Symptome treten nicht immer gleich und meist sehr plötzlich auf.

URSACHEN EINER UNTERZUCKERUNG

- zu wenig Zufuhr von Kohlenhydraten
- zu viel Zufuhr von Insulin
- zu viel Bewegung, insbesondere Sport
- zu hoher Alkoholkonsum

4.1.8. Hyperglykämie

Als Hyperglykämie wird ein erhöhter Blutzuckerwert bezeichnet. Dies bedeutet, zu viel Zucker befindet sich im Blut.

SYMPTOME EINER HYPERGLYKÄMIE

- erhöhte Harnausscheidung
- Durstgefühl
- Gewichtsverlust
- Bewusstseinstörung

Auch hier treten die Symptome nicht immer gleich und meist sehr plötzlich auf.

URSACHEN EINES ERHÖHTEN BLUTZUCKERWERTES

- zu viel Zufuhr von Kohlenhydraten
- zu wenig Zufuhr von Insulin
- zu hoher Alkoholkonsum

4.2. Entstehung

In der Bauchspeicheldrüse, fachlich Pankreas genannt, wird in den Langerhans Inseln Insulin gebildet. Um den Zucker aus der Blutbahn aufzunehmen, brauchen Körperzellen dieses Hormon, da der Körper den aufgenommenen Zucker zur Energiegewinnung benötigt. Können aber Zellen genau diesen Zucker nicht aufnehmen, steigt der Blutzuckerspiegel im Blut an. Wird eine gewisse Menge an Glukose im Blut überschritten, gelangt der Zucker in den Harn und wird ausgeschieden. [DR. EIGLER, Beate, 2013]

4.2.1. Ursachen bei Diabetes mellitus Typ 1

Die Zellen, welche Insulin produzieren, werden durch Antikörper, welcher der Körper bildet, zerstört. Dies passiert, wenn das Immunsystem oder der Körper selbst die eigenen Betazellen, aus bisher unbekanntem Ursachen, angreift. Es kann in kurzer Zeit zu Beschwerden kommen, da die Krankheit sehr rasch fortschreitet.

Um den Blutzucker zu senken, fehlt den Betroffenen das Insulin. Dies hat zur Folge, dass der Blutzuckerwert erhöht ist. Diabetes mellitus Typ1 bleibt ein Leben lang bestehen, da es nicht heilbar ist. [MERCK GESELLSCHAFT MBH, 2016]

SYMPTOME

Diabetes Mellitus Typ 1 tritt sehr plötzlich auf. Dabei können folgende Symptome auftreten:

- starkes Durstgefühl
- Gewichtsverlust
- Konzentrationsschwäche und Leistungsschwäche
- trockene Haut und Juckreiz
- vermehrte Harnausscheidung
- Übersäuerung des Blutes

4.2.2. Ursachen bei Diabetes mellitus Typ 2

Etwa 90% der Diabeteserkrankten sind an Diabetes mellitus Typ 2 erkrankt. Die Ursache dieser Form ist nicht vollständig geklärt. Allerdings weiß man, dass der Körper nicht mehr richtig auf das Insulin anspricht. Die Bauchspeicheldrüse ist anfangs noch voll funktionsfähig und produziert Insulin. Dabei wirkt dieses aber nicht so, wie es sein sollte, da die Zellen trotz Insulin weniger Glukose aus der Blutbahn aufnehmen. Dadurch bleibt der Blutzuckerspiegel hoch.

Diabetes Typ 2 entwickelt sich schleichend. Der Zuckerstoffwechsel erscheint anfangs normal, obwohl schon krankhafte Prozesse laufen. Die noch gesunde Bauchspeicheldrüse greift anfangs noch korrigierend ein, indem sie mehr Insulin produziert.

Schreitet diese Krankheit jedoch voran, ändert sich auch der zuvor beschriebene Zustand. Zwischen den Mahlzeiten kann der Blutzucker im normalen Bereich liegen. Nach den Mahlzeiten jedoch, wird die Glukose jedoch langsamer als üblicherweise aus dem Blut in die Zellen aufgenommen. Die führt dazu, dass der Blutzucker nach den Mahlzeiten zu lange erhöht bleibt. [MERCK GESELLSCHAFT MBH, 2016]

SYMPTOME

Diabetes Mellitus Typ 2 wird häufiger festgestellt, weil die Symptome auffälliger sind. Folgende Symptome können auftreten:

- erhöhter Blutzucker
- Austrocknung des Körpers
- Müdigkeit und Abgeschlagenheit
- Pilzerkrankungen und Hautjucken

[Pflegeassistenz, 2011]

4.3. Berechnung

Für die Berechnung der zu verabreichenden Insulineinheiten müssen viele Faktoren beachtet werden:

Vor jedem Essen, wird zuerst der Blutzuckerwert gemessen. Ist dieser zu hoch oder zu niedrig, muss dies auch bei der Eingabe der Insulineinheit berücksichtigt werden.

Bei einer Hyperglykämie wird pro zehner Schritt 0,1 IE¹ dazu addiert. Angenommen man misst vor dem Essen den Blutzucker und hat einen Wert von 150mg, dann muss man 0,5 IE zusätzlich zu den IE der zu verzehrenden Lebensmittel hinzurechnen.

Schwieriger wird es bei einer aufgetretenen Hypoglykämie. Eigentlich müsste man mehr Essen oder einen Traubenzucker zu sich nehmen. In vielen Fällen wird aber meist eine BE² nicht berücksichtigt. Zum Beispiel misst man vor dem Essen einen Blutzuckerwert von 50mg und isst 4 BE. Dann werden hierbei aber nur 3 BE bei der Berechnung berücksichtigt.

Zusätzlich zum aktuellen Blutzuckerwert muss auch noch der Umrechnungsfaktor berücksichtigt werden. Dieser ist bei jedem Betroffenen unterschiedlich und kann sich auch laufend verändern. Dieser Faktor wird mit den Broteinheiten der zu verzehrenden Produkte multipliziert.

Außerdem müssen die Broteinheiten der Produkte dementsprechend in die Berechnung einfließen. Die Kohlenhydrate sind meist auf der Verpackung der Produkte angegeben. Eine BE beinhaltet 12g Kohlenhydrate.

Die Insulineinheiten berechnen sich somit mit der Anzahl der Broteinheiten multipliziert mit dem Umrechnungsfaktor unter Berücksichtigung des Blutzuckerwertes.

¹ IE = Insulineinheit

² BE = Broteinheit

4.4. Behandlung

Grundsätzlich gibt es keine Heilung von Diabetes, ausgenommen der Schwangerschaftsdiabetes. Das absolut oberste Ziel der Therapie ist immer die Normalisierung des Blutzuckers. Dies kann durch die Anpassung des Lebensstils und zusätzlichem Einsatz von Medikamenten erfolgen.

4.4.1. Insulintherapie

Kann die Bauchspeicheldrüse selbst nicht mehr genügend Insulin produzieren, muss die Therapie zusätzlich durch Zugabe von Insulin ergänzt werden. Eine Insulintherapie ist in Form von Tabletten zum jetzigen Stand nicht möglich, da dieses Insulin im Verdauungsapparat verdaut werden würde. Daher muss das Insulin mittels einer Spritze unter die Haut verabreicht werden. Diese Technik nennt man auch subkutane Injektion.

Bei einer richtigen Durchführung einer subkutanen Injektion ist dies nahezu schmerzlos und einfach zu erlernen. Die Verabreichung kann von den Betroffenen nach einer Schulung problemlos selbst durchgeführt werden.

Bei der Insulintherapie ist unbedingt zu beachten, dass nicht zu viel Insulin verabreicht wird. Passiert dies, sinkt der Blutzucker stark ab und es kann zu einer Hypoglykämie kommen. Eine Unterzuckerung kann aber auch eintreten, wenn die korrekte Insulindosis verabreicht worden ist, aber zu wenig Nahrung zu sich genommen wird oder man sich stark körperlich betätigt und somit der Glukoseverbrauch gesteigert wird.

[MERCK GESELLSCHAFT MBH, 2016]

Man unterscheidet zwischen einer konventionellen und einer intensivierten Insulintherapie.

KONVENTIONELLE INSULINTHERAPIE

Es wird meist nach einem festgelegten Plan zwei- bis dreimal täglich Mischinsulin verabreicht. Auf dieses muss die Nahrung dann abgestimmt werden. Nach dem Verabreichen von Insulin sollte zirka 20-30 Minuten mit der Mahlzeit gewartet werden, aufgrund der verzögerten Wirkung des Insulins.

INTENSIVIERTE INSULINTHERAPIE

Ein- bis zweimal täglich wird zusätzlich zur Basisinsulinversorgung ein Verzögerungsinsulin gespritzt und zu den jeweiligen Mahlzeiten wird ein kurz wirksames Insulin verabreicht.

Bei einer Insulinpumpentherapie wird mittels einer Pumpe permanent eine bestimmte Menge an kurz wirksamen Insulin abgegeben.

[Pflegeassistenz, 2011]

4.4.2. Pumpe

Dies ist ein kleines, batteriebetriebenes Hilfsmittel für Diabetiker. Sie ersetzt die Insulinspritze oder den Pen. Es hat in etwa die Größe eines Handys und wird außerhalb und permanent am Körper getragen.

Die Insulinpumpe gibt permanent eine kleine Menge Insulin ins Blut ab und ersetzt somit die Funktion der Bauchspeicheldrüse. Bei diesem Gerät kommt nur das kurzwirksame Insulin zum Einsatz. Nimmt ein Diabetiker Nahrung zu sich, ruft er über die Pumpe, mit einem Knopfdruck, zusätzlich Insulin ab. Dabei wird die Insulinabgabe eines gesunden Menschen imitiert. [MSC. QUIRGST, Helga et al., 2015]



Abbildung 2: Insulinpumpe

4.4.3. Pen

Ein Insulinpen hat die Form einer Füllfeder. Er ist einfacher zu bedienen als eine Spritze und dient zur mehrmaligen Injektion einer eingestellten Insulindosis. Er beinhaltet eine Ampulle und eine Dosisvorrichtung.

Es werden zwei verschiedene Insulinpens für ein Verzögerungs- und ein kurz wirkendes Insulin verwendet. Die Nadeln werden nach jeder Injektion gewechselt. [Pflegeassistent, 2011]



Abbildung 3: Insulinpen

5. technische Grundlagen

5.1. verwendete Technologien

5.1.1. HTML/CSS3

Hypertext Markup Language 5 ist eine textbasierte Auszeichnungssprache. HTML baut auf eine Struktur auf, welche genutzt wird, um digitale Inhalte darzustellen. Beispiele dieser Inhalte können Bilder, Texte oder Hyperlinks sein. [PFEIFFER, Thomas]

HTML 5 bietet neue Funktionen wie Video, Audio, lokalen Speicher und dynamische 2D- und 3D-Grafiken anzuzeigen.

Cascading Style Sheets ist zusammen mit HTML und DOM eine Kernsprache des Internets.

DOM (Document Object Model) definiert eine logische Struktur für Dokumente, das heißt es beschreibt eine Schnittstelle zum Aufruf von HTML und XML Dokumenten. [ALTMANN, U., 2013]

CSS ist eine Stylesheet-Sprache mit der Gestaltungsanweisungen erstellt werden, welche mit HTML und XML eingesetzt wird. [Wikipedia, 2015]

HTML5 und CSS3 sind die technische Basis für Responsive Design, welches man zum Erstellen von Webseiten verwendet. Das Besondere dabei ist, dass der Code zur Darstellung der Seite einmal programmiert wird, aber die Webseite die Eigenschaft der Endgeräte, z.B. Smartphone oder Tablet, erkennt und darauf reagiert. Die Ansicht der Webseite passt sich dabei an die Endgeräte an. [Wikipedia, 2015]

5.1.2. MySQL

MySQL ist ein relationales Datenbankverwaltungssystem, welches von der Oracle Corporation entwickelt wurde. Als Open Source Software ist *MySQL* für verschiedene Betriebssysteme verwendbar. Am häufigsten wird diese Datenbank für Webserver und für die Skriptsprache PHP eingesetzt.

Derzeit befindet sich Version 5.7.11 am Markt. Zu Performancezwecke ist *MySQL* in *ANSI C* und *ANSI C++* implementiert.

Vorteile:

- verbraucht kaum Ram
- leichte Installation
- kostenlos
- guter Support

Nachteile:

- alle Features sind nur in der kostenpflichtigen Variante vorhanden

[Wikipedia MySQL, 2016]

5.1.3. Windows Server 2012 R2

Der *Windows Server 2012* ist eine Erweiterung des *Windows Servers 2008* und umfasst Verbesserungen im Bereich Cloud – Computing und Storage-Infrastruktur. Vor allem bietet der *Windows Server 2012* Erweiterungen in folgenden Punkten:

- grafische Benutzeroberfläche (GUI)
- Adressverwaltung

[SearchDataCenter, 2014]

5.1.4. Visual Studio

Visual Studio ist eine von Microsoft bereitgestellte Entwicklungsumgebung, welche es ermöglicht in verschiedenen Programmiersprachen Programme zu entwickeln. Es kann zur Erstellung von Konsolenapplikationen, GUI (Graphical User Interfaces), *Windows Forms*, *Web Services* und *Web-Applikationen* verwendet werden.

Weiters werden noch zahlreiche andere Entwicklungswerkzeuge angeboten. Durch die umfassende Plattformunterstützung können sowohl mobile Anwendungen als auch *Cloud Services* entwickelt werden.

Visual Studio unterstützt auch agile Softwaremethoden. Es stellt zum Beispiel *Backlogs* zur Verfügung. Des Weiteren stellt das Programm Features zur Verfügung, wie das *Debugging* oder die *Software Tests*.

5.1.5. ASP.NET

ASP.NET (Active Server Pages .NET) ist eine Bibliothek, welche sich auf die Programmierung von dynamischen Webseiten spezialisiert, welches programmiersprachenunabhängig ist. Dabei ist es auf keine Programmiersprache gebunden. Programmiert wird in Scripts, welche dann vom Server beim Aufruf interpretiert werden, das Ergebnis wird dabei an den Klient geschickt.

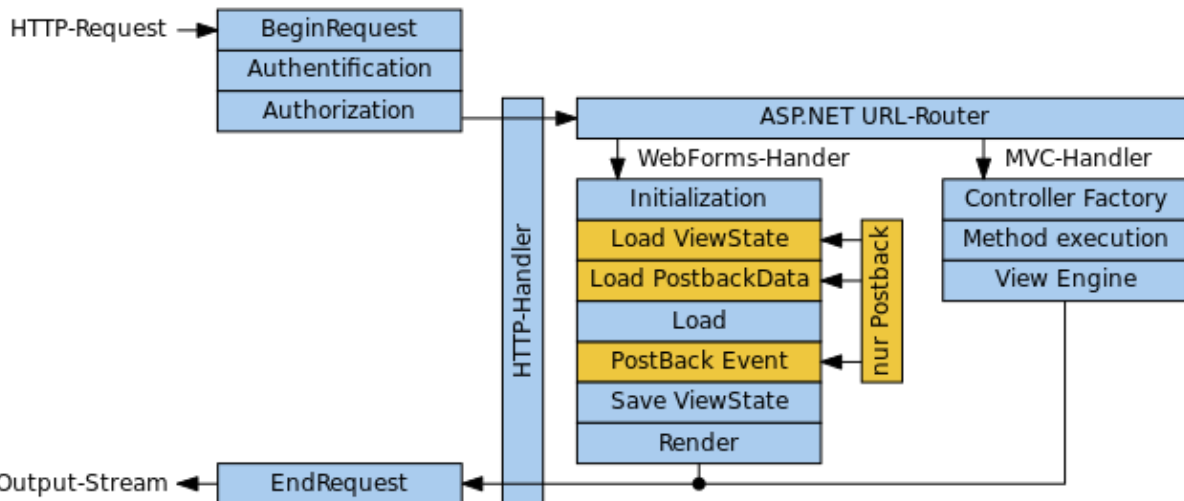


Abbildung 4: ASP.NET Funktionsweise

[Tutorialspoint, 2016]

5.1.6. Entity Framework

Das *Entity Framework* ist eine Erweiterung von ADO.NET und eine objektrelationale Abbildung, welche einen Mechanismus für die Verwendung und Speicherung von benötigten Daten aus einer Datenbank zur Verfügung stellt. Dadurch erspart sich der Entwickler den Großteil an Code, welcher notwendig wäre, um die Daten aus der Datenbank zu lesen. Mit Entity Framework erstellen die Entwickler *Queries*, welche *LINQ* benutzen, die wiederum dann Daten aufrufen und bearbeiten.

Entity Framework ist bei folgenden Szenarien sinnvoll:

- Wenn bereits eine Datenbank vorhanden ist oder man die Datenbank vor den Rest der Applikation erstellen will.
- Wenn man sich auf die Domain Klassen konzentriert und seine Datenbank auf Grundlage dieser Klassen aufbaut.
- Wenn man seine Datenbank mittels Visual Designer entwerfen will und erst danach die Klassen erstellt.

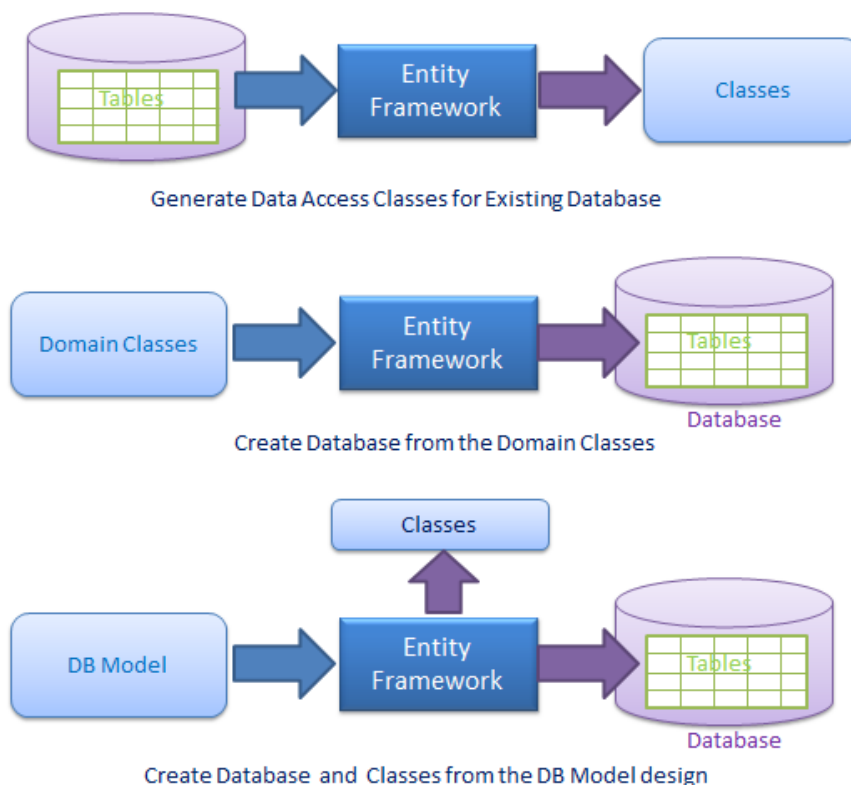


Abbildung 5: 3 Szenarien

[Entity Framework Tutorial, 2015]

5.1.7. Linq

Linq (Language Integrated Query) ist eine Abfragesprache, die in C# integriert ist. Die Daten werden mittels SQL ähnlicher Syntax abgefragt und aktualisiert.

Linq besteht aus einer 3-schichtigen Architektur, wo die oberste Schicht aus den Spracherweiterungen besteht und die untere Schicht aus Datenquellen, welche üblicherweise Objekte sind, die IEnumerable<T> oder IQueryable<T> Interfaces implementieren.

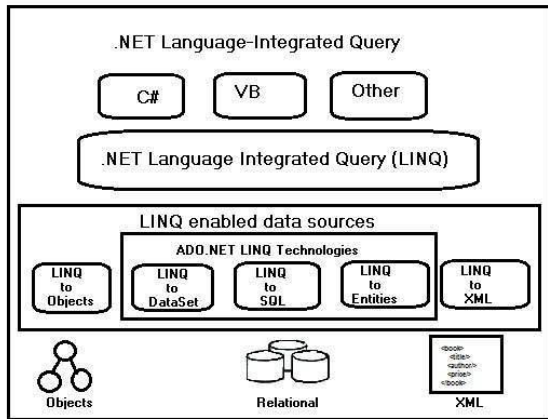


Abbildung 6: LINQ Architektur

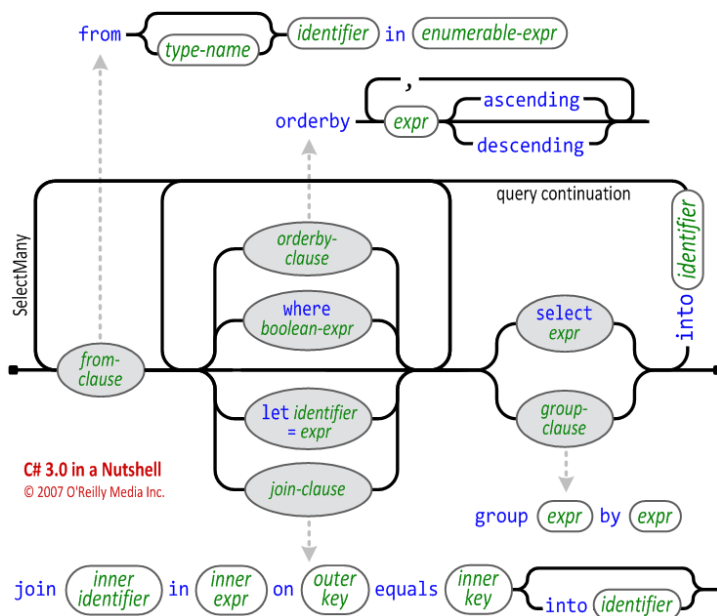


Abbildung 7: LINQ Syntax

TYPEN VON LINQ

- LINQ zu Objekten
- LINQ zu XML
- LINQ zu Datensatz
- LINQ zu SQL
- LINQ zu Entities

WICHTIGE OPERATOREN

- From
- Where
- Select
- Group
- Into
- OrderBy
- Join
- Reverse

[Tutorialspoint, 2016]

5.1.8. jQuery

jQuery ist eine schnelle, kompakte und funktionsreiche JavaScript Bibliothek und steht als Open-Source-Software kostenfrei zur Verfügung. Es vereinfacht die Bearbeitung von HTML-Objekten, Event Handling, Animationen und Ajax, indem eine einfache API verwendet wird und über mehrere Browser funktioniert.

Die jQuery Bibliothek besitzt folgende Funktionen:

- HTML/DOM Bearbeitung
- CSS Bearbeitung
- HTML Event Methoden
- Effekte und Animationen
- AJAX

Utilities JQuery hat bei Verwendung bei JavaScript-Anwendungen folgende Vorteile:

- leicht zu lernen
- der Aufwand für Code und Zeit werden reduziert
- das Laden der Bilder des Dokuments entfällt
- JavaScript wird um weitere Funktionen erweitert [Wikipedia JQuery, 2016]

5.1.9. JSON

JSON (JavaScript Object Notation) ist ein Format zum Datenaustausch, welches auf JavaScript basiert. Sein Haupteinsatzgebiet findet sich bei Web-Applikationen und es wird häufig gemeinsam mit Ajax verwendet. Somit ist es möglich Informationen wie Arrays, Objekte oder sonstige Variablen zu speichern. Es ist komplett sprachenunabhängig, verwendet aber Konventionen wie Java, C, C++, C#, u.a.

Ein Vorteil von JSON ist die Lesbarkeit, und dass es leicht zu schreiben ist. Für Maschinen ist es einfacher zu parsen und zu generieren.

Objekt

Objekt ist eine Menge, die ungeordnet Name/Wert Paare speichert. Der Aufbau eines Objekts sieht folgendermaßen aus:

object

```
{ }
```

```
{ members }
```

Dem Namen folgt ein : (Doppelpunkt) und der Wert gefolgt vom Namen/Wert Paar getrennt werden diese durch , (Komma)

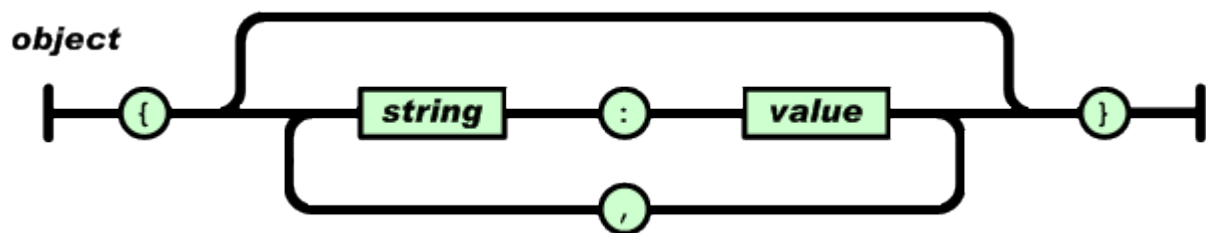


Abbildung 8: JSON Object

Array

Array ist eine Liste von Werten, welche geordnet werden. Der Aufbau eines Arrays sieht folgendermaßen aus:

array

`[]`

`[elements]`

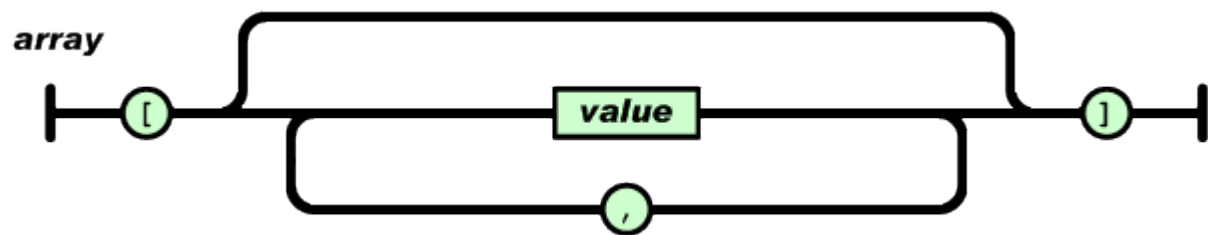


Abbildung 9: JSON Array

Value

Ein Wert kann aus folgenden Typen bestehen:

- value
- string
- number
- object
- array
- true
- false
- null

value

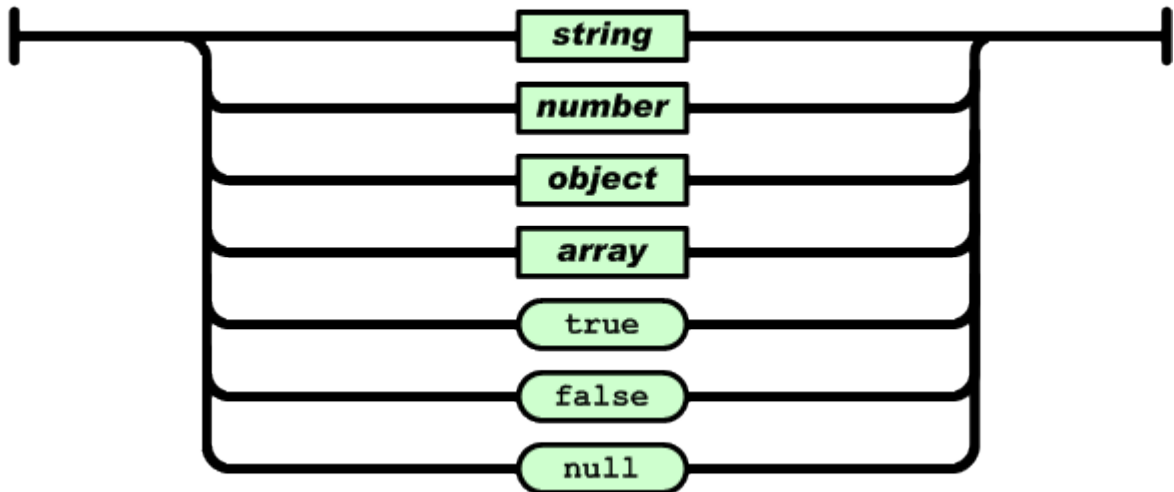


Abbildung 10: JSON Value

Diese können ineinander verschachtelt sein.

String

Ein *String* kann entweder aus keinem oder aus mehreren Unicode-Zeichen bestehen und wird mittels dopptelten Anführungszeichen umschlossen. Ein String in JSON ist dem in C oder Java sehr ähnlich.

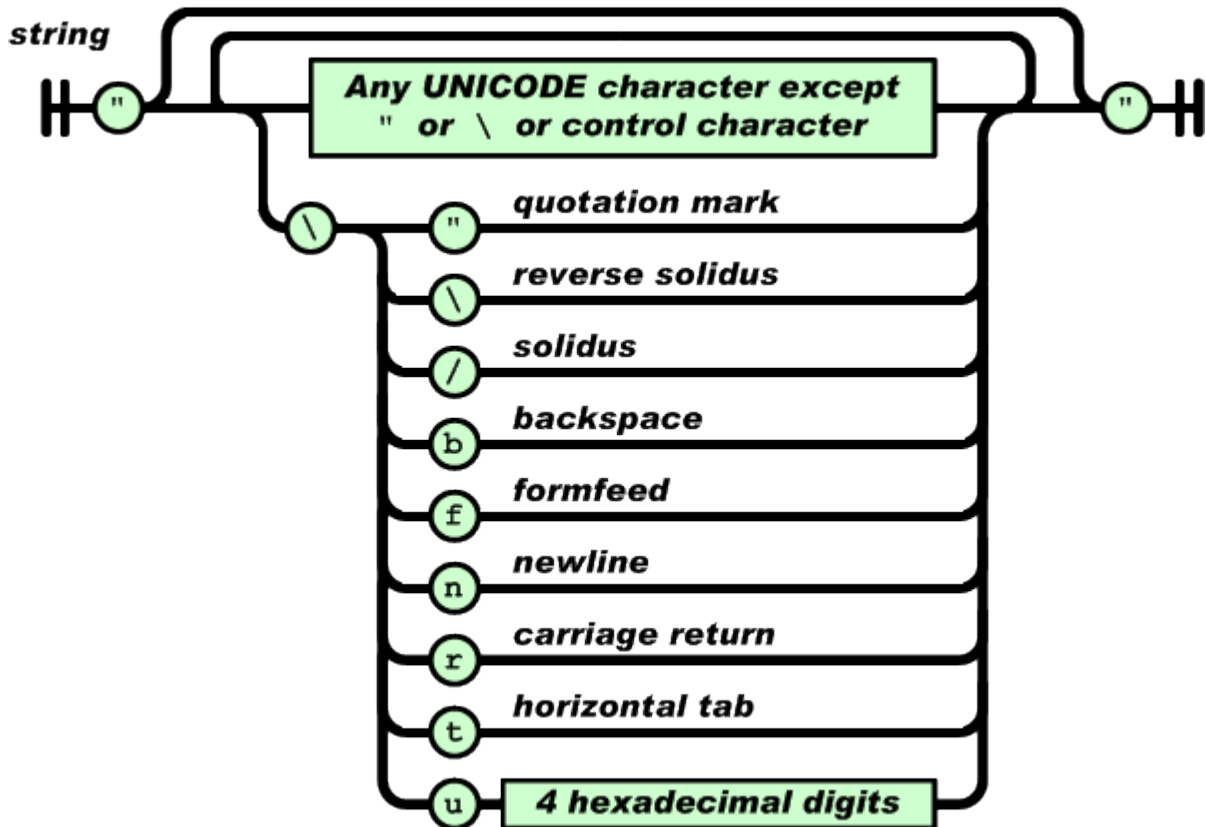


Abbildung 11: JSON String

Number

Oktale und hexadezimale Zahlen werden im Gegensatz zu Java oder C in JSON nicht verwendet, ansonsten ist es sehr ähnlich.

number

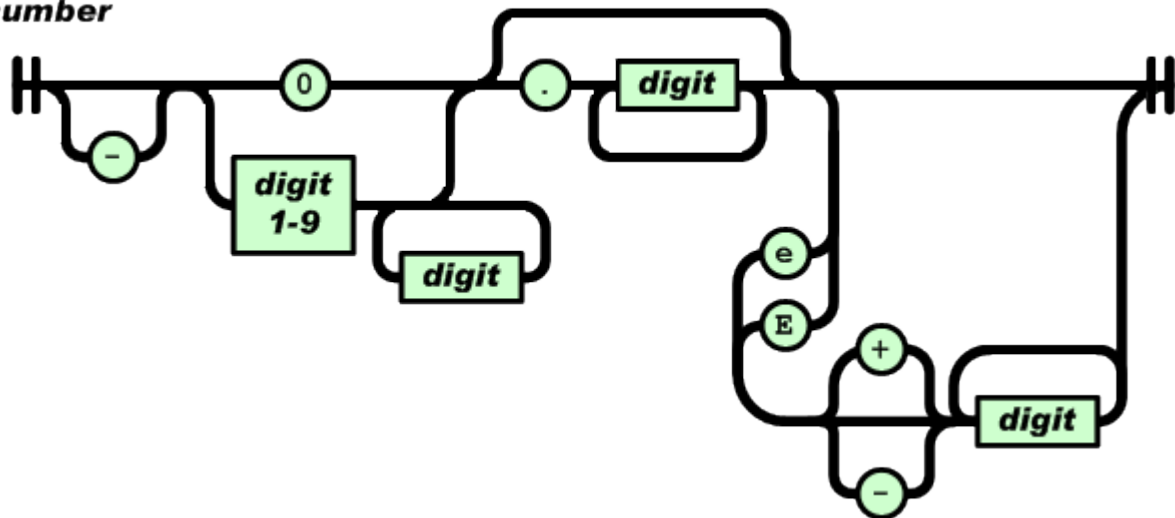


Abbildung 12: JSON Number

[Json, 2016]

5.1.10. SmartGit

SmartGit ist eine benutzerfreundliche Versionsverwaltung für Windows. Damit können Git Repositories verwaltet werden. [Planio, 2013] Ein Git Repository ist ein Versionsverwaltungssystem, welche Änderungen an einem Dokument zur Vorversion feststellt und diese Dokumente zusammenführt. Es dient zur Verbreitung neuer Versionen und zum Erstellen von Workflows. [Pilcrow, 2012]

6. Benutzung

6.1. Programmablauf

Der Benutzer erreicht über <http://diabeteshelfer.htl-perg.ac.at/> die Startseite unserer Web-Applikation. Es gibt die Möglichkeit sich anzumelden oder sich neu zu registrieren.

6.1.1. Startseite

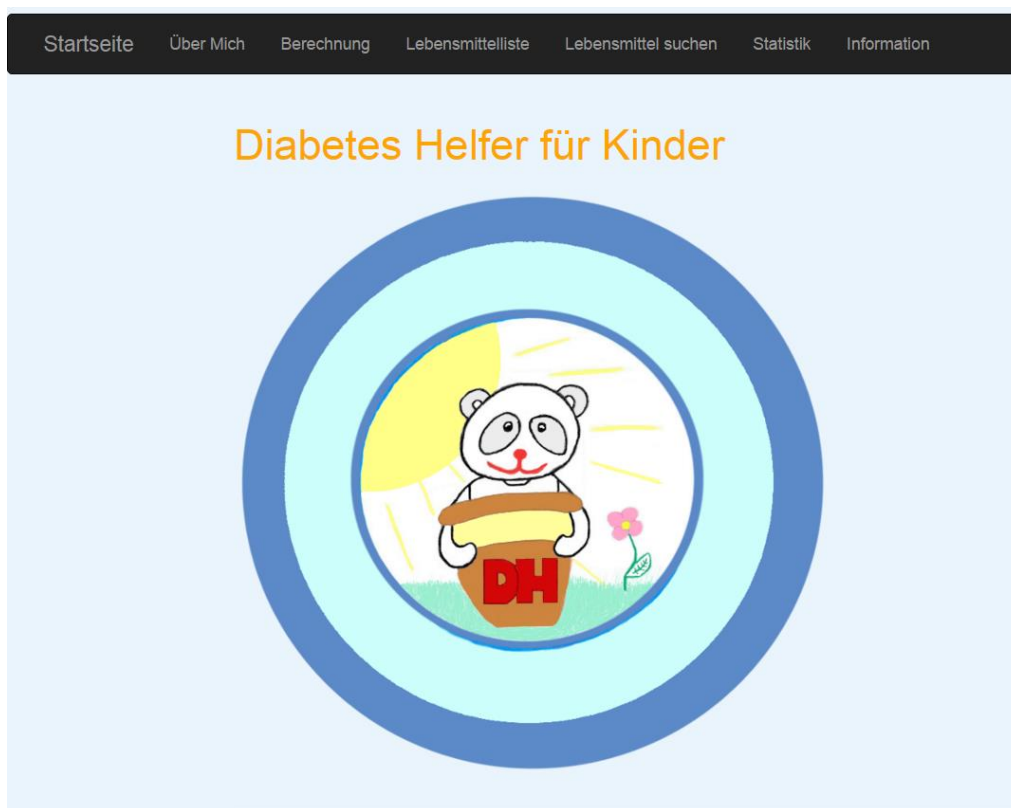


Abbildung 13: Startseite

6.1.2. Login

Wenn der Benutzer auf *Anmelden* klickt, wird er auf die Anmeldeseite verlinkt, wo ein Menü erscheint, welches es ihm ermöglicht, sich mit einem schon vorhandenen Konto anzumelden oder sich neu zu registrieren. Dafür gibt er seine E-Mail Adresse und sein Passwort ein und drückt auf anmelden. Falls der Benutzer noch keinen Account hat, gibt es die Möglichkeit einen Neuen zu erstellen. Wenn der Benutzer auf *Anmelden* klickt wird geprüft, ob die E-Mail Adresse und das dazugehörige Passwort in der Datenbank vorhanden sind. Falls dies der Fall ist, wird der Benutzer auf die Startseite weiterverlinkt und eine Session wird ihm zugeteilt um zu sehen, welcher User gerade eingeloggt ist.

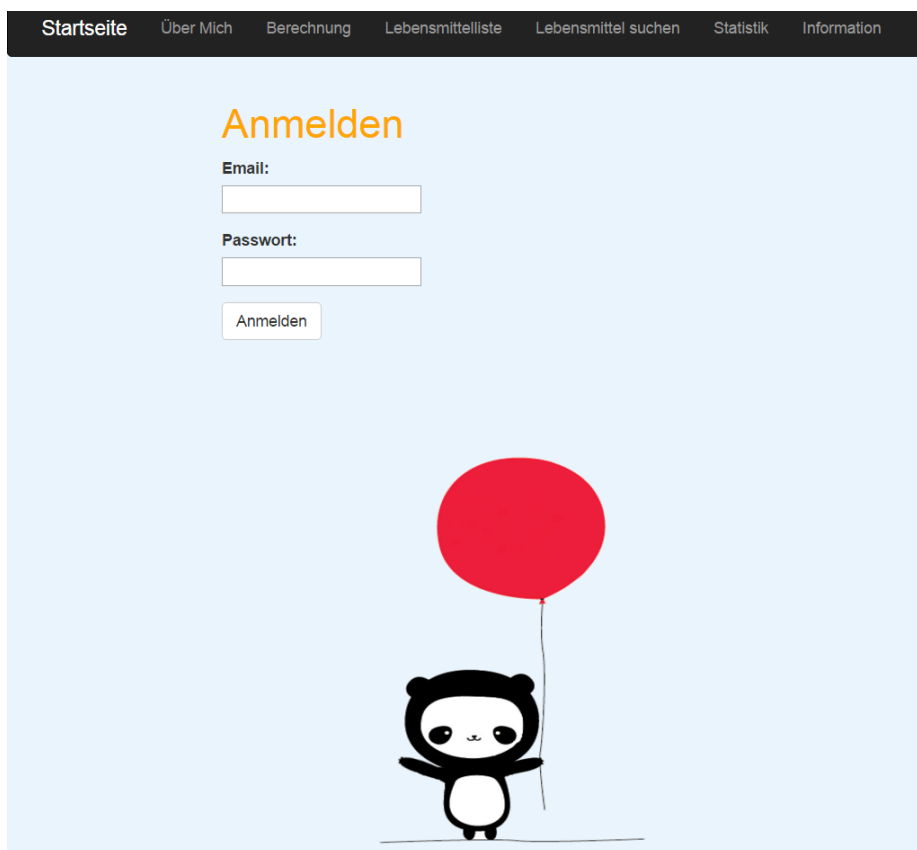


Abbildung 14: Login

Im Falle, dass der Benutzer seine Login Informationen vergessen haben sollte, kann er sich diese zuschicken lassen.

Es gibt auch die Möglichkeit, sich neu zu registrieren. Dafür sind zuerst nur die E-Mail-Adresse und ein Passwort notwendig. Die erforderlichen medizinischen Informationen werden erst später eingegeben. Um Fehler bei der Passwortwahl zu vermeiden, muss das Passwort doppelt eingegeben werden. Wenn auf *Registrieren* geklickt wird, wird der Benutzer auf eine Seite verlinkt, wo er sich registrieren kann. Wenn er mit seinen Eingaben zufrieden ist und diese bestätigt, werden diese kontrolliert, ob sie den Kennwortrichtlinien entsprechen. Falls dies der Fall ist, werden die Daten an den Server gesendet und dort abgespeichert um sie in Zukunft verwenden zu können. Wenn dies nicht der Fall ist, wird der Benutzer aufgefordert seine Eingaben zu überarbeiten und neu zu bestätigen. Nachdem die Eingaben korrekt eingegeben wurden, wird der Benutzer auf die Startseite verlinkt, wo er sich mit seinem neu erstellten User anmelden kann.

6.1.3. Über mich

Nach dem Erstellen des Benutzers werden im Reiter *Über mich* die Informationen der Person eingegeben, welche für die Berechnung notwendig sind. Diese Informationen dienen als Basis für die folgenden Schritte. Dabei ist es wichtig festzustellen, ob es sich beim Insulingerät um einen Pen oder einer Pumpe handelt. Ebenfalls der Umrechnungsfaktor, welcher im Krankenhaus ermittelt wird, muss korrekt eingegeben werden. Die Daten werden danach an den Server gesendet, wo sie zur jeweiligen Person zugeordnet werden.

Startseite Über Mich Berechnung Lebensmittelliste Lebensmittel suchen Statistik Information

Über Mich

Vorname:

Nachname:

Geschlecht: weiblich männlich

Geburtstag:

Größe(cm):

Gewicht:

Insulingerät: Pen Pumpe

Umrechnungsfaktor:

[bearbeiten](#)




Abbildung 15: Über Mich

6.1.4. Berechnung

Um die Insulinwerte zu berechnen wird eine Seite dargestellt, wo der Benutzer das Datum und die derzeitige Uhrzeit eingibt, zudem wird der derzeitige Blutzuckerwert eingegeben, welcher zuvor gemessen wurde. Nachdem dies erledigt ist, wählt der Benutzer aus der Lebensmittelliste das Gericht aus, welches er verzehren wird. Schlussendlich ist nur mehr auf den Button *Berechnen* zu drücken und die berechneten Insulineinheiten werden angezeigt. Die Daten aus der Eingabe werden an den Server gesendet. Diese Daten sind notwendig zur Darstellung der Statistik. Der berechnete Insulinwert ist notwendig, um zu sehen wieviel Insulin zu spritzen ist, damit das Nahrungsmittel problemlos zu verspeisen ist.

Startseite Über Mich Berechnung Lebensmittelliste Lebensmittel suchen Statistik Information

Berechnung der Insulinwerte

Datum:

Uhrzeit:

Blutzuckerwert:

Nahrungsmittel: [auswählen](#)
Kartoffelknödel

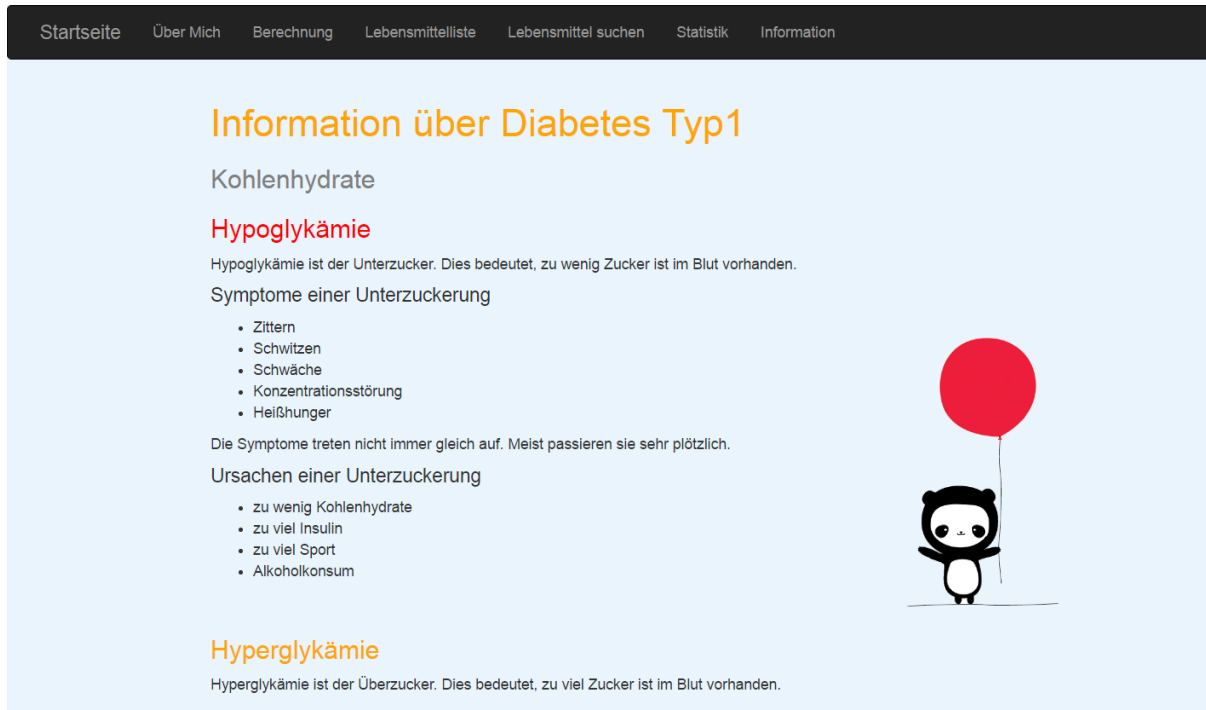
berechnete Insulineinheit

© 2016 - Diabetes Helfer für Kinder

Abbildung 16: Berechnung der Insulinwerte

6.1.5. Informationen

Zusätzlich kann sich der Benutzer über Diabetes Typ1 informieren, indem er auf den Tab *Information* klickt. Bei den Informationen handelt es sich um eine allgemeine Auskunft und Verlinkungen auf externe Seiten, welche ausführlichere Erklärungen über das Themengebiet liefern.



Startseite Über Mich Berechnung Lebensmittelliste Lebensmittel suchen Statistik Information

Information über Diabetes Typ1

Kohlenhydrate

Hypoglykämie

Hypoglykämie ist der Unterzucker. Dies bedeutet, zu wenig Zucker ist im Blut vorhanden.

Symptome einer Unterzuckerung

- Zittern
- Schwitzen
- Schwäche
- Konzentrationsstörung
- Heißhunger

Die Symptome treten nicht immer gleich auf. Meist passieren sie sehr plötzlich.

Ursachen einer Unterzuckerung

- zu wenig Kohlenhydrate
- zu viel Insulin
- zu viel Sport
- Alkoholkonsum

Hyperglykämie

Hyperglykämie ist der Überzucker. Dies bedeutet, zu viel Zucker ist im Blut vorhanden.

Abbildung 17: Information

6.1.6. Lebensmittelliste

Im Reiter *Lebensmittelliste* hat der Benutzer die Möglichkeit, die bereits gespeicherten Lebensmittel einzusehen oder neue hinzuzufügen. Die Lebensmittel werden in Kategorien eingeteilt und jeder Eintrag hat ein entsprechendes Bild, um es benutzerfreundlicher zu gestalten. Bei Klick auf einen Lebensmittelnamen oder auf eine Kategorie erscheinen die Lebensmittel, welche dort gespeichert sind.

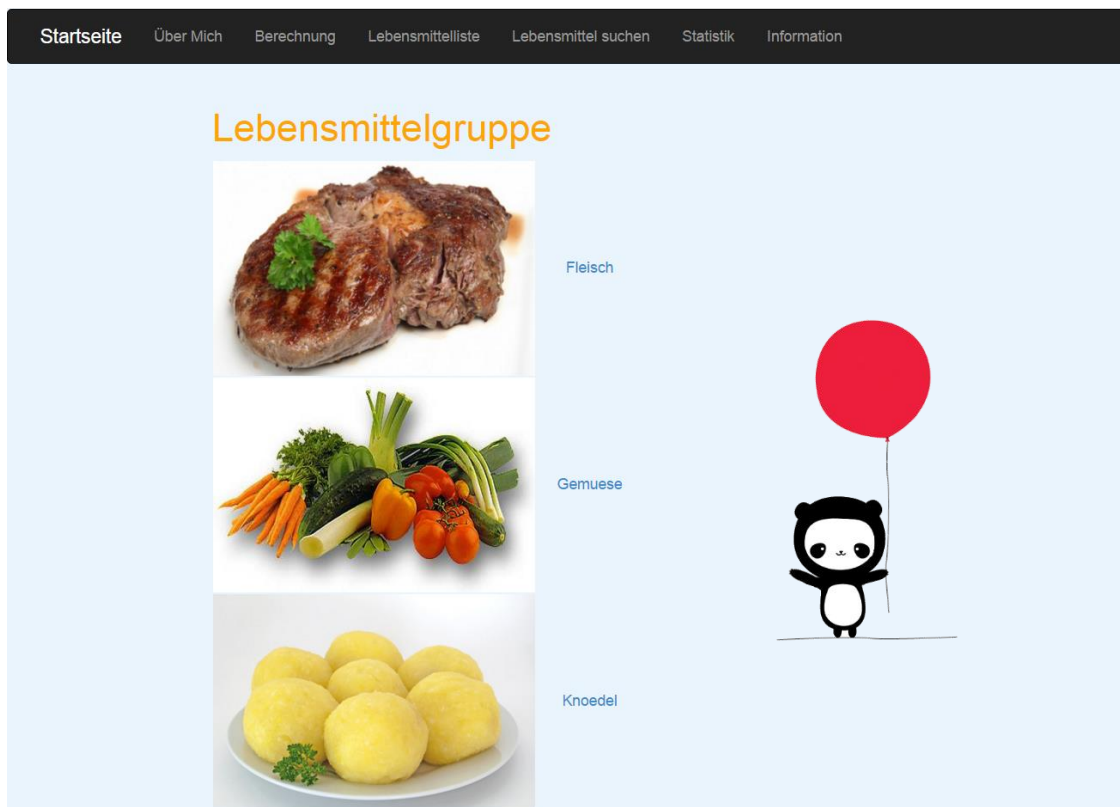


Abbildung 18: Lebensmittelliste Gruppe



Abbildung 19: Lebensmittelliste ‚Gemüse‘

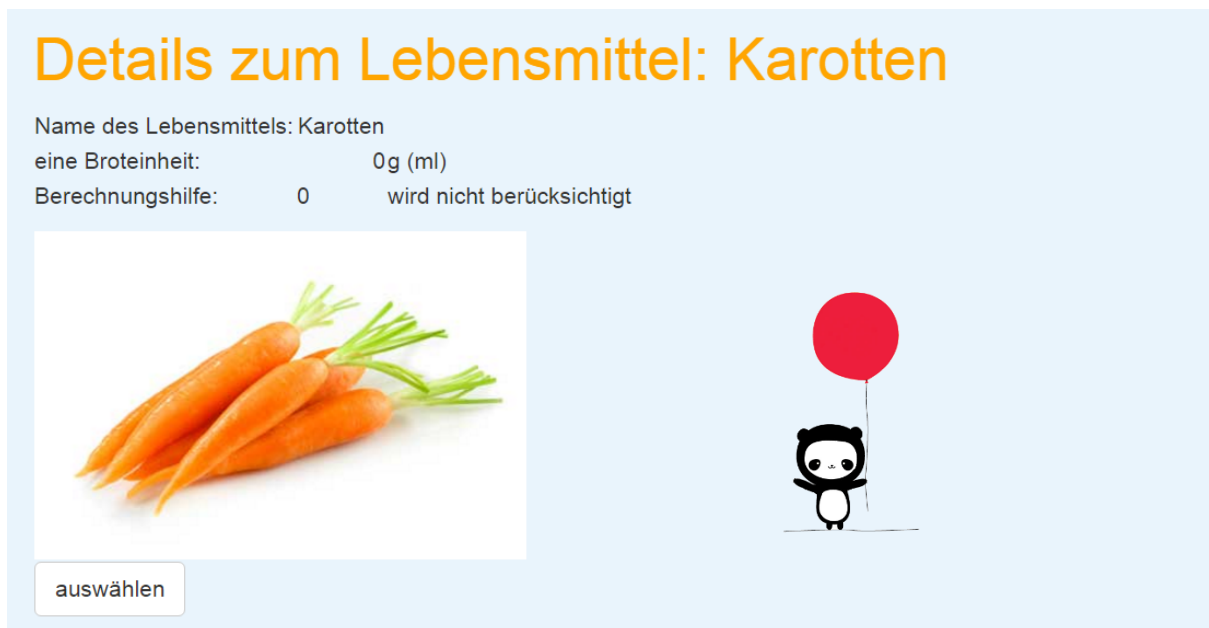


Abbildung 20: Lebensmittel Detail

6.1.7. Statistik

Im Reiter *Statistik* ist es möglich die Insulinwerte und Blutzuckerwerte der letzten Zeit einzusehen. Dies kann für einen gewissen Tag oder über eine gewisse Dauer geschehen und wird in einer Grafik angezeigt. Somit ist der Vergleich schnell und einfach zu ersehen. Die Auswahl des Tages erfolgt mittels Drop-Down Menü.

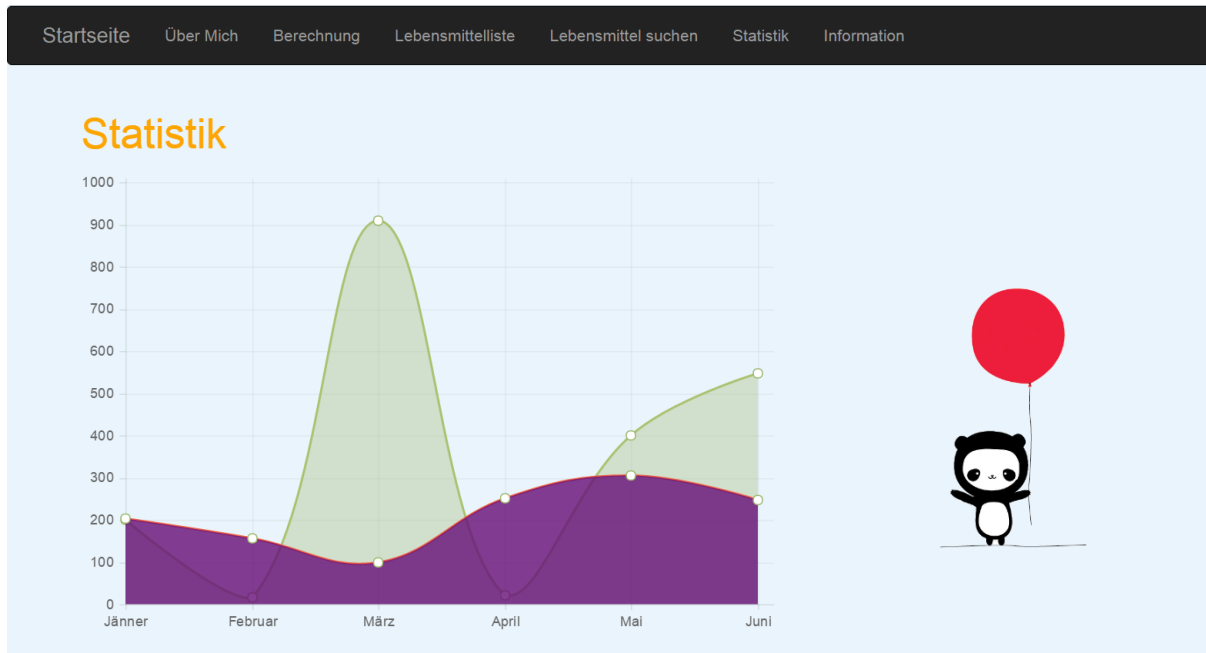


Abbildung 21: Statistik

6.2. Use – Case Diagramm

Ein Use-Case Diagramm ist ein Verhaltensdiagramm und zeigt einen Anwendungsfall. Dafür werden Akteure, Programmschritte und Systemgrenzen gezeigt. Das Diagramm wird immer aus der Sicht des Benutzers angezeigt.

VERWENDBARE ELEMENTE:

- **Akteur:**
Unter Akteur wird eine Person oder ein Prozess bezeichnet.
- **Anwendungsfall:**
Ein Anwendungsfall ist ein Vorgang, welcher von einem Akteur ausgeführt wird. Dabei kann auch unter zwei Varianten unterschieden werden:
 - **include:**
Ein Use-Case ist mit einem anderen Use-Case verbunden.
 - **extend:**
Ein Use-Case kann mit einem anderen Use-Case verbunden werden.
- **Rahmen:**
Als Rahmen wird eine Systemgrenze bezeichnet.
- **Äußerer Rahmen:**
Der äußere Rahmen allerdings begrenzt und bezeichnet das gesamte Use-Case Diagramm.

[SCHWAB, Felix, Schwab-Matkovits, Ingrid, 2013]

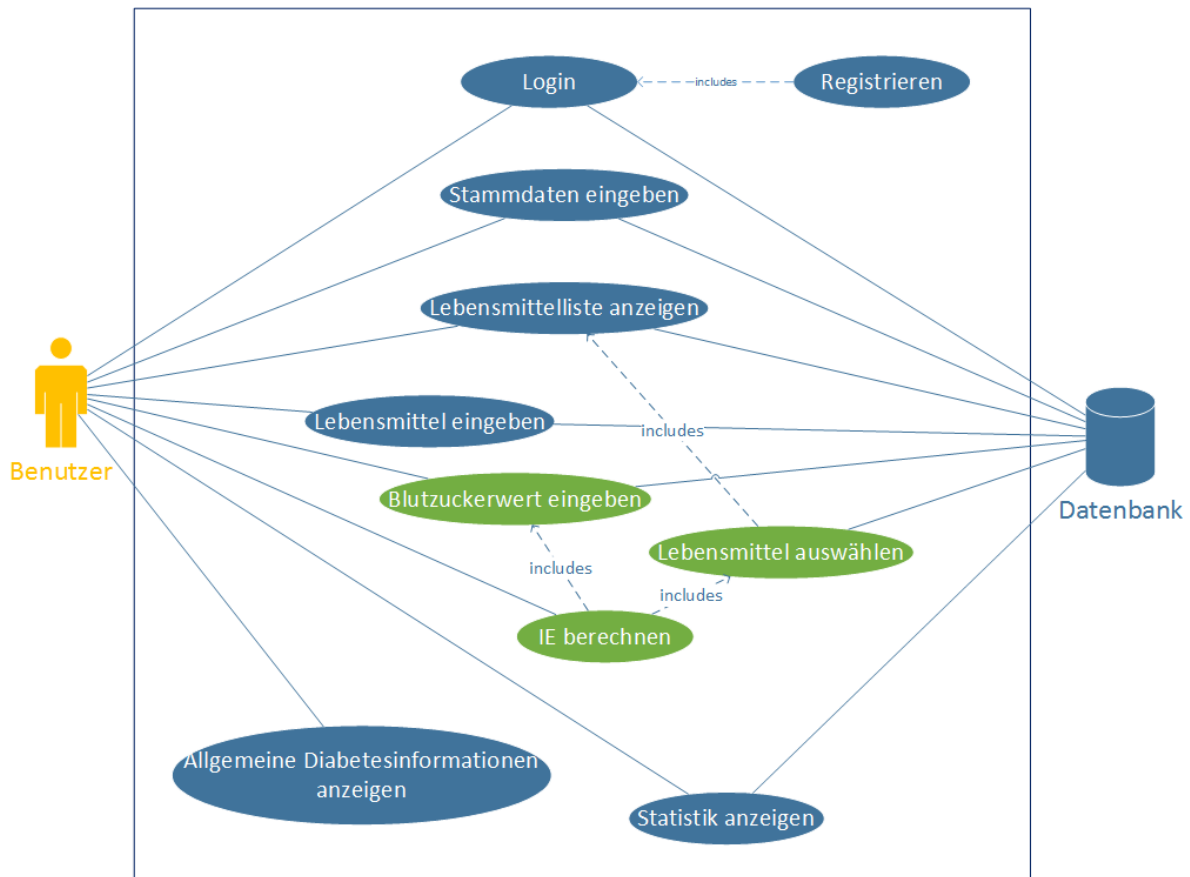


Abbildung 22: Use-Case Diagramm

LOGIN & REGISTRIEREN

Wie das Use-Case Diagramm zeigt, muss sich ein Benutzer zuerst einloggen. Die notwendigen Daten dafür, werden aus einer Datenbank geladen und überprüft. Ist ein Benutzer noch nicht registriert, muss er seine E-Mail-Adresse und ein Passwort eingeben, welche anschließend in der Datenbank festgehalten werden.

STAMMDATEN EINGEBEN

Das obig dargestellte Diagramm, zeigt außerdem, dass der Benutzer die Möglichkeit hat, seine Stammdaten einzugeben, welche anschließend auf der Datenbank gespeichert werden. Diese Daten beinhalten den Vor- und Nachnamen des Benutzers, das Geschlecht und das Geburtsdatum. Außerdem werden auch die Größe in Zentimeter und das Gewicht in Kilogramm gespeichert. Zusätzlich kann der Benutzer noch angeben, ob er eine Pumpe oder einen Pen besitzt. Als Letztes wird noch der Umrechnungsfaktor gespeichert.

Diese Daten können jederzeit vom Benutzer geändert und aktualisiert werden.

LEBENSMITTEL EINGEBEN

Im Use-Case Diagramm ist ersichtlich, dass der Benutzer neue Lebensmittel, welche noch nicht in der Datenbank vorhanden sind, eingeben kann. Hierfür werden der Name und die Kohlenhydrate des Produkts eingetragen. Die Kohlenhydrate werden in Gramm angegeben und diese entsprechen genau eine Broteinheit des Lebensmittels. Wenn eine Berechnungshilfe für dieses Produkt vorhanden ist, wird diese auch mitgespeichert. Vor dem Speichern, lädt der Benutzer noch ein Bild dieses Lebensmittels hoch.

LEBENSMITTELLISTE ANZEIGEN

Der Benutzer kann sich eine Lebensmittelliste anzeigen lassen. In dieser sind Informationen, wie die Kohlenhydrate der Lebensmittel, vorhanden. Die Lebensmittel sind in Gruppen unterteilt. Die Lebensmittelliste ist von zentraler Bedeutung für die Berechnung der Insulineinheiten. Der Benutzer braucht diese um Lebensmittel auswählen zu können.

BLUTZUCKERWERT EINGEBEN

Die Funktion *Blutzuckerwert eingeben* ist im Use-Case Diagramm färbig hervorgehoben, da dies für die Berechnung der IE, eine Hauptaufgabe des Programmes, benötigt wird. Der Benutzer misst mit einem für ihn geeigneten Blutzuckermessgerät seinen aktuellen Blutzuckerwert und gibt diesen anschließend ins Programm ein. Dies ist dahingehend für die Berechnung wichtig, da der Blutzuckerwert berücksichtigt werden muss, wenn dieser zu niedrig oder zu hoch ist.

LEBENSMITTEL AUSWÄHLEN

Diese Funktion ist für die Berechnung von sehr hoher Wichtigkeit. Aus diesem Grund ist sie im oben dargestellten Diagramm auch färbig hervorgehoben, da sie für die Berechnung der Insulineinheiten benötigt wird. Ein Benutzer wählt die Lebensmittel aus der Liste aus, welche er verzehren wird. Dies wird für die Berechnung benötigt, da aus den Broteinheiten dieser Lebensmittel die Insulineinheiten berechnet werden.

IE BERECHNEN

Die Berechnung der Insulineinheiten ist das Herzstück unseres Programmes, daher ist es auch im Use-Case Diagramm farblich hervorgehoben. Für die Berechnung werden der aktuelle Blutzuckerwert und die zu verzehrenden Lebensmittel berücksichtigt.

STATISTIK ANZEIGEN

Wie im oben dargestellten Diagramm ersichtlich ist, hat der Benutzer außerdem die Möglichkeit, sich eine Statistik anzeigen zu lassen. Der Benutzer wählt einen Tag aus und das Programm zeigt für diesen Tag die eingegebenen Insulineinheiten und die Blutzuckerwerte zu einer bestimmten Uhrzeit in Diagrammform an.

ALLGEMEINE DIABETESINFORMATION ANZEIGEN

Das Programm bietet die Möglichkeit, sich über Diabetes im Allgemeinen zu informieren. Dabei wird im näheren auf Kohlenhydrate, Hyperglykämie und Hypoglykämie eingegangen. Möchte man weitere Informationen bezüglich dieser Krankheit erhalten, bietet die Webseite Links zu weiteren Informationsquellen an.

6.3. Ergebnis

Das Ergebnis unserer Arbeit ist, dass die Berechnung der Insulineinheiten vereinfacht wird. Außerdem werden der Blutzuckerwert und die Insulineinheiten visuell dargestellt. Dadurch kann man sehen, wie sich die Ernährung auf die Gesundheit auswirkt. Zusätzlich wird die Korrektheit der Berechnungen garantiert, da keine Rechenfehler mehr passieren können, welche zuvor bei der händischen Art vorgekommen sind. Überdies hinaus ist das Programm kinderfreundlich und übersichtlich gestaltet, da es sich hauptsächlich an junge Diabetiker und deren Eltern richtet. In der Lebensmittelliste ist es möglich alle relevanten Informationen der Lebensmittel herauszulesen ohne sie vom jeweiligen Produkt abzulesen. Auch ist es möglich neue Lebensmittel hinzuzufügen, falls diese noch nicht in der Liste angeführt sind. Um sich über die Thematik Diabetes zu informieren, stehen verschiedene Internetlinks und Informationen zur Verfügung.

7. Realisierung

7.1. Projektstrukturplan

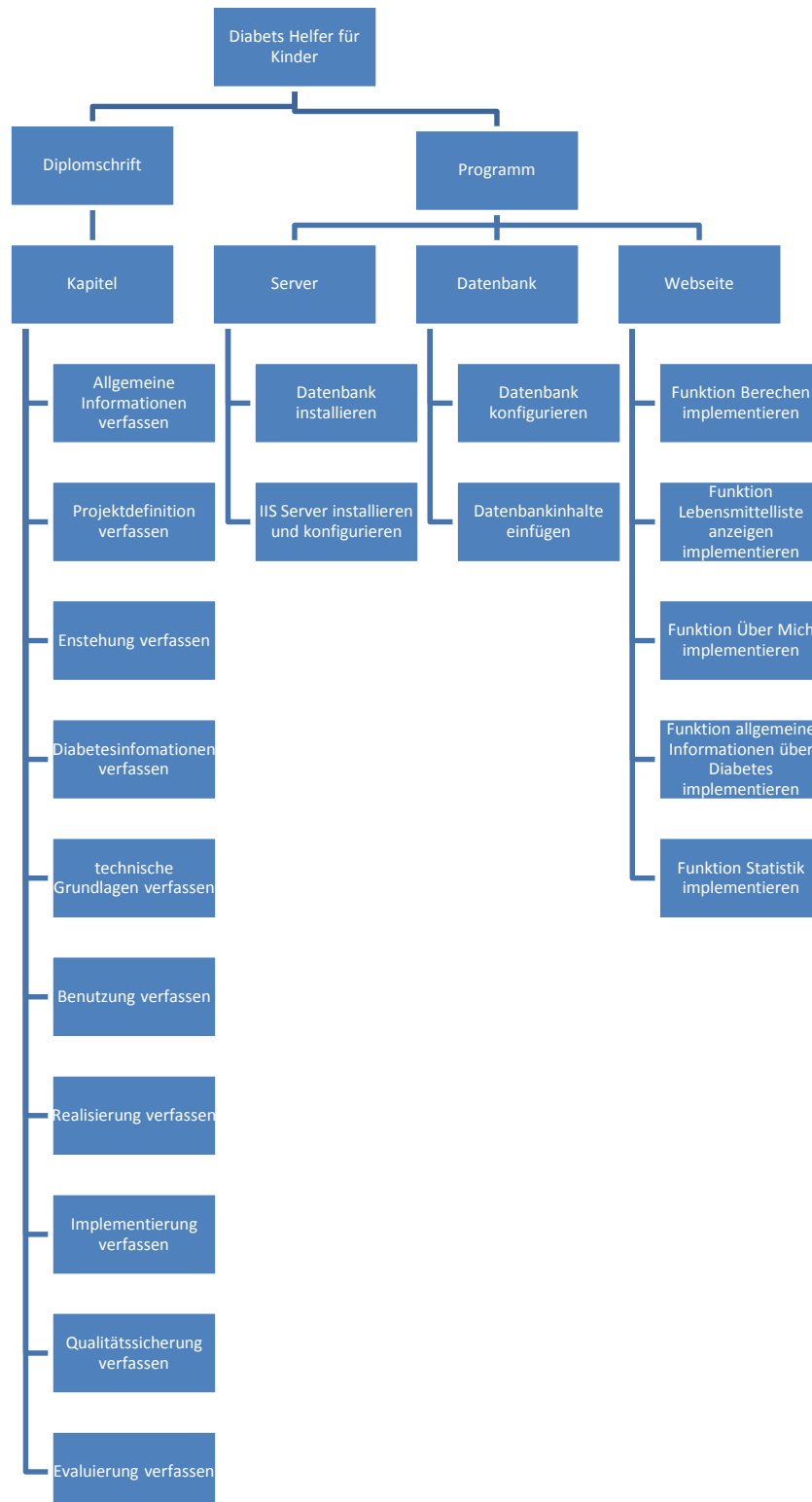


Abbildung 23: Projektstrukturplan

7.2. Aufwandsschätzung

Die Aufwandsschätzung beträgt für alle Diplomarbeitmitglieder insgesamt:

Task	Stunden
Einführung in die Thematik Diabetes	14
Installation und Konfiguration vom externen Server	6
Installation und Konfiguration von MySQL Datenbank auf dem externen Server	6
Datenmodell erstellen und Datenbanksätze einfügen	30
Entwicklungsumgebung einrichten	9
Funktion „Login“ implementieren	40
Funktion „Berechnen“ implementieren	90
Funktion „Lebensmittelliste anzeigen“ implementieren	80
Funktion „Statistik“ implementieren	80
Funktion „Information“ implementieren	6
Funktion „Über mich“ implementieren	9
alle Funktionen testen	15
Dokumentation / Diplomschrift	120
Summe	515

7.3. Schnittstellen und Datenfluss (P)

Um die Diplomarbeit *Diabetes Helfer für Kinder* realisieren zu können, sind folgende Schnittstellen zu beachten:

7.3.1. MySql Connector

SCHNITTSTELLE ZWISCHEN DATENBANK UND PROGRAMM

Das Programm wurde mit dem Programm *Visual Studio* programmiert. Damit das Programm auf die gespeicherten Daten der Datenbank zugreifen kann, wird ein Connector benötigt. Dieser wird kostenlos im Internet zur Verfügung gestellt. [MySql Connector, 2016]

Zu beachten ist, dass dieser Connector vor dem Datenverarbeitungssystem *MySql* installiert werden muss. Ist bereits *MySql* vorinstalliert, ist es notwendig diesen zu deinstallieren und den Connector als Erster zu installieren.

Beim Connector ist zu beachten, dass er auf die Entwicklungsumgebung angepasst ist, dies bedeutet, dass für ein Java Projekt ein dementsprechender Connector und für *.NET* ein anderer Connector installiert werden muss.

Installation

Um den Connector herunterzuladen zu können, wird eine aktive Internetverbindung benötigt. Ebenfalls muss die Internetverbindung beim Installieren aktiv sein.

Nach erfolgreicher Installation ist erkennbar, dass in *Visual Studio* beim Hinzufügen einer neuen Datenbankverbindung der *MySql Driver* vorhanden ist. Wenn dieser Driver nicht sichtbar ist, ist ein Neustart des Computers notwendig.

7.3.2. MySql

SCHNITTSTELLE ZWISCHEN SERVER UND DATENBANK

Damit die eingegebenen Daten dauerhaft gespeichert werden, wird das kostenlose Datenverarbeitungssystem *MySql* verwendet. [Wikipedia MySql, 2016] Dieses System, welches direkt auf den Server installiert werden muss, beinhaltet den *MySql Server* und seine Dienste. *MySql* wird verwendet, weil es ein Open Source Produkt ist und es auf verschiedenen Betriebssystemen lauffähig ist.

Installation

Bei der Installation für diese Diplomarbeit ist eine Standard Konfiguration ausreichend. Um *MySql* installieren zu können, ist eine aktive Internetverbindung erforderlich. Nach erfolgreicher Installation wird der *MySql Instance Configuration Wizard* aufgerufen. Dieser Dienst ermöglicht die Einrichtung der Datenbank mithilfe einer grafischen Oberfläche. Weiters muss angegeben werden, welcher *MySql* Typ installiert wird, das heißt, ob *MySql* als Server oder als Klient verwendet wird. Auf dem Server werden die Daten gespeichert und der Klient schickt Anfragen an den Server. Damit eine vorgenommene Einstellung ausgeführt wird, muss der Button *Execute* gedrückt werden. [Netzwelt MySql, 2011]

7.3.3. MySql Workbench

DATENEINGABE ÜBER MYSQL WORKBENCH

Da die Daten in einem Datenverarbeitungssystem gespeichert werden, stellt sich die Frage, wie man diese Daten speichern kann. Dafür gibt es die Sprache *SQL* (Structured Query Language). *SQL* ist eine Datenbanksprache für Datenbankstrukturdefinitionen. Diese Sprache dient ebenfalls für Abfragen der Datenbankinhalten, das bedeutet zum Einfügen, Bearbeiten und Löschen von einzelnen Datenbankinhalten. [Wikipedia Sql, 2016]

Damit nicht jeder diese Sprache ins kleinste Detail beherrschen muss, stellt *MySql* ein Programm mit dem Namen *MySql Workbench* zur Verfügung. Dieses Programm ermöglicht das Erstellen einer neuen Datenbank und das Anlegen von Tabellen in dieser Datenbank. Dafür müssen keine selbsterstellten *SQL Kommandos* zum Server geschickt werden, sondern diese werden mithilfe einer grafischen Oberfläche zusammengestellt und an den Server geschickt.

Damit danach Datensätze eingefügt werden können, muss man nur mehr die einzelnen Werte in den Tabellen eingetragen. Sobald diese zum Server geschickt werden, sind diese Daten gespeichert.

7.3.4. IIS Server Konfiguration

SCHNITTSTELLE ZWISCHEN PROGRAMM UND SERVER

Diese Diplomarbeit ist als Web Applikation im Internet aufrufbar, daher ist es notwendig einen Server zu installieren und das Programm auf diesen Server auszuführen. In diesem Fall muss der Server Windows als Betriebssystem installiert haben.

Damit auf dem lauffähigen Server das Programm aus dem Internet aufrufbar ist, ist es notwendig einen IIS Server zu installieren und zu konfigurieren.

Warum IIS Server

Ein IIS Server ist eine Plattform, welche leicht zu verwalten, sicher und erweiterbar ist. Es eignet sich besonders für das Hosting von Webseiten, Dienste und Anwendungen. Weiters ist es eine einheitliche Plattform, welche IIS, Asp.Net, FTP – Dienste und PHP einschließen.

VORTEILE:

- Websicherheit durch reduzierten Speicherbedarf
- einfaches Ausführen von ASP.NET Programmen
- einfaches Hinzufügen, Bearbeiten, Löschen von IIS – Komponenten
- dynamische Zwischenspeicher

[Tech Microsoft, 2016]

Installation & Konfiguration IIS Server

Nachdem der Server konfiguriert wurde, wird ein neuer Web Server installiert. Dieser enthält zusätzlich zu seinen Voreinstellungen einige Eigenschaften, welche ergänzt werden müssen. Nach erfolgreicher Installation ist der IIS Server in der Management Konsole sichtbar. Ab diesem Zeitpunkt gibt es auf der Festplatte des Servers ein Verzeichnis, welches *inetpub* heißt. Darunter gibt es wieder ein Verzeichnis mit dem Namen *wwwroot*. In dieses Verzeichnis muss das komplette Programm hineinkopiert werden. Nun muss das Programm als Startpage auf den IIS Server installiert werden. Dafür muss in den *Internet Information Services* des *IIS Servers* eine neue Webseite unter Site erstellt werden. Wichtig dabei ist, dass als *Path* das Verzeichnis des Programms in *wwwroot* ausgewählt wird und als *Hostname* die URL (Uniform Resource Locator) angegeben wird, unter welcher das Programm im Internet aufgerufen wird.[Youtube IIS Server, 2012]

8. Implementierung

Bei dieser Diplomarbeit ist bei der Erstellung der einzelnen Webseiten HTML5 und CSS 3 verwendet worden. Diese Version von HTML passt sich an das Gerät an, auf welchem die Webseite aufgerufen wird.

In den einzelnen Funktionen wird jeweils der Implementiervorgang beschrieben. Zusammen ergeben sie die gesamte Webseite, welche aufgerufen werden kann.

8.1. Model-View-Controller

Die Realisierung unserer Web-Applikation erfolgt nach dem *Model-View-Controller-Prinzip* (MVC). Dabei gibt es folgende drei Komponenten:

- Model
- View
- Controller

Im Model werden die Daten, welche in der Applikation dargestellt und gebraucht werden, vorbereitet. In der View werden die Daten aus dem Model dargestellt und im Controller werden die Funktionalitäten bereitgestellt und ausprogrammiert.

Zum Beispiel an der Funktion der *Berechnung*. Im Model *CalculationModel* werden die Daten, welche für die Berechnung benötigt werden, zwischengespeichert. Im Controller *CalculationController* gibt es eine Methode für die Berechnung und auch für die Überprüfung des Blutzuckerwertes. In der View *Calculation*, werden Daten, wie Uhrzeit, Blutzuckerwert und berechneter Insulinwert, dargestellt. Außerdem gibt es in der View die *TextBoxen*, in welchen es möglich ist, das Datum, Uhrzeit und den Blutzuckerwert einzugeben. Zusätzlich gibt es in der View auch einen *ActionLink*, welcher auf die View der Lebensmittelliste verlinkt. Durch diesen Link ist es möglich, die zu verzehrenden Lebensmittel auszuwählen.

Obwohl mit der Vorgehensweise des Model-View-Controllers mehr Aufwand verbunden ist, überwiegen dennoch folgende Vorteile:

- guter Aufbau der Struktur, durch die strikte Trennung der Komponenten
- somit bessere Übersicht des Programmcodes
- Änderungen können leichter vorgenommen werden
- Fehler können leichter gefunden werden, da jede Komponente eigenständig arbeitet
- in einem Model kann es mehrere View geben

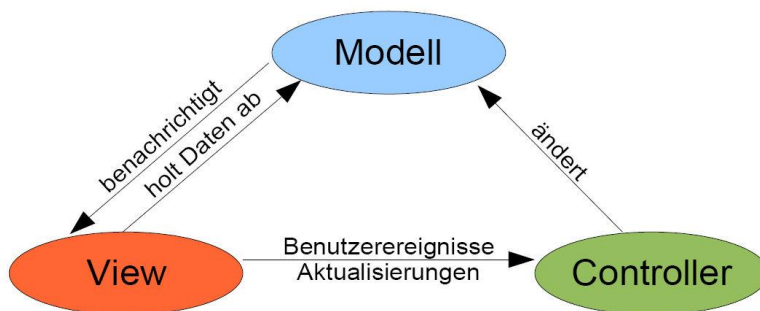


Abbildung 24: Model-View-Controller

In der Abbildung 24: Model-View-Controller ist nochmal veranschaulicht, wie das Model-View-Controller-Prinzip arbeitet. Die View holt sich Daten aus dem Modell. Der Controller ändert diese Daten und sendet sie zurück an das Modell und die View.

8.2. Allgemein

Damit bei der Web-Applikation Daten aus der Datenbank am Server bzw. auf die Datenbank geschrieben werden kann, muss eine Datenbankverbindung eingerichtet werden. Hierfür muss in der *MySQL – Workbench* am Server eine Änderung vorgenommen werden. Die Änderung befindet sich in der Einstellung der Zugriffe auf die Datenbank. Der User muss ein % als *Wildcard* definiert bekommen. Mit diesem Zeichen darf jeder Rechner auf diese Datenbank zugreifen, wenn es mit dem Passwort übereinstimmt.

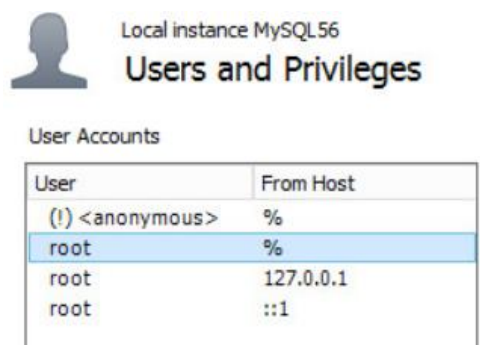


Abbildung 25: MySQL Users and Privileges

Jetzt kann aus der Entwicklungsumgebung eine Verbindung zur Datenbank hergestellt werden. Dafür wird eine neue Verbindung am *Server Explorer* erstellt. Als Data Source wird hierbei ein *MySQL* Treiber ausgewählt. Das Serverfeld beinhaltet die IP – Adresse des Servers. Der Benutzername ist der User von unserer Datenbank und das Feld *Passwort* enthält das Passwort von der Datenbank. Bei Databasename wird jene Datenbank ausgewählt, welche mittels dem Datenbankmodell, siehe 8.4 dieser Arbeit, erstellt wurde.

Damit Daten ausgelesen werden können, müssen die Tabellen als Objekt dargestellt werden. Da wir bereits im Vorhinein die Datenbank erstellt haben und Daten eingefügt wurden, haben wie die *Database first - Variante* beim Erstellen der Applikation gewählt. Hierbei erstellt die Entwicklungsumgebung für uns die Objekte. Die Namen der Objekte entsprechen exakt den Tabellennamen in der Datenbank.

Hierbei fügen wir lediglich in dem Projekt, welches für die Objekte erstellt wurde, ein Item Namens *ADO.NET Entity Data Model* hinzu. Es wird die Datenbankverbindung, welche zuvor eingerichtet wurde, angegeben. Jetzt werden die Objekte automatisch erstellt.

Es wird außerdem der Connection String zur Datenbank in der *App.conf* Datei eingetragen.

```
<connectionStrings>  
  <add name="DiabetesHelferFuerKinderModel" connectionString="server=10.114.57.40;user id=root;password=Celina2016!;persistsecurityinfo=True;data  
</connectionStrings><system.data>
```

Abbildung 26: connectionString

Nun sind alle Vorbereitungen getroffen und die Funktionen können erstellt werden.

8.3. Funktionen

8.3.1. Login

DEFINITION

Wird im Browser die Applikation aufgerufen, erscheint eine Startseite. Durch Klicken auf *Anmelden*, erscheint die Anmeldeseite. Auf dieser Seite werden der Benutzername, welcher eine E-Mail Adresse ist, und das zugehörige Passwort eingetragen. Durch Klicken auf den Button *Anmelden* erscheint die persönliche Webseite.

Sind Zugangsdaten noch nicht vorhanden, befindet sich der *Registrieren* Button neben dem *Anmelden* Button.

IMPLEMENTIERUNG

Die Eingabe der E-Mail Adresse und dem Passwort erfolgen jeweils über das HTML-Tag *TextBox*. Diese Eingabe wird im dazugehörigen Model *LoginModel* zwischengespeichert.

Im *LoginController* haben wir eine Methode, welche mit Hilfe von *Repositories*, die Daten im *LoginModel* mit denen aus der Datenbank prüft und abfragt. Stimmt die eingegebene E-Mail Adresse und das Passwort mit den Daten aus der Datenbank überein, wird eine Session gesetzt und der Anwender ist erfolgreich eingeloggt.

```
var user = repOperatorAccess.Get(u => u.Username == lm.Username && u.Password == lm.Password).FirstOrDefault();
if (user != null )
{
    FormsAuthentication.SetAuthCookie(lm.Username, true);
    Session["user"] = user;
    return RedirectToAction("Index", "Home");
}
```

Abbildung 27: Anmeldedaten überprüfen und Setzung der Session

Stimmen diese Daten nicht überein, wird eine entsprechende Fehlermeldung zurückgegeben.

Klickt ein Anwender auf den Button *Registrieren*, muss er seine E-Mail Adresse eingeben.

Zusätzlich muss er noch zweimal sein Passwort eingeben. Die zwei Passwörter werden geprüft, sollten sie nicht ident sein, wird eine entsprechende Fehlermeldung ausgegeben und der Vorgang des Registrierens muss wiederholt werden.

```
if (op.Password == lm.Passwordwh)
{
    repOperatorAccess.Create(op);
    FormsAuthentication.SetAuthCookie(lm.Username, true);
    return RedirectToAction("Index", "Home");
}
```

Abbildung 28: Überprüfung der Passwörter

8.3.2. Stammdaten eingeben

DEFINITION

Nach erfolgreichem Registrieren, müssen die Stammdaten des Benutzers eingegeben werden. Diesbezüglich klickt man auf *Über mich*. Dort werden grundlegende Daten wie Name, Geburtsdatum, Größe und Gewicht eingegeben. Zusätzlich wählt man noch aus, ob eine Insulinpumpe oder ein Pen verwendet wird. Das für die Berechnung wichtigste Feld ist der Umrechnungsfaktor. Diese Daten können jederzeit, durch Klick auf *bearbeiten*, aktualisiert werden.

IMPLEMENTIERUNG

Die Eingabe dieser Daten werden mit dem HTML Tag *TextBox* erfasst. Ausgenommen hierbei sind das Geschlecht und das Insulingerät. Da es bei diesen beiden Daten eine Auswahlmöglichkeit zwischen weiblich oder männlich sowie Pen oder Pumpe gibt, haben wir hierfür das HTML Tag *RadioButton* verwendet.

Klickt man im oberen Bereich auf *Über mich*, erhält man eine Ansicht der oben genannten Daten. In diesem Fenster kann man diese Daten aber nicht ändern, sondern nur ansehen. Dies ist durch den Befehl *ReadOnly = true* möglich.

Klickt man auf den *ActionLink bearbeiten*, öffnet sich ein neues Fenster, indem die Daten bearbeitet werden können. Im *AboutMeModel* werden die eingegebenen Daten kurzzeitig gespeichert. Mit Hilfe der Methode *Update* des *Repositories*, werden diese Daten auf der Datenbank gespeichert.

```
<td>Insulingerät:</td>
<td>
  <div>
    <label>@Html.RadioButtonFor(m => m.insulinDevice, "pen")Pen</label>
    <label>@Html.RadioButtonFor(m => m.insulinDevice, "pumpe")Pumpe</label>
  </div>
</td>
```

Abbildung 29: HTML Tag *RadioButton*

8.3.3. Lebensmittelliste anzeigen

DEFINITION

Nach erfolgreichem Einloggen und nach Eingabe der Berechnungsparameter, siehe 8.3.8 dieser Arbeit, müssen die zu konsumierenden Lebensmittel ausgewählt werden. Dafür wird bei der Berechnung auf den Link *auswählen* geklickt und es erscheint eine neue Ansicht, nämlich eine Liste der Lebensmittelgruppe. Diese Darstellung beinhaltet ein definiertes Bild für diese Gruppe und einen Gruppennamen. Durch Klick auf den Namen der Lebensmittelgruppe erscheint eine Seite mit allen Lebensmitteln, welche der gewählten Lebensmittelgruppe zugeordnet sind.

IMPLEMENTIERUNG:

Beim Anzeigen der Lebensmittelliste gibt es zwei Seiten zu betrachten. Einerseits die Darstellung der Lebensmittelgruppe und andererseits die einzelnen Lebensmittel.

Bei beiden Varianten werden jeweils das Bild und der Text dargestellt. Das Bild und der Name werden aus der Datenbank am Server ausgelesen.

Da das Bild als Datentyp *Blob* in der Datenbank eingetragen ist, ist das Auslesen eines Bildes schwieriger als der Name.

BILD AUSLESEN:

Damit das Bild eingelesen werden kann, muss eine eigene Klasse erstellt werden. Diese Klasse implementiert den *IHttpHandler*, welcher mittels der Verwendung der Bibliothek *System.Web* benutzt werden kann. Dieser Handler enthält eine Methode namens *ProcessRequest* und an die Methode wird ein *HttpContext* übergeben. Diese Methode wird aufgerufen, sobald in einem HTML Tag, Elemente welche auf einer Webseite angezeigt werden können, der Name dieser Klasse aufgerufen wird.

In der erstellten Klasse wird mittels der oben angeführten Methode festgelegt, dass es sich um ein Bild handelt. Ebenfalls wird das definierte Bild, welches mittels eines SQL Statements abgerufen wird, byteweise eingelesen und an jener Stelle auf der Webseite angezeigt, wo sich das HTML Tag befindet. Mittels der *id* wird der *Primary Key* des Datensatzes in der auszulesenden Tabelle angegeben.

```

```

Abbildung 30: HTML Tag – Anzeige eines Bildes aus einer Datenbank

Damit aber festgelegt ist, dass es sich um eine Klasse handelt, muss diese in der *Web.config* Datei unter dem Punkt *handlers* eingetragen werden. Bei *name* wird der Klassenname definiert, *verb* gibt die HTTP Methode an. *Get* bedeutet, dass nur von der Datenbank gelesen wird. Unter dem Punkt *Path* wird der Klassenname mit dem Datentyp *ashx* definiert. Der *Type* zeigt, wo sich die Klasse im gesamten Projekt befindet.

```
<add name="showImage" verb="GET" path="showimage.ashx" type="DiabetesHelferFuerKinderData.showImage" />
```

Abbildung 31: Handler definieren

Damit das Bild auch erkannt werden kann, muss der Dateityp *ashx* definiert werden. Daher wird in der *RouteConfig.cs* Datei ein *IgnoreRoute* eingetragen. Dieser Eintrag besagt, dass diese Dateien nicht für weitere Zwecke akzeptiert werden. Lediglich für das Einlesen der Bilder.

```
routes.IgnoreRoute("{resource}.ashx/{*pathInfo}");
```

Abbildung 32: IgnoreRoute eintragen

Text anzeigen

Der Text hingegen wird mittels der sogenannten *Repositories* ausgelesen. Hierbei wird für das Einfügen, Bearbeiten und Löschen von Datensätzen jeweils eine Methode geschrieben. Diese Methoden können mit der Datenbank kommunizieren. Um den Text anzuzeigen, wird die Methode *GET* aufgerufen, welche nur Daten aus der Datenbank liest.

Am Beginn werden die Lebensmittelgruppen dargestellt. Damit danach die zugeordneten Lebensmittel angezeigt werden können, muss der Primary Key mitgegeben werden, welcher die Lebensmittelgruppe definiert. Der Name wird hierbei als *Action Link* angezeigt, dies bedeutet, dass auf den Namen geklickt werden kann und dabei wird eine Aktion ausgeführt. Der *Action Link* enthält als ersten Parameter den Wert, welcher auf der Webseite angezeigt werden soll, als zweiten die *HTML* Seite, welche aufgerufen werden soll. Der Dritte Parameter definiert den Ordner, in welcher sich die *HTML* Seite befindet und der Vierte sind Variablen, welche mitgeschickt werden.

```
<td align="center">  
    @Html.ActionLink(line.Label, "Index", "FoodList",  
    new { idfoodgroup = @line.FoodGroupID }, null)  
</td>
```

Abbildung 33: Übergabe des Primary Keys von der Lebensmittelgruppe

In der *Abbildung 33*: Übergabe des Primary Keys von der Lebensmittelgruppe ist zu erkennen, dass unter *idfoodgroup* der *Primary Key* der gewählten Lebensmittelgruppe mitgegeben wird. Somit hat man den Schlüssel und kann in der Datenbank jene Lebensmittel anzeigen lassen, welche einen Fremdschlüssel besitzen, die identisch mit dem Primary Key der Lebensmittelgruppe sind.

Nun wird eine zweite Seite, welche die speziellen Lebensmittel anzeigt, nach dem oben genannten Schema aufgebaut. Es muss lediglich eine neue Klasse geschrieben werden, da es sich bei den Bildern um eine andere Tabelle handelt.

8.3.4. Lebensmittel zur Lebensmittelliste hinzufügen

DEFINITION

Nach erfolgreichem Einloggen klickt der Benutzer auf *Lebensmittelliste*. Es werden im ersten Schritt die Lebensmittelgruppen angezeigt. Nach Auswahl der Lebensmittelgruppe werden alle Lebensmittel grafisch dargestellt, welche dieser Lebensmittelgruppe zugeordnet sind. Für weitere Informationen zu dieser Funktion siehe 8.3.3. dieser Arbeit.

Ein Lebensmittel wird hinzugefügt, sobald dieses Produkt in der Datenbank gespeichert ist.

Wenn die Lebensmittel angezeigt werden, befindet sich am Ende der Webseite ein Button mit dem Namen *hinzufügen*. Nach dem Klick auf diesen Button erscheint eine Maske. Auf dieser Maske müssen alle relevanten Daten zu diesem Lebensmittel angegeben werden. Dazu zählt der Name des Lebensmittels, die Anzahl der Kohlenhydrate in Gramm oder bei Getränken in Milliliter. Diese Angabe entspricht einer Broteinheit. Weiters wird ein Bild angegeben und es kann optional eine Berechnungshilfe eingegeben werden.

IMPLEMENTIERUNG

Nachdem die einzelnen Lebensmittel bereits angezeigt werden, haben wir eine eigene HTML Seite geschrieben, welche für das Lebensmittel hinzufügen zuständig ist.

In dieser Seite gibt es insgesamt fünf Eingabefelder. Es gibt jeweils 4 Texteingabefelder. Diese sind bestimmt für den Namen des Lebensmittels, die Anzahl an Gramm, welche eine Broteinheit entsprechen, und zwei Felder für optionale Messungshilfen. Das letzte Eingabefeld ist für das Lebensmittelbild zuständig.

Das Lebensmittelbild wird mittels *input* HTML Tag ausgewählt. Der *Type* dieses HTML Tag ist *file*. Aufgrund dieser Einstellung öffnet sich der Explorer des Anwenders und dieser kann dadurch mit der Maus das Bild auswählen. Sobald er das Bild bestätigt, wird das Bild zwischengespeichert.

Drückt der Anwender auf *speichern*, wird das Bild in ein *byte[]* umgewandelt. Dafür ist eine eigene Methode geschrieben worden, wobei nur das übergebene *Image* konvertiert wird.

```
public static byte[] ImageToByte(Image img)
{
    ImageConverter converter = new ImageConverter();
    return (byte[])converter.ConvertTo(img, typeof(byte[]));
}
```

Abbildung 34: Image in byte[] umwandeln

Als letzten Schritt werden alle Daten in der Datenbank gespeichert. Hierfür werden die eingegebenen Daten in eine Variable zwischengespeichert. Aus dieser Variable werden diese Daten herausgeholt und als Eigenschaften für ein eigenes *Food* Objekt definiert. Das Bild wird nach der Konvertierung ebenfalls als Eigenschaft definiert.

Damit nun das Objekt dauerhaft gespeichert wird, muss es noch zur Datenbank geschickt werden. Dies geschieht wieder mit den erstellten *Repository*.

8.3.5. Lebensmittel auswählen

DEFINITION

Nach erfolgreichem Einloggen klickt der Benutzer auf *Lebensmittelliste*. Danach werden die Lebensmittelgruppen angezeigt. Sobald die Lebensmittelgruppe ausgewählt worden ist, erscheinen die Lebensmittel, siehe 8.3.3 dieser Arbeit.

Wird der Name des gewünschten Lebensmittels angeklickt, erscheint eine Ansicht, die Berechnungsansicht, siehe dazu auch 8.3.8. dieser Arbeit. In dieser Ansicht erscheint nun das gewählte Lebensmittel. Drückt man wiederrum auf den Lebensmittelname in dieser Ansicht, erscheinen die Details zu diesem Lebensmittel.

IMPLEMENTIERUNG

Um ein Lebensmittel auswählen zu können, muss zuerst die Funktion *Lebensmittelliste anzeigen* implementiert werden, siehe 8.3.3.

Dass ein Lebensmittel ausgewählt werden kann, muss lediglich der *Action Link* auf die HTML Seite von Berechnen verlinkt werden. Außerdem wird hierbei der *Primary Key* des gewählten Lebensmittels mitgegeben.

```
<td align="center">  
    @Html.ActionLink(line.Label, "Index", "Calculation",  
        new { foodid = @line.FoodId }, null)  
</td>
```

Abbildung 35: ActionLink Lebensmittel auswählen

Im Hintergrund wird nun der Name des gewählten Lebensmittels aus der Datenbank geholt. Danach wird dieser auf der Berechnungsseite wieder als *Action Link* angezeigt.

Sobald in der Berechnungsmaske nun der Name ausgewählt wird, erscheint eine eigene *HTML* Seite, welche alle Details zu diesem Lebensmittel anzeigt.

Im Hintergrund hierbei werden wieder alle Daten zu diesem Lebensmittel mittels der Repository aus der Datenbank geholt. Die Darstellung selbst geschieht wiederum mit den *HTML* Tags und verwendet die Variablen, welche aus der Datenbank geholt und zwischengespeichert wurden.

8.3.6. Informationen einholen

Im Reiter *Information* kommt der Benutzer auf eine weitere Seite. Es erscheint ein grober Überblick über das Thema Diabetes und weitere Verlinkungen auf Internetseiten, welche dieses Thema genauer erklären.

8.3.7. Statistik anzeigen

IMPLEMENTIERUNG

Nach Eingabe der notwendigen Daten im Reiter *Über Mich* und der Berechnung der Insulinwerte, klickt der Benutzer auf *Statistik*. Nun muss der Zeitraum für die Statistik angegeben werden. Dies erfolgt über drei *Comboboxen*, in denen der Tag, das Monat und das Jahr ausgewählt werden. Nach Klicken auf den Button *Auswerten* erscheint eine grafische Statistik zu diesem Tag. Die Statistik zeigt die Insulin- und Blutzuckerwerte des Tages an und wann sich diese Werte veränderten. Durch Klick auf einen definierten Punkt im Diagramm erscheinen detaillierte Informationen zu diesem Zeitpunkt.

8.3.8. Berechnungen

DEFINITION

Nach dem erfolgreichen Einloggen hat der Benutzer die Möglichkeit sich seine IE berechnen zu lassen. Hierfür klickt er im oberen Bereich der Web-Applikation auf *Berechnung*. Dort muss das Datum und die Uhrzeit eingegeben werden. Für die Berechnung ist außerdem der aktuelle Blutzuckerwert nötig. Dafür muss der Benutzer diesen mit einem geeigneten Blutzuckermessgerät messen und gibt auch diesen Wert ins Programm ein.

Ist dieser Wert höher als 120, ist er zu hoch und das Eingabefeld wird orange hinterlegt. Ist der Wert kleiner als 70, ist er zu niedrig und das Feld wird rot hinterlegt.

Als letztes muss der Benutzer noch die Lebensmittel auswählen, welche er verzehren wird.

Durch Klick auf den Button *Berechnen*, werden die zu verabreichenden Insulineinheiten, unter Berücksichtigung des Umrechnungsfaktors der Stammdaten, des aktuellen Blutzuckerwertes und der ausgewählten Lebensmittel berechnet.

IMPLEMENTIERUNG

Das Datum und die Uhrzeit werden über das HTML Tag *TextBox* eingegeben. Im Model *CalculationModel* werden diese zwei Daten unter den Datentyp *DateTime* gespeichert. Außerdem ist für die Berechnung der aktuelle Blutzuckerwert wichtig. In einer Methode im *CalculationController* wird dabei geprüft, ob der Wert zu hoch oder zu niedrig ist.

8.4. Datenbankmodell

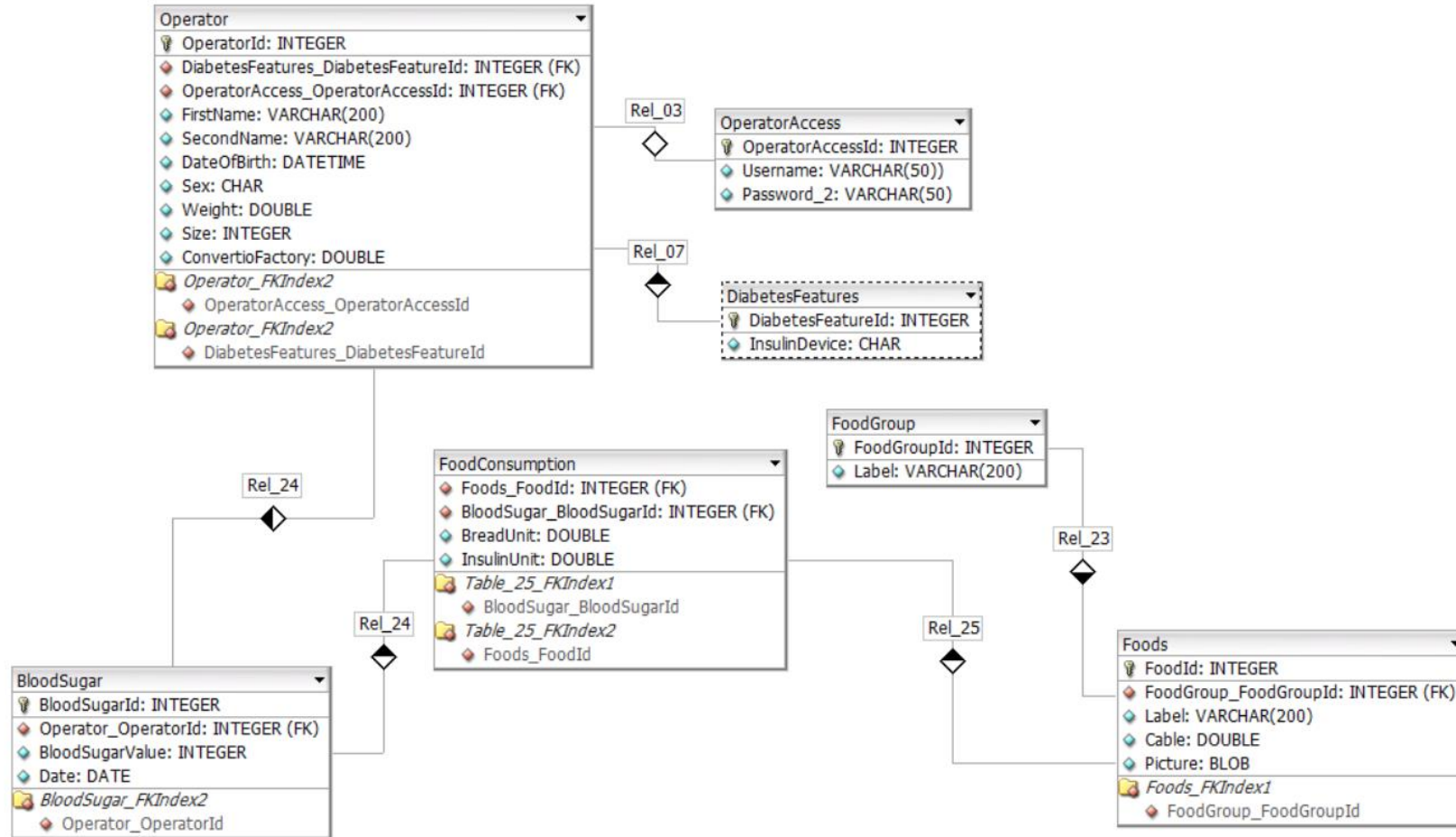


Abbildung 36: Datenbankmodell

Damit Daten dauerhaft gespeichert werden, verwenden wir eine MySQL Datenbank.

Im Gesamten gibt es sieben Tabellen, nämlich *Operator*, *OperatorAccess*, *DiabetesFeatures*, *BloodSugar*, *FoodConsumption*, *Foods* und *FoodGroup*. Die Datenbank MySQL ist auf dem Server, welcher unter dem Namen „*diabeteshelfer.htl-perg.ac.at*“ erreichbar ist, installiert. Das Datenbankmodell selbst ist auf dem Server in MySQL implementiert. Wenn Daten aufgerufen oder gespeichert werden, geschieht dies in der Datenbank am Server.

Die Tabelle *Operator* speichert alle relevanten Daten zu einem Benutzer. Diese sind *Vorname* und *Nachname* des Anwenders, welche als Zeichenkette gespeichert werden. Geburtsdatum wird als Datumformat gespeichert.

Weiters wird das Geschlecht als einzelner Buchstabe gespeichert. Dies bedeutet, dass der Anwender entweder männlich, in diesem Fall „*m*“, oder Weiblich „*w*“ ist. Das Gewicht wird in Kilogramm angegeben. Aus diesem Grund ist der Datentyp ein „*double*“, das heißt, dass der Wert Kommastellen enthalten kann. Die Größe hingegen wird in Zentimeter gespeichert, deshalb ist der Datentyp eine Dezimalzahl. Als letzter Wert in dieser Tabelle wird der Umrechnungsfaktor gespeichert, welcher vom Krankenhaus berechnet wird. Dieser Faktor wird für die Berechnungen benötigt und wird ebenfalls als *double* gespeichert.

Ab diesem Zeitpunkt sind die Daten dauerhaft gespeichert. Es besteht aber auch die Möglichkeit diese Daten zu ändern. Ein Grund dafür ist, dass die Körpergröße eines Kleinkindes aufgrund des Wachstums im Vergleich zu einem Erwachsenen weitaus stärkeren Änderungen unterliegt.

Die Tabelle *OperatorAccess* speichert die Anmeldedaten des Anwenders. Diesbezüglich wird der Username in Form einer Zeichenkette gespeichert, da er bei diesem Programm eine E-Mail Adresse ist. Weiters speichert diese Tabelle das zugehörige Passwort.

Das Passwort muss mindestens einen Großbuchstaben, eine Zahl und ein Sonderzeichen enthalten. Diese Richtlinie wird beim Registrieren abgeprüft, weil dabei der Benutzername und das Passwort festgelegt werden. Ebenfalls sind ab der Registrierung diese Daten gespeichert.

Die Tabelle *OperatorAccess* ist mit einer *1:1-Beziehung* mit der *Operator* Tabelle verbunden, da einem Anwender genau ein Username und ein Passwort zum Einloggen zur Verfügung stehen.

Weiters speichert die Tabelle *DiabetesFeatures* das Insulingerät, welches der Anwender zum Spritzen der notwendigen Insulindosis verwendet. Dabei unterscheidet man zwischen einem Pen und einer Pumpe.

Die Tabelle *DiabetesFeatures* ist mit einer *1:n- Beziehung* mit der Tabelle *Operator* verbunden, da ein Anwender entweder einen Pen oder eine Pumpe zum Spritzen der Insulineinheiten verwenden muss.

Die Tabelle *BloodSugar* speichert den Blutzuckerwert. Da ein Diabetiker mehrmals am Tag seinen Blutzuckerwert messen muss, gibt er diesen öfter am Tag ein. Deshalb wird zum Blutzuckerwert das Datum und die Uhrzeit mitgespeichert. Dieses Datum mit Uhrzeit wird in Form eines Datumsformat gespeichert.

Die Tabelle *BloodSugar* ist mit der Tabelle *Operator* mit einer *1:n- Beziehung* verbunden, weil zu einem Anwender mehrere Blutzuckerwerte gehören, aber ein Blutzuckerwert genau zu einem Anwender gehört. Das heißt dass bei der Tabelle *BloodSugar* als Fremdschlüssel der Primärschlüssel der *Operator* Tabelle steht, damit man den Blutzuckerwert dem Anwender zuordnen kann.

Die Tabelle *FoodGroup* speichert die Lebensmittelgruppen, in welche die Lebensmittel eingeteilt werden können. Dabei wird der Name der Lebensmittelgruppe als Zeichenkette gespeichert.

Weiters wird dazu ein Bild in Form eines *Blobs* gespeichert. Das Bild in der Datenbank ist wichtig, weil diese Webseite nicht nur auf einem Computer sondern auch auf einem Handy angezeigt werden soll. Weiters sind die Bilder notwendig, da Kleinkinder noch nicht lesen können, jedoch aufgrund der Bilder das betreffende Lebensmittel auswählen können.

Eine weitere Tabelle namens *Foods* speichert die einzelnen Lebensmittel. Zu einem Lebensmittel wird dessen Name mittels einer Zeichenkette gespeichert. Außerdem werden die Kohlenhydrate zu diesem Lebensmittel gespeichert. Die Kohlenhydrate werden in Gramm angegeben, die Getränke in Milliliter, und diese entsprechen genau einer Broteinheit. Damit das Kleinkind wieder weiß, um welches Lebensmittel es sich handelt, wird ein Bild in Form eines *Blobs* gespeichert.

Die Tabelle *Foods* ist mit der Tabelle *FoodGroup* mit einer *1:n-Beziehung* verbunden. Ein Lebensmittel gehört genau zu einer Lebensmittelgruppe, wobei zu einer Lebensmittelgruppe mehrere Lebensmittel gehören. Der Primärschlüssel der *FoodGroup* wird als Fremdschlüssel in der Tabelle *Foods* gespeichert, da ein Lebensmittel zu einer Lebensmittelgruppe zugeordnet werden kann.

Die letzte Tabelle ist *FoodConsumption*. Diese Tabelle verbindet die Tabelle *BloodSugar* mit der Tabelle *Foods*. Deshalb hat die Tabelle *FoodConsumption* eine Spalte, welche den Primärschlüssel von *BloodSugar* als Fremdschlüssel enthält und eine Spalte, welche den Primärschlüssel von *Foods* als Fremdschlüssel enthält. Diese beiden Spalten bilden gemeinsam den Primärschlüssel dieser Tabelle, damit zu einem Blutzuckerwert mehrere Lebensmittel konsumiert werden können. Im Gegenzug dazu kann ein Lebensmittel zu mehreren Blutzuckerwerten zugeordnet werden.

Um die Berechnung der Gesamtinsulinwerte für eine Mahlzeit zu erleichtern, werden jeweils der Insulinwert, der aufgrund eines Lebensmittels gespritzt werden soll und die Broteinheiten für ein Lebensmittel gespeichert. Die Broteinheit wird vom Anwender bestimmt, wie viel er vom diesem Lebensmittel zu diesem Zeitpunkt zu sich nehmen wird. Diese beiden Werte werden für jedes Lebensmittel einzeln berechnet.

9. UML Klassendiagramm

Diabetes Helfer für Kinder Data

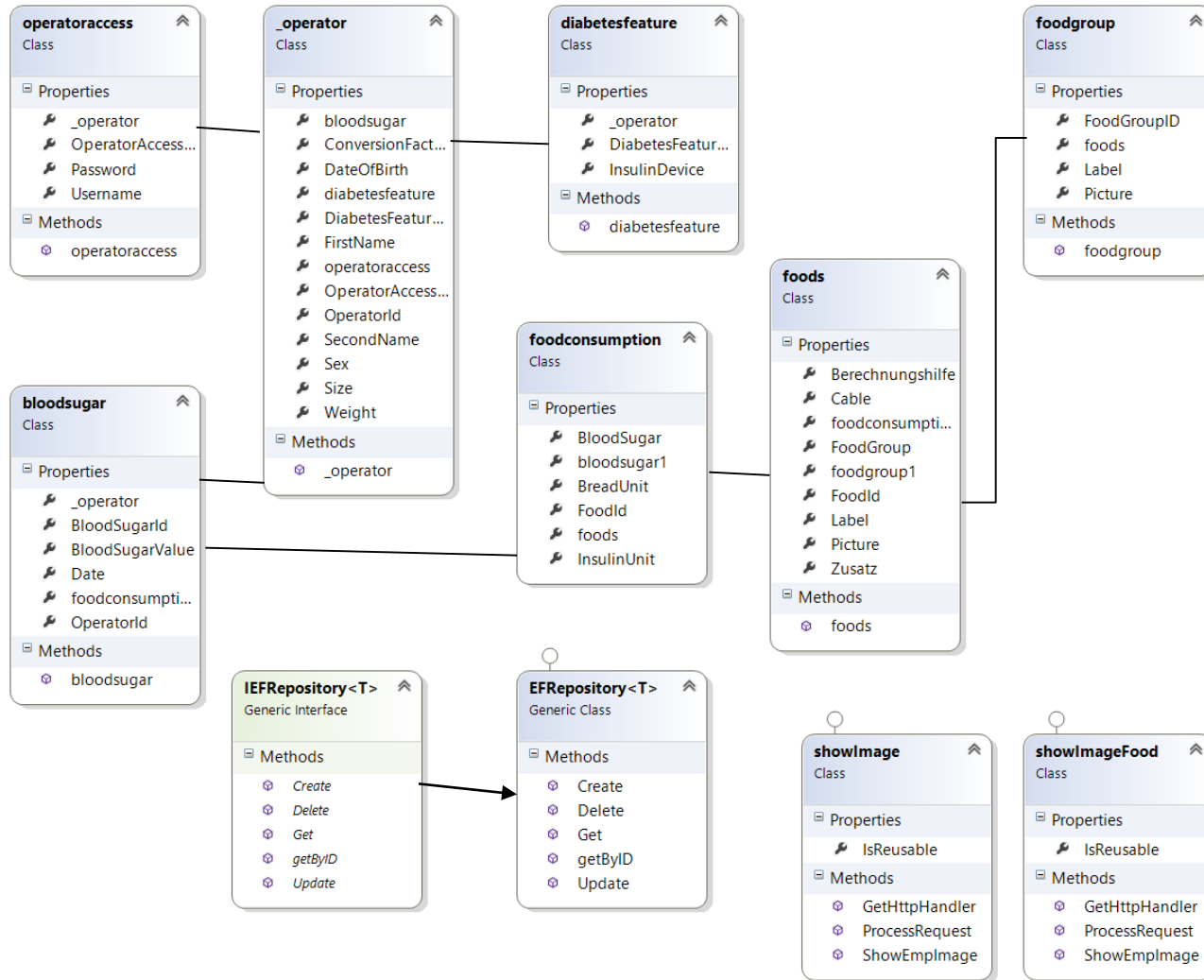
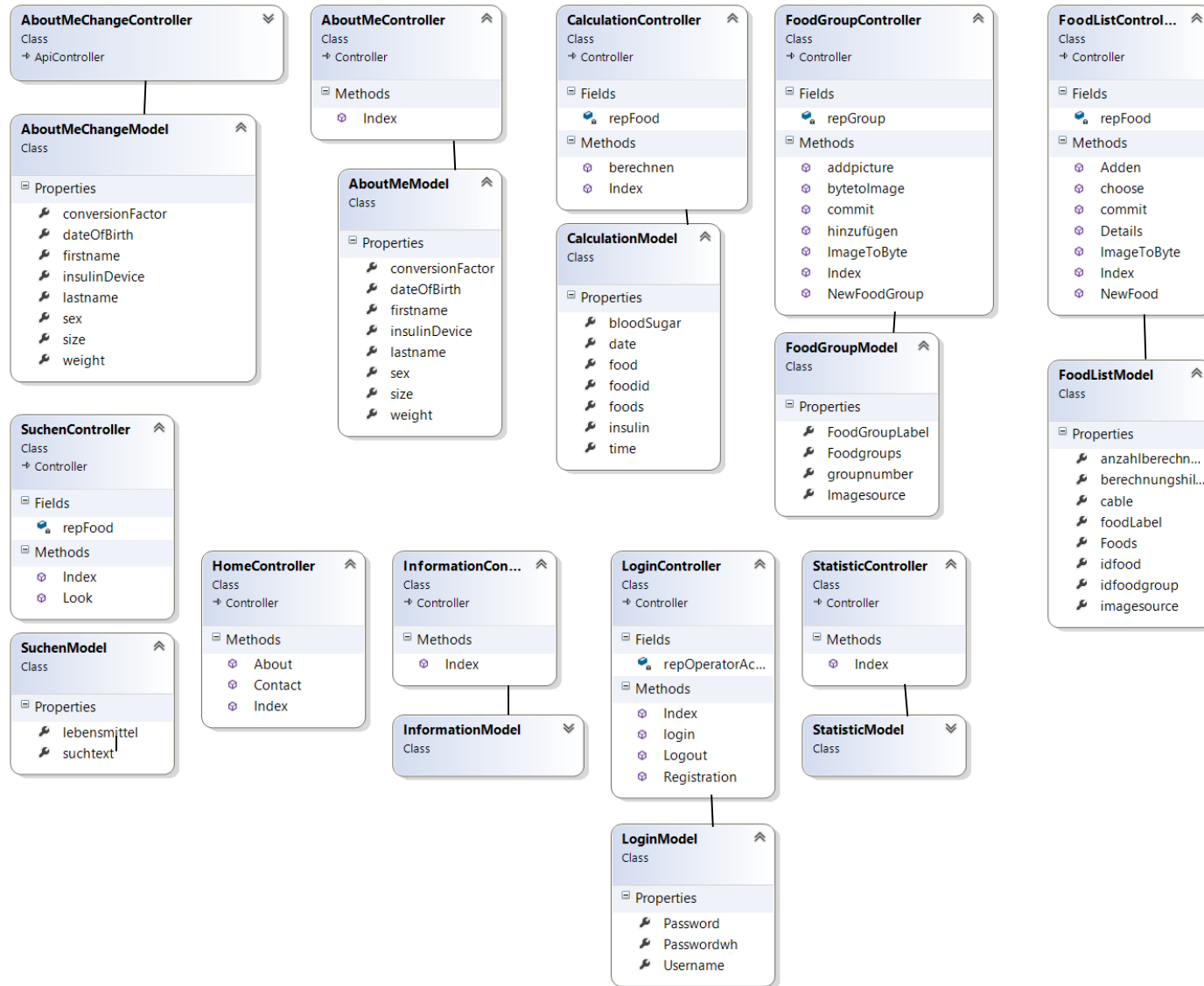


Abbildung 37: UML - Klassendiagramm Package Data

Diabetes Helfer für Kinder



Diabetes Helfer für Kinder Web

Abbildung 38: UML - Klassendiagramm Package Web

Das UML Klassendiagramm zeigt zwei Projekte, *Diabetes Helfer für Kinder Data* und *Diabetes Helfer für Kinder Web*. Das *Data* – Projekt beinhaltet alle Objektklassen. Die Klassen *operatorAccess*, *operator*, *diabetesfeature*, *bloodsugar*, *foodconsumption*, *foods*, *foodgroup* haben wir uns automatisch erstellen lassen. Diese Klassen sind die Objektklassen der Datenbank, das heißt mit diesen Klassen können Daten aus der Datenbank gelesen und in die Datenbank gespeichert werden.

Die Klasse *EFRepository* ist von *IEFRepository* abgeleitet und implementiert die im Interface festgelegten Methoden. Diese Methoden sind Schnittstellen für die Datenbank. Diese ermöglichen es Daten aus der Datenbank zu lesen oder Daten in die Datenbank zu speichern.

Das *Web* – Projekt ist für die Darstellung der Daten auf der Webseite verantwortlich. Für jede Funktion, welche die Web-Applikation bietet, gibt es einen *Kontroller*, ein *Model* und ein oder mehrere *Views*. Die *Views* werden im Diagramm nicht dargestellt.

Der *Kontroller* beinhaltet die Logik der Funktion. Der *AboutMeController* beispielsweise hat Methoden zum Herauslesen der Benutzerdaten. Das wichtigste eines Kontrollers ist, dass jede Methode in diesem Kontroller genau ein *Model* übergeben bekommt, mit welcher die Verbindung zur *View* hergestellt wird.

Das *Model* speichert die Variablen, welche bei der Darstellung benötigt werden. Diese Variablen sind Zwischenspeicher. Es werden nur jene Variablen gesetzt, welche in einer *View* verwendet werden. Alle anderen werden nicht gesetzt.

Eine *View* ist die eigentliche Darstellung der HTML Seite. Dafür werden die *HTML Tags* verwendet. Einer *View* wird genau ein *Model* übergeben, welches der *View* ermöglicht auf die Variablen im *Model* zuzugreifen.

10. Qualitätssicherung

10.1. Aufgetretene Probleme und Lösungen

10.1.1. E-Mail Adresse statt Benutzername

Beim Erstellen des Grundkonzepts, wurde von uns zuerst geplant, dass sich ein Benutzer mit einem Benutzernamen einloggen kann. Jedoch wurde uns bei der Umsetzung bewusst, dass dies nicht die optimale Lösung war, da mehrere Benutzer denselben Benutzernamen haben könnten. Deswegen entschlossen wir uns statt Benutzernamen E-Mail Adressen zu verwenden. Dadurch kann das Problem, dass zwei Benutzer denselben Benutzernamen haben, nicht mehr auftreten, da E-Mail Adressen weltweit eindeutig sind.

Außerdem sind wir bei der Umsetzung auf das Problem gestoßen, dass bei einer neuen Registrierung der Benutzer das Passwort zweimal eingeben sollte, damit keine Komplikationen mit dem gewählten Passwort in Form von Rechtschreibfehlern auftreten können.

10.1.2. MySQL Connector erstellen

Bei allen Diplomarbeitern war die Datenbank *MySQL* am Laptop bereits installiert. Wir probierten, den *MySQL Connector* danach zu installieren. Die Installation war aber nie erfolgreich. Unsere Lösung für dieses Problem war, dass wir *MySQL* deinstallierten.

Das nächste Problem ließ nicht lange auf sich warten. *MySQL* konnte nicht deinstalliert werden. Der Grund dafür ist uns immer noch unbekannt. Durch den Neustart des Gerätes lösten wir jedoch dieses Problem.

Da wir nun zuerst den *MySQL Connector* installiert hatten und danach die *MySQL Datenbank* installierten, sind folglich beide Programme erfolgreich installiert worden.

10.1.3. Verbindung zum Server

Da bei dieser Diplomarbeit zuerst das Datenmodell erstellt wurde, haben wir uns die *Entities* erstellen lassen. Das Problem allerdings bestand darin, dass eine Verbindung zum Server und zur Datenbank schon bestand. Wir konnten die *Entities* einmal erstellen, jedoch konnte die Verbindung der Datenbank nicht mehr aufgebaut werden.

Der Connection String, welcher die Verbindung zur Datenbank herstellt war jedoch korrekt. Wir haben zur Lösung dieses Problems ein eigenes *.NET Projekt* erstellt und die Verbindung erneut programmiert.

Hierfür sind wir identisch wie beim ersten Mal vorgegangen. Es wurde die Verbindung hergestellt und danach haben wir die *Entities* erstellen lassen. Bei dem neuen Projekt hat dies auch funktioniert, also wollten wir dieses in unser Programm einfügen. Jedoch wurde hierbei nur der *ConnectionString* übernommen und nicht die bereits erstellten *Entities*. Diese *Entities* haben wir uns einfach von *Visual Studio* nachinstallieren lassen. Seitdem ist die Datenbankverbindung vorhanden.

Worin genau das Problem lag, ist uns bis heute unbekannt.

10.1.4. Bilder anzeigen

Beim Anzeigen der Lebensmittelliste wird jeweils ein Bild angezeigt. Da in der Datenbank aber der Datentyp ein *Blob* ist, auf einer Webseite aber ein Bild mit dem Datentyp *Image* angezeigt wird, stellte das Herausholen des Bildes ein großes Problem dar.

Schließlich gab uns unser Diplomarbeitbetreuer den Tipp mit der eigenen Klasse, da wir bis zu diesem Zeitpunkt keine geeignete Lösung fanden. Diese Lösung ist bereits in der Implementierung festgehalten, siehe 8.3.3 dieser Arbeit.

10.2. Qualitätsmerkmale

10.2.1. Funktionalität

Die Hauptfunktionalität dieser Diplomarbeit ist das Berechnen der Insulineinheiten pro Brot-einheit, welche ohne dieser Diplomarbeit sehr aufwendig, zeitintensiv und teilweise kompliziert ist. Da für diese Berechnung die Lebensmittel, welche der Benutzer zu sich nimmt wichtig sind, gibt es auch die Funktion Lebensmittel auszuwählen.

Eine weitere Funktion Statistik anzeigen bietet den Benutzern eine Reihe von Möglichkeiten. Der Benutzer kann bei dieser Statistik seinen Blutzuckerwert zu einem gewissen Zeitpunkt einsehen. Weiteres kann er sich in der Grafik anzeigen lassen, wie viel Insulin er zu diesem Zeitpunkt spritzen musste.

Für genauere Informationen zu den einzelnen Funktionen siehe auch 8.3.

10.2.2. Bedienbarkeit

Das Programm selbst ist sehr kinderfreundlich aufgebaut. Im Programm selbst sind Hilfen in Form von Pandabären eingebaut, welche in die Richtung zeigen, die gewählt werden müssen. Die Lebensmittel werden auch als Bild angezeigt, damit ein Kind anhand des Bildes erkennen kann, um welches Lebensmittel es sich handelt. Zur Sicherheit werden auch die Namen der einzelnen Lebensmittel angezeigt, weil manche Lebensmittel optisch verwechselt werden könnten.



Abbildung 39: Teddybär

Trotzdem soll erwähnt sein, dass dieses Programm nicht ohne elterliche Aufsicht verwendet werden soll, da Kleinkinder eher zur einseitigen Kost neigen. Eltern sollen dabei den Überblick über den Blutzucker und über die Essgewohnheiten des Kindes beibehalten.

Im Vergleich zu anderen Programmen liegt der Fokus dieser Diplomarbeit auf der Visualisierung der Lebensmittel in Form von Bildern. Das Kleinkind kann somit optisch wahrnehmen, welche Bereiche ausgefüllt werden sollen, damit der zu berechnende Insulinwert angezeigt wird.

10.2.3. Portabilität

Die Webseite ist auf allen internetfähigen Geräten aufrufbar. Die Darstellung wird an das Gerät angepasst, aufgrund des *Responsive Design*. Der Vorteil auf der Entwicklerseite ist, dass nur eine Version erstellt werden muss. Aus der anderen Perspektive, der Anwendersicht, ist anzumerken, dass sich der Anwender keine Gedanken über das Zugriffsmedium machen muss. Der Anwender kann mit jedem ihm zur Verfügung stehenden, internetfähigen Gerät, die Webseite aufrufen und unser Programm verwenden.

Ein wichtiger Punkt, der hier noch Erwähnung finden soll, ist dass die Webseite betriebssystemunabhängig aufgerufen und benutzt werden kann. Ob Android, Windows oder IOS, es hat keine Bedeutung.

10.2.4. Installierbarkeit

Das Programm ist eine Webanwendung und über das Internet aufrufbar. Es muss kein zusätzliches Programm auf dem Gerät installiert werden, von dem aus auf die Webseite zugegriffen wird.

Das Gerät muss lediglich einen funktionierenden Browser besitzen und eine aktive Internetverbindung haben.

10.2.5. Korrektheit

Die Korrektheit bezieht sich im Wesentlichen auf die gespeicherten Daten in der Datenbank. Dies bezieht sich vor allem auf die Gramm pro Lebensmittel, welche eine Broteinheit ergeben. Dieser Wert wird mit den Kohlenhydraten eines Lebensmittels errechnet. Diese speziellen Werte haben wir von einer externen Person, welche jeden Tag mit Diabetes konfrontiert ist, kontrollieren lassen.

Bei den Berechnungen selbst ist die Korrektheit dadurch gegeben, dass nicht nur das Programm den Wert errechnet, sondern wir zur Kontrolle diese auch durchgerechnet haben. Dies geschah ebenfalls mit Absprache der oben genannten externen Person.

10.2.6. Erlernbarkeit

Die Webseite ist in erster Linie kinderfreundlich aufgebaut. Die Menüleiste befindet sich im oberen Bereich der Seite. Im Hauptfenster wird jene Funktion dargestellt, welche aufgerufen wird. Diese Ansicht ist sehr häufig im Internet vertreten, daher ist die Erlernbarkeit sehr einfach. Da die Webseite für Kinder erstellt wurde, ist die Bedienbarkeit zusammenhängend aufgebaut. Aus diesem Grund ist diese Webseite sehr einfach erlernbar und dadurch auch sehr leicht zu bedienen.

10.2.7. Ressourcen

Wenn man den Speicherbedarf auf dem Gerät, auf welchem die Webseite aufgerufen wird, analysiert, kann man erkennen, dass dieser sehr gering ist. Der Grund dafür ist, dass sich alle wesentlichen Informationen auf den externen Server befinden. Diese Informationen befinden sich somit nicht direkt auf dem Gerät sondern extern.

10.3. Testfälle

In den folgenden Testfällen wird der Insulinwert berechnet, welcher der Diabetiker sich verabreichen soll. Dabei ist das Gericht, welches der Diabetiker zu sich nimmt gleich.

Der Insulinwert wird für ein Kleinkind berechnet, welches zu einer Mahlzeit etwa vier Broteinheiten konsumieren soll. Aus einer Liste werden die Gramm für eine Broteinheit herausgelesen.

UMRECHNUNGSFAKTOR:

- 0,6

MENÜ:

- Kopfsalat
- Käsespätzle
- Schokoladenmousse

Menü	Gramm / 1 BE	Bemerkung
Kopfsalat		wird nicht berücksichtigt
Käsespätzle	66,7 g	
Schokomousse	120 g	

Da der Kopfsalat nicht berücksichtigt wird werden die 4 Broteinheiten auf die Käsespätzle und das Schokomousse gleichmäßig aufgeteilt.

10.3.1. Testfall: 1

Gemessener Blutzuckerwert: 40 mg

Da der Blutzuckerwert nicht in der Norm liegt (70-120), ist es notwendig diesen zu korrigieren. Der Blutzuckerwert ist zu niedrig, daher muss das Kind mehr essen. In unserem Fall bekommt das Kind aber nicht mehr, deshalb werden statt vier BE nur drei BE gerechnet. Das Resultat ist dasselbe, als würde das Kind mehr zu sich nehmen.

Broteinheit * Umrechnungsfaktor

$3 * 0,6 = 1,8 \text{ IE}$

Das Kleinkind darf von den Käsespätzle 133,4g essen und vom Schokomousse darf es 240g essen. Vom Kopfsalat darf das Kind essen so viel es möchte. Damit das Kind diese Menge an Lebensmittel zu sich nehmen kann und damit der Blutzuckerwert wieder in der Norm ist, muss es sich 1,8 IE zuführen.

10.3.2. Testfall: 2

Gemessener Blutzuckerwert: 90 mg

Da der Blutzuckerwert in der Norm liegt (70-120), ist es nicht notwendig diesen zu korrigieren. Es muss lediglich das Menü in Insulineinheiten umgewandelt werden. Pro Broteinheit ist der Umrechnungsfaktor in Insulineinheiten abgegeben.

Broteinheit * Umrechnungsfaktor

$4 * 0,6 = 2,4 \text{ IE}$

Das Kleinkind darf von den Käsespätzle 133,4g essen und vom Schokomousse darf es 240g essen. Vom Kopfsalat darf das Kind essen so viel es möchte. Damit das Kind diese Menge an Lebensmittel zu sich nehmen kann, muss es sich 2,4 IE zuführen.

10.3.3. Testfall: 3

Gemessener Blutzuckerwert: 150 mg

Da der Blutzuckerwert nicht in der Norm liegt (70-120), ist es notwendig diesen zu korrigieren. Im Regelfall wird dieser Wert auf 100 gesenkt. Pro Zehnerschritt werden 0,1 IE dazurechnet.

<p>Broteinheit * Umrechnungsfaktor + korrigierter Wert</p>
--

$4 * 0,6 + 0,5 = 2,9 \text{ IE}$

Das Kleinkind darf von den Käsespätzle 133,4g essen und vom Schokomousse darf es 240g essen. Vom Kopfsalat darf das Kind essen so viel es möchte. Damit das Kind diese Menge an Lebensmittel zu sich nehmen kann und damit der Blutzuckerwert wieder in der Norm ist, muss es sich 2,9 IE zuführen.

11. Evaluierung

11.1. Kompatibilität

Die Vorteile des *Responsive Designs* sind die dynamische Darstellung der Daten und die automatische Anpassung der Ansicht je nach Bildschirmgröße. Dabei ist es egal, ob es sich beim anzeigenden Gerät um einen Desktop, Tablet oder Smartphone handelt. Dies ist von Vorteil, da so das Programm auch mobil verwendet werden kann, wenn der Benutzer zum Beispiel in einen Restaurant isst.

11.2. Voraussetzungen

Damit die Webseite erfolgreich genutzt werden kann, sind folgende Voraussetzungen zu beachten:

11.2.1. internetfähiges Gerät

Damit die Webseite überhaupt aufgerufen werden kann, benötigt man ein elektronisches Gerät. Auf diesem Gerät ist ein Internetbrowser installiert. Durch diesen installierten Browser ist das Gerät internetfähig, das heißt, dass das Gerät Internetseiten aufrufen kann. Bei diesem Gerät muss es sich nicht um ein mobiles Gerät handeln. Es kann zum Beispiel ein Mobiltelefon, Notebook oder ein Stand PC sein.

11.2.2. aktive Internetverbindung

Eine weitere Voraussetzung ist eine aktive Internetverbindung. Das internetfähige Gerät, siehe 11.2.1 dieser Arbeit, muss mit einem Kabel oder kabellos direkt mit dem Internet verbunden sein.

11.2.3. Blutzuckermessgerät

Der Anwender unserer Webseite muss bei der Berechnung der Insulineinheiten seinen aktuellen Blutzuckerwert eingeben. Dieser Blutzuckerwert kann nur mit einem externen Blutzuckermessgerät gemessen werden. Dieses Gerät bekommt der Diabetiker, sobald die Diagnose gestellt wird.



Abbildung 40: Blutzuckermessgerät „Bayer“

Das Blutzuckermessgerät besteht aus zwei Teilen. Das Erste ist das Gerät selbst, welcher den Blutzuckerwert misst. Das Zweite ist ein Teststreifen. Dieser wird beim vorgesehenen Einstechloch im Messgerät befestigt. Auf der anderen Seite wird ein Blutstropfen aufgetragen.

Um einen Blutstropfen zu erhalten, benötigt man eine Stechhilfe. Mit dieser Stechhilfe wird ein gezielter Stich gesetzt. Bei diesem Hilfsgerät muss die Lanzette, die Nadel, eingesetzt und die Stechtiefe eingestellt werden. Danach kann der zweite Blutstropfen zur Messung herangezogen werden. Beim Ersten könnten durch den Stich Messfehler entstehen.

[Diabetes Ratgeber, 2014]

11.3. Resümee

Durch die Diplomarbeit konnte unsere Gruppe ihr Wissen in C#, ASP.NET und der Webentwicklung erweitern. Durch die aufgetretenen Probleme waren wir gezwungen uns tiefer mit der Materie zu befassen und konnten so Lösungen finden. Dadurch, dass wir das Projekt selbst planen konnten und die Aufgaben so eingeteilt haben, dass jeder die Aufgaben zu erledigen hatte, welche er am besten konnte und falls doch Probleme auftraten, wurden diese mit Hilfe der anderen Teammitglieder behoben. Falls diese ebenfalls ratlos waren, konnten wir uns immer auf die tatkräftige Unterstützung der Professoren verlassen. Zum Schluss lässt sich sagen dass die Diplomarbeit das selbständige Arbeiten und Planen der einzelnen Personen der Diplomarbeitsgruppe verbessert hat und das alle Erfahrung dadurch sammeln konnten, welche sie in Zukunft einsetzen können.

.

12. Impressum

Schule

Höhere Technische Lehranstalt für Informatik

Machlandstraße 48

4320 Perg

Schuljahr

2015-2016

Klasse

5AHIF

Projekttitel

Diabetes Helfer für Kinder

Projektteam

Tamara Holzer

Petra Enengl

Patrick Mittermayr

Betreuungslehrer

Dipl.-Ing. Dr. Michael Buchberger

12.1. Diplomanden

3.3.1 Tamara Holzer

Persönliche Daten

Name	Tamara Holzer
Geburtsdatum und –ort	06. Juni 1997, Amstetten
Religionsbekenntnis	römisch-katholisch

Ausbildung

Seit 09/2011	HTL Perg Höhere Lehranstalt für Informatik
09/2007 – 07/2011	HIT Grein Hauptschule für Informationstechnologie
09/2003 – 07/2007	Volksschule Pergkirchen



Abbildung 41: Tamara Holzer

3.3.2 Petra Enengl

Persönliche Daten

Name	Petra Enengl
Geburtsdatum und –ort	13. September 1995, Amstetten
Religionsbekenntnis	römisch-katholisch

Ausbildung

Seit 09/2010	HTL Perg Höhere Lehranstalt für Informatik
09/2006 – 07/2010	HIT Grein Hauptschule für Informationstechnologie
09/2002 – 07/2006	Volksschule Grein



Abbildung 42: Petra Enengl

3.3.3 Patrick Mittermayr

Persönliche Daten

Name	Patrick Mittermayr
Geburtsdatum und –ort	9. November 1996, Linz
Religionsbekenntnis	römisch-katholisch

Ausbildung

Seit 09/2011	HTL Perg Höhere Lehranstalt für Informatik
09/2007 – 07/2011	Hauptschule St. Valentin
09/2003 – 07/2007	Volksschule Ernsthofen

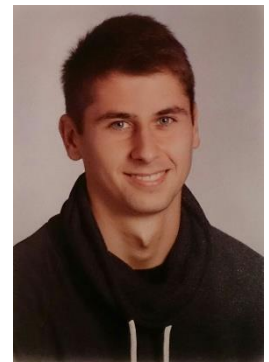


Abbildung 43: Patrick Mittermayr

13. Anhänge

13.1. Quellenverzeichnis

- ALTMANN, U. 2013. *Was ist DOM*. [online]. [Accessed 1 April 2016]. Available from World Wide Web: <<http://www.med.uni-giessen.de/akkk/xml/DOM/>>
- BUNDESMINISTERIUM FÜR GESUNDHEIT. 2015. *www.gesundheit.gv.at*. [online]. [Accessed 23 Mar 2016]. Available from World Wide Web: <<https://www.gesundheit.gv.at>>
- de.statista.com*. 2016. [online]. [Accessed 10 Februar 2016]. Available from World Wide Web: <<http://de.statista.com/themen/262/diabetes/>>
- Diabetes Ratgeber*. 2014. [online]. [Accessed 13 März 2016]. Available from World Wide Web: <http://www.diabetes-ratgeber.net/Blutzucker/So-messen-Sie-Ihren-Blutzucker-richtig-54134_5.html>
- Diabetes Ratgeber*. 2015. [online]. [Accessed 10 Februar 2016]. Available from World Wide Web: <http://www.diabetes-ratgeber.net/Diabetes-Schwangerschaft/Schwangerschaftsdiabetes-Gestationsdiabetes-28090_2.html>
- DR. EIGLER, Beate. 2013. *www.netdokter.at*. [online]. [Accessed 20 Februar 2016]. Available from World Wide Web: <<http://www.netdokter.at/krankheit/diabetes-mellitus-7447>>
- Entity Framework Tutorial*. 2015. [online]. [Accessed 2 März 2016]. Available from World Wide Web: <<http://www.entityframeworktutorial.net/what-is-entityframework.aspx>>
- Json*. 2016. [online]. [Accessed März Mar 2016]. Available from World Wide Web: <<http://www.json.org/>>
- MAG. TIEFENBÖCK, Waltraut. 2014. *netDoktor*. [online]. [Accessed 23 Mar 2016]. Available from World Wide Web: <<http://www.netdokter.at/>>
- MERCK GESELLSCHAFT MBH. 2016. *Patientenratgeber Diabetes*. Wien: MerckSerono.
- MSC. QUIRGST, Helga, Ingrid OA DR. SCHÜTZ- FURHRMAN, and Kerstin DR.MED. LEHEMAYR. 2015. *netDoktor*. [online]. [Accessed 23 Mar 2016]. Available from World Wide Web: <<http://www.netdokter.at/>>
- MySql Connector*. 2016. [online]. [Accessed 22 Februar 2016]. Available from World Wide Web: <<http://dev.mysql.com/downloads/connector/>>
- MySql Tutorial*. 2016. [online]. [Accessed 23 Februar 2016]. Available from World Wide Web: <<http://www.mysqltutorial.org/install-mysql/>>
- Netzwelt MySql*. 2011. [online]. [Accessed 20 Februar 2016]. Available from World Wide Web: <<http://www.netzwelt.de/news/86545-mysql-5-5-installation-freien-datenbank-windows.html>>
- PFEIFFER, Thomas. *Gründerszene*. [online]. [Accessed 03 März 2016]. Available from World Wide Web: <<http://www.gruenderszene.de/lexikon/begriffe/html>>
- Pflegeassistentz*. 2011. Stuttgart: Georg Thieme Verlag KG.

- Pilcrow*. 2012. [online]. [Accessed 2 März 2016]. Available from World Wide Web: <<http://thepilcrow.net/explaining-basic-concepts-git-and-github/>>
- Planio*. 2013. [online]. [Accessed 2 März 2016]. Available from World Wide Web: <http://web.archive.org/web/*/https://plan.io/de/smart-git/>
- SCHWAB, Felix and Ingrid SCHWAB-MATKOVITS. 2013. *Systemplanung und Proektentwicklung*. Wien: MANZ.
- SearchDataCenter*. 2014. [online]. [Accessed 2 März 2016]. Available from World Wide Web: <<http://www.searchdatacenter.de/definition/Windows-Server-2012-2012-R2>>
- SearchWinDevelopment*. 2016. [online]. [Accessed 17 Februar 2016]. Available from World Wide Web: <<http://searchwindevelopment.techtarget.com/definition/Visual-Studio-NET>>
- Tech Microsoft*. 2016. [online]. [Accessed 27 Februar 2016]. Available from World Wide Web: <<https://technet.microsoft.com/de-de/library/hh831725.aspx>>
- TechTarget LINQ*. 2008. [online]. [Accessed 27 Februar 2016]. Available from World Wide Web: <<http://searchwindevelopment.techtarget.com/definition/LINQ>>
- Tutorialspoint*. 2016. [online]. [Accessed 29 Februar 2016]. Available from World Wide Web: <<http://www.tutorialspoint.com/asp.net/>>
- Tutorialspoint*. 2016. [online]. [Accessed 22 Mar 2016]. Available from World Wide Web: <http://www.tutorialspoint.com/linq/linq_overview.htm>
- Wikipedia*. 2015. [online]. [Accessed 03 März 2016]. Available from World Wide Web: <https://de.wikipedia.org/Cascading_Style_Sheets>
- Wikipedia JQuery*. 2016. [online]. [Accessed 1 März 2016]. Available from World Wide Web: <<https://de.wikipedia.org/wiki/JQuery>>
- Wikipedia MySql*. 2016. [online]. [Accessed 2 März 2016]. Available from World Wide Web: <<https://de.wikipedia.org/wiki/MySQL>>
- Youtube IIS Server*. 2012. [online]. [Accessed 27 Februar 2016]. Available from World Wide Web: <https://www.youtube.com/watch?v=eU-VVggY_Vs>

13.2. **Abbildungsverzeichnis**

Abbildung 1: Statistik Diabetes weltweit	10
Abbildung 2: Insulinpumpe	22
Abbildung 3: Insulinpen	23
Abbildung 4: ASP.NET Funktionsweise	27
Abbildung 5: 3 Szenarien.....	28
Abbildung 6: LINQ Architektur	29
Abbildung 7: LINQ Syntax.....	29
Abbildung 8: JSON Object.....	31
Abbildung 9: JSON Array	32
Abbildung 10: JSON Value	33
Abbildung 11: JSON String.....	33
Abbildung 12: JSON Number.....	34
Abbildung 13: Startseite	35
Abbildung 14: Login.....	36
Abbildung 15: Über Mich.....	38
Abbildung 16: Berechnung der Insulinwerte.....	39
Abbildung 17: Information	40
Abbildung 18: Lebensmittelliste Gruppe.....	41
Abbildung 19: Lebensmittelliste ‚Gemüse‘	42
Abbildung 20: Lebensmittel Detail.....	42
Abbildung 21: Statistik	43
Abbildung 22: Use-Case Diagramm.....	45
Abbildung 23: Projektstrukturplan	49
Abbildung 24: Model-View-Controller	56
Abbildung 25: MySql Users and Privileges.....	57
Abbildung 26: connectionString	58
Abbildung 27: Anmeldedaten überprüfen und Setzung der Session	59
Abbildung 28: Überprüfung der Passwörter	60
Abbildung 29: HTML Tag RadioButton	61
Abbildung 30: HTML Tag – Anzeige eines Bildes aus einer Datenbank	63
Abbildung 31: Handler definieren	63
Abbildung 32: IgnoreRoute eintragen	63

Abbildung 33: Übergabe des Primary Keys von der Lebensmittelgruppe	64
Abbildung 34: Image in byte[] umwandeln	66
Abbildung 35: ActionLink Lebensmittel auswählen	67
Abbildung 36: Datenbankmodell	70
Abbildung 37: UML - Klassendiagramm Package Data	74
Abbildung 38: UML - Klassendiagramm Package Web	75
Abbildung 39: Teddybär	79
Abbildung 40: Blutzuckermessgerät „Bayer“	87
Abbildung 41: Tamara Holzer	90
Abbildung 42: Petra Enengl	90
Abbildung 43: Patrick Mittermayr	90