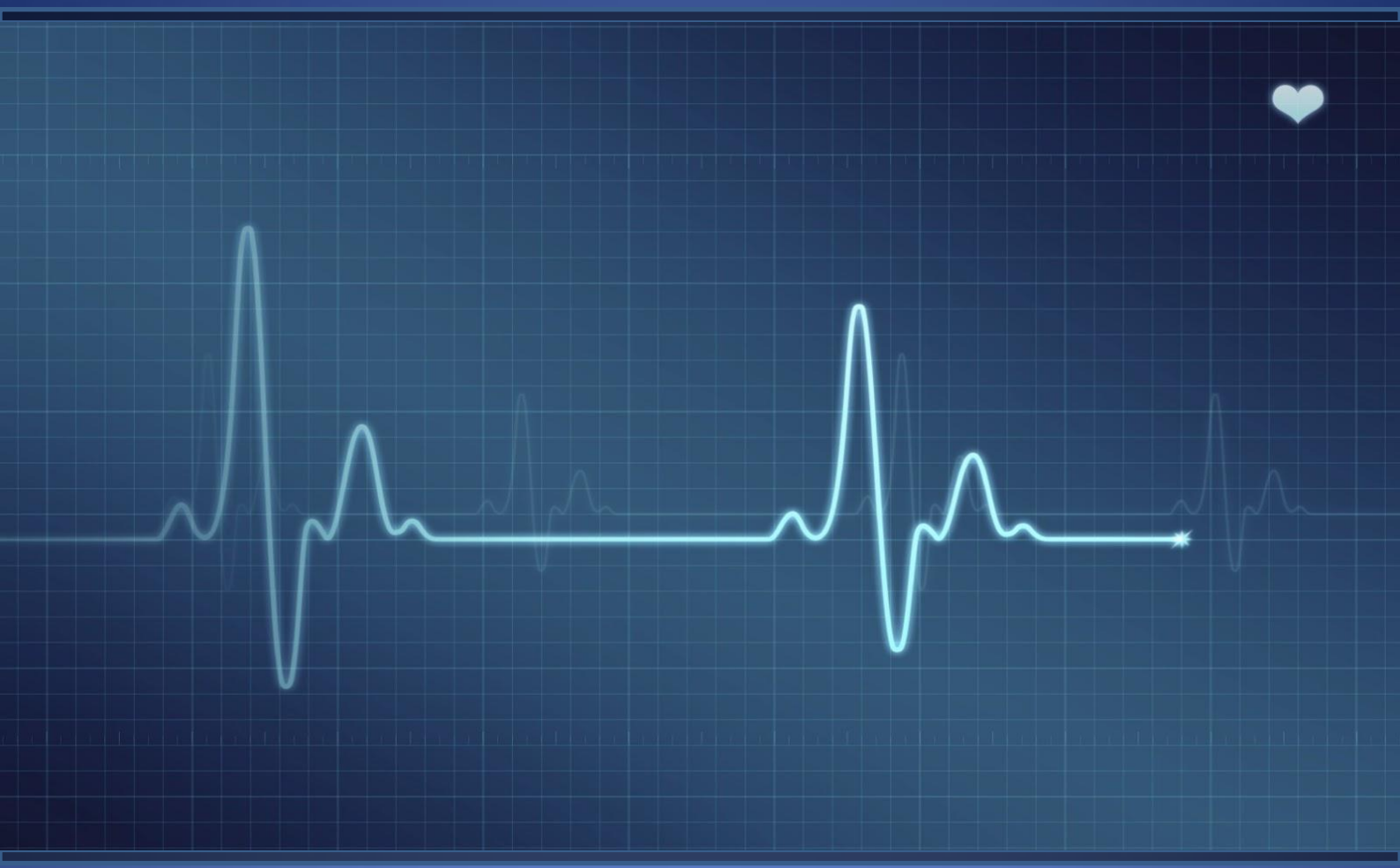

Model based Hypoglycaemia Detection in Patients with Type 1 Diabetes using Electrocardiogram monitoring

Juri Osmolovski

Biomedical Engineering and Informatics
10th semester spring 2013
Aalborg University

Department of Health
Science and Technology
Aalborg University
Fredrik Bajers Vej 7 D2
Telefon +45 99 40 99 40
Fax +45 98 15 40 08
http://www.hst.aau.dk/

Title: Model based Hypoglycaemia Detection in Patients with Type 1 Diabetes using Electrocardiogram monitoring

Project period: ST9-ST10, autumn 2012-spring 2013

Group: G977

Author:

Juri Osmolovski

Supervisor : Ole K. Hejlesen
Morten Hasselstrøm Jensen

Copies: 5

Pages: 85

Deadline: 13th of May 2013

Abstract:

Diabetes mellitus is a global disease with a worldwide prevalence of 366 million people (6% of the world's population). Diabetes is mainly treated with insulin injections; however this introduces the risk of hypoglycaemia. If an oncoming hypoglycaemia can be predicted the people with diabetes will be able to avert it and be more confident with an aggressive insulin treatment. The present study analyses the changes in the electrocardiogram (ECG) and constructs a model that can be used to detect hypoglycaemia based on the ECG.

The data set of this study contains two episodes of insulin induced hypoglycaemia (experiment visits) and two control visits for 8 subjects with Type 1 Diabetes. During control visits the same amount of insulin was injected as during experiment visits, however glucose was injected to prevent BG from falling.

For this study two submodels were constructed using logistic regression: model 1 and model 2. Model 1 can detect insulin induced hypoglycaemia. Model 2 was trained to see difference between an actual hypoglycaemia and merely the effect of insulin. The final model was a combination of the 2 submodels. The final model performance was found to differ depending on what visits were used to train the model on and what visits were used to test the model on, suggesting that the model was overfitting. For the model to be clinically relevant more experiment and control visits need to be included.

For this study an application was developed. The aim with developing an application was to make the model accessible to the people with diabetes. The application was designed as a hypoglycaemia alarm for people with diabetes. Additionally in case of severe hypoglycaemia the developed application includes guides that might aid the person trying to help the person with diabetes.

Abstrakt

Diabetes mellitus (diabetes) er en global sygdom med 366 mio. sygdomsramte på verdensplan, svarende til 6% af jordens befolkning. Det er estimeret at der i år 2030 vil være 439 mio. mennesker med diabetes. Antallet af sygdomstilfælde er stigende i Danmark, såvel som i resten af verden. Der er p.t. 301.500 mennesker med diagnosen diabetes i Danmark og dette antal forventes at blive dobbelt så stor i år 2025. I 2030 var de estimerede omkostninger til forebyggelse og behandling af diabetes 376 mia. US. Dette beløb svarer til 11,6 % af de globale sundhedsomkostinger. I år 2030 disse omkostninger forventes at blive højere en 490 mia US.I dette beløb er ikke indregnet de indirekte omkostninger som sygdommen medfører, f.eks nedsat arbejdsevne. Hvis disse omkostninger medregnes, bliver det samlede beløb betydeligt højere.

Hovedsymptom på diabetes er høj blod glukose (hyperglykæmi), som skyldes ingen eller utilstrækkelig insulin produktion. Det er påvist at kronisk hyperglykæmi hos mennesker med diabetes giver øget risiko for at udvikle bestemte sygdomme. For at kontrollere blod glukose og forebygge hyperglykæmi, modtager mennesker med diabetes insulinbehandling. Insulin behandling øger dog risikoen for lav blod glukose (hypoglykæmi). En af de største udfordringer for effektiv kontrol af blodsukkeret er frygten for hypoglykæmi hos de sygdomsramte. Hvis et kommende hypoglykæmi kan forudsiges vil mennesker med diabetes være i stand til at afværge den, og være mere sikker med en agressiv insulin behandling.

Hypoglykæmi har en kendt effekt på elektrokardiogrammet I dette studie analyseres ændringerne i elektrokardiogrammet og der udarbejdes en model som kan detektere hypoglykæmi ud fra ændringer i elektrokardiogrammet.

Datasættet anvendt til denne studie indeholdt to episoder af insulin induceret hypoglykæmi (eksperiment besøg) og to kontrolbesøg for 8 personer med Type 1 Diabetes. Ved kontrolbesøg blev den samme mængde insulin givet som ved experiment besøg, dog blev der givet glukose for at forhindre blodsukkeret i at falde. Elektrokardiogram og blodglukose blev målt ved alle besøg.

Til dette studie blev der lavet to patient specifikke matematiske modeller baseret på logistisk regression: model 1 og model 2. Model 1 blev trænet til at detektere insulin induceret hypoglykæmi. Model 2 blev trænet til at se forskel mellem en faktisk hypoglykæmi og blot virkningen af insulin. Det endelige model bestod af to submodeller sat i serie (begge modeller skal detektere hypoglylæmi).Den endelige models ydeevne fandtes at variere afhængigt af, hvilket besøg blev brugt til at træne modellen på,hvilket tyder på, at modellen blev overfittet. Før modellen kan være klinisk signifikant flere kontrol og eksperiment besøg skal inkluderes i træningssættet for at undgå overfitting.

I tillæg til den matematiske model blev der udviklet en applikation. Formålet applikationen var at lave modellen lettilgængelig for folk med diabetes. Applikationen er designet som en hypoglykæmi alarm. Derudover i tilfælde af en svær hypoglykæmi applikationen kan hjælpe den person, der forsøger at hjælpe personen med diabetes, ved at vise en aktionguide.

Preface

This report was developed by Juri Osmolovski as a part of the 9th-10th semester master project for Biomedical Engineering and Informatics at Aalborg University. This report was written in the period from September 1st 2012 to May 13th 2013. This report is aimed at fellow students, researchers and people with interest in diabetes mellitus.

The author would like to give thanks to the contributors:

Ole K. Hejlesen, Professor with special responsibilities Aalborg University,
Morten Hasselstrøm Jensen, PhD fellow Aalborg University,
Toke Folke Christensen, Clinical Pharmacology Scientist Novo Nordisk.

And a special thanks to Novo Scholarship Programme for financial support.

Reading guide
Bibliography can be found in the back of the main report, with references ordered by the order of appearance. Graphs, tables and listings are numbered according to the chapter, e.g the first figure in chapter 1 is numbered figure 1.1. The first time an abbreviation is used the word will be explained. Afterwards only the abbreviation will be mentioned. The developed application and MATLAB code can be found on a CD attached to this report. The report is divided into 4 parts:

- Part I: Problem analysis
- Part II: Problem solving
- Part III: Synthesis
- Part IV: Appendix

Contents

1 Introduction

Diabetes mellitus is a global disease with a world-wide prevalence of 366 million people, corresponding to 6% of the world's population [1]. Furthermore, it is estimated that by 2030, 439 million people will have the disease [1]. The incidence of diabetes is increasing in the world as well as in Denmark. Currently there are 301 500 people that were diagnosed with diabetes in Denamrk, by the year 2025 this number is expected to double [2].

In 2010 the global costs due to treatment and prevention of diabetes were estimated to 376 billion US$. This corresponds to 11.6% of the total healthcare expenditures worldwide. By the year 2030 the total expenditures are expected to exceed 490 billion US $. This number, however, does not include the indirect economic costs that the disease imposes e.g: reduced ability to work. If such effects were included, the total costs would be much higher [3].

The increasing prevalence of the disease and the growing economic burden mean that actions have to be taken to minimize the effect of the disease on the every day life of people with diabetes. Such actions will in turn lower the economic burden as people with diabetes will be able to live and work normally.

Since the discovery of insulin in 1922, diabetes was transformed from a mortal disease to a disease that could be managed with insulin injections. Since then, diabetes treatment was constantly being improved by new insulin types and developments in IT technology. The cornerstone of diabetes control remains keeping the blood glucose (BG) as low as possible to avoid late diabetic complications. However, this introduces a new danger in form of hypoglycaemia, which forces the patients to balance between too low and too high BG. One of the main problems in effective BG control is fear of hypoglycaemia in people with diabetes [4].

If hypoglycaemic episodes can be predicted in time, patients will be able to avoid critical situations, providing patients with confidence to use a more aggressive insulin therapy.

Initial problem statement:
How can hypoglycaemia be predicted in people with diabetes?

Part I

Problem Analysis

This part will start by describing diabetes and the current treatment possibilities. Then the link between the changes in BG and the heart will be described. Followed by a section addressing the growing importance of IT applications in diabetes treatment. Finally the problem analysis is concluded with a problem statement.

1.1 Diabetes

Diabetes mellitus is characterized by insufficient insulin production and/or insulin resistance [5]. Insulin is a hormone that promotes the uptake of glucose in liver, muscle and fat tissue. Insulin is produced in Islets of Langerhals by the pancreatic beta cells. Most of body's cells need insulin in order to absorb glucose, these cells are called insulin dependent [6]. Insulin can lower the blood glucose levels by increasing the rate of glucose uptake in insulin dependent cells [6]. The lack of insulin or insulin resistance results in elevated blood glucose concentration, hyperglycaemia, which is the main symptom of diabetes. Diabetes mellitus is a condition that is generally divided into two subcategories namely Type 1 diabetes (T1D) and Type 2 diabetes (T2D).

1.1.1 Type 1 diabetes

T1D appears as insufficient or total absence of insulin production, which leads to hyperglycaemia. The lack of insulin production is caused by an autoimmune response where pancreatic beta cells in the Islets of Langerhans are attacked and destroyed by the body's own immune system [7], resulting in halted or insufficient insulin production [5]. Before the condition is diagnosed the mass of insulin producing beta cells may have already been reduced to 10% of the normal mass, as a result of disease progression for several years [7]. Therefore, people with T1D are dependent on daily insulin injections. Without insulin people with T1D will die from ketoacidosis within a short time. The pathogenesis of T1D is still largely unknown [5]. However some researchers suggest that T1D might be caused by a combination of genetic and environmental factors [7]. It is known that people who suffer from T2D or other autoimmune diseases have a greater risk of developing T1D. In addition family history of T1D increases the risk of developing the disease [5]. The latter can be shown by examining the T1D incidence rate. The overall risk for the Danish population is 0.4% while risk among twins is 30-70% [7]. Several studies have found that occurrence of some specific genes, may increase the risk of developing T1D, while other genes may decrease the risk of T1D [5].

1.1.2 Type 2 diabetes

T2D is caused by insulin resistance in insulin-dependent cells, often combined with decreased ability to produce insulin, leading to hyperglycaemia. Insulin resistance means that insulin dependent cells have a decreased sensitivity to insulin and thereby a higher insulin concentration

is needed to promote glucose uptake in insulin dependent cells [7]. Unlike Type 1 diabetes, insulin resistance is a problem with the cells that respond to insulin rather than a problem with the production of insulin. However insulin resistance in T2D is often combined with a defect in the pancreas insulin production [8]. T2D is the most common diabetes type. People with T2D, in contrast to people with T1D, retain a certain production of insulin. The amount of produced insulin, however, is insufficient to keep the blood glucose within a normal range. The progression rate of the disease is slow, resulting in late discovery. Treatment is divided into substages, starting with oral medications that increase insulin sensitivity, later as the disease progresses it is treated with insulin injections. The prevalence of T2D increases with age with a cumulative incidence rate of 11 % by age 70. Several risk factors for developing T2D were identified, most important are obesity and a family history of disease. [7] [5]

People with diabetes have a high risk of developing long term complications.

1.2 Long time effects of hyperglycaemia

Severe hyperglycaemia is a deadly condition that must be treated. Moreover long time effects of chronic mild hyperglycaemia are well known and documented. It has been shown that chronically elevated BG levels in people with diabetes greatly increase the risks of developing certain high BG related conditions and diseases[5]. Some of the most common are:

- Cardiovascular diseases: People with diabetes have an increased risk of developing atherosclerosis, which is a complication that is the most frequent cause of death among people with diabetes. Arteriosclerosis in people with diabetes is caused by abnormal metabolic regulations [7]. These abnormalities include chronic hyperglycaemia and insulin resistance. [9]
- Eye diseases: Diabetes is associated with an increased risk of developing a number of eye diseases. The most common is diabetic retinopathy. This condition ultimately causes blindness. [5]
- Diabetic neuropathy: A serious complication that appears as destruction of sensory, motor, and autonomous nervous system. Somatic neuropathy, i.e. destruction of the sensory and motor nervous system is the most common neuropathy among people with diabetes. Diabetic neuropathy is mainly caused by hyperglycaemia. [5]
- Diabetic nephropathy: Diabetic nephropathy is a kidney disease which appears as destruction of the kidney tissue. Chronic hyperglycaemia increases the workload on the kidneys, and increases the risks of developing diabetic nephropathy. [5]

1.3 Treatment of diabetes

In order to avoid the long term complications associated with hyperglycaemia a treatment is needed. The intake of a carbohydrate rich meal will increase the blood sugar levels, as the

carbohydrates are broken into glucose molecules. When blood glucose levels rise, in a healthy individual, insulin is released in response. The graph 1.1 shows this relationship.

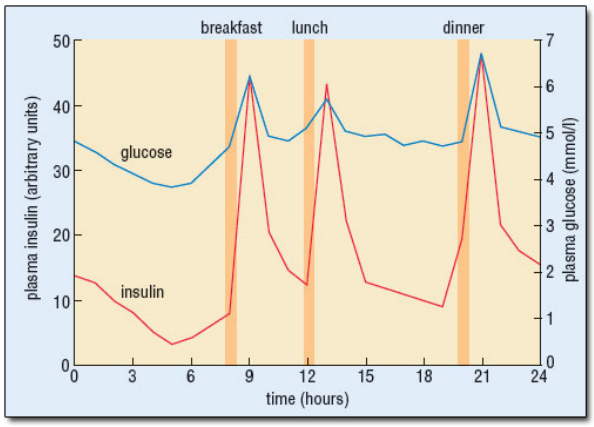

Figure 1.1: Insulin and glucose levels during the day in a healthy person.

In a person with diabetes the insulin is not produced at all, produced in insufficient amounts, or the cells are not sensitive to insulin, so it must be administered. The aim of diabetes treatment is to maintain BG within the normal range of a healthy person (4-7.8mmol/l), by mimicking the insulin secretion of healthy people with insulin injections [6]. However, the administered insulin is absorbed gradually, hence sharp rises and falls in its levels as seen in the healthy individuals are hard to obtain. In addition in a healthy person insulin level never falls to 0, instead a basal level is always maintained. In order to recreate normal insulin levels in a person with diabetes and compensate for the food intake it is necessary to have multiple injections over the course of the day. Different types of insulin are used at different situations, short acting insulin is used to compensate for a meal, long acting insulin is used to maintain basal insulin level between the meals and during the night. The insulin level between the meals and at night is called basal insulin. Insulin used at meal time is called prandial insulin. [10] [11]

Insulin is usually injected with an insulin pen. Another way to inject insulin is by using an insulin pump. The pump is attached with a small needle subcutaneously. It is therefore also called a continuous subcutaneous insulin infusion.

The evidence suggests that insulin therapy reduces the risks of long term complications associated with diabetes, by avoiding chronic hyperglycaemia [12]. Effectiveness of insulin treatment is measured using the percentage of glycosylated haemoglobin, HbA1c. HbA1C is a measure of the amount of haemoglobin, which was glucated due to exposure to plasma glucose. It is used to identify the average plasma glucose concentration over a prolonged period of 2-3 months which indicates the average BG over this period. There is a close link between HbA1c and the risk of developing long term complications such as retinopathy, nephropathy, neuropathy and atherosclerosis [11]. The American Diabetes Association recommends a target level of HbA1c of < 7 %, however this level of HbA1c is not achieved in all patients. The

problem is that keeping a low HbA1c increases the risk of hypoglycaemia (abnormally low blood glucose consentraion).[11] [7]

1.4 Hypoglycaemia

Hypoglycaemia is the main obstacle in effective glycaemic control. To prevent long term effects of diabetes, the goal of treatment is to keep BG as low as possible, but this strategy increases risk of hypoglycaemic episodes. People with diabetes have to adjust their life in order to minimize the negative effects that the disease imposes. For instance when people with diabetes consume alcohol, it can cover up the symptoms of hypoglycaemia, as begin drunk and having hypoglycaemia makes the individual act in a similar way. Other factors such as skipped meal, bad timing of insulin injection or too high insulin dose are all frequent causes of hypoglycaemia. Mild hypoglycaemia can be averted by simply drinking a juice. During a severe hypoglycaemia event person may be unconscious or unable to swallow so a glucagon injection has to be applied by a trained helper in order to restore glucose concentration to normal values. As a result fear of hypoglycemia is high among people with diabetes and their families. The dilemma faced by people with diabetes is that low HbA1c will reduce the risk of developing long term complications, on the other hand low HbA1c increases the risk of hypoglycaemia.

1.4.1 Definition of hypoglycaemia

There is no uniform definition of hypoglycaemia. The uncertainty in definition is caused by the variation and subjectiveness of assessment of symptoms in different patients. Likewise the BG concentration at which hypoglycaemia symptoms manifest has a large inter-person as well as intra-person variability, depending on several factors such as duration of diabetes and glycaemic control. [13]

Based on [6], [14], this report defines glycaemic ranges as:

Severe hypoglycemia: *<2.2mmol/L*
Mild hypoglycemia: *2.2-3.9mmol/L*
Normoglycaemia (normal BG range): *4.0-7.8mmol/L*
Hypoglycemia can be avoided by simply drinking a juice. However detection of oncoming hypoglycaemia is made difficult by the patient variability in symptoms and inability to measure actual plasma glucose concentration continuously.

1.5 Response to Hypoglycaemia

A fall in blood glucose normally causes a chain of reactions illustrated on 1.2. These reactions are:

- Reduction of insulin level in plasma due to reduction of insulin secretion by pancreatic B cells. Occurs during normal decrease in glucose levels [15].
- Increase of glucagon and epinephrine secretion when glucose level falls just below the normal lower limit[15].
- Increase in secretion of cortisol and growth hormone. This occurs when glucose levels are severely decreased [15].

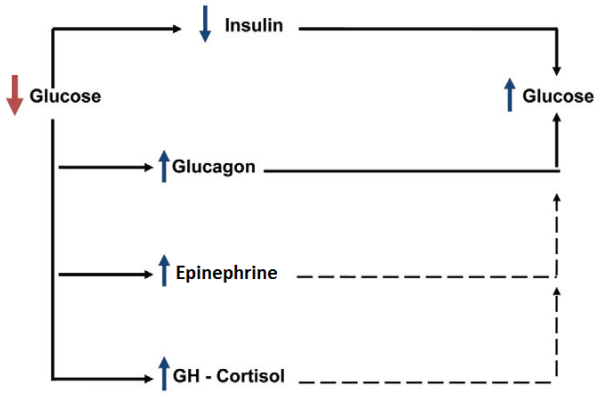

Figure 1.2: Mechanisms of Hypoglycaemia Counterregulation in a healthy individual

The aim of these processes is increase of BG concentration to normal levels in order to protect the brain, that is glucose dependent. If the blood glucose levels fall sufficiently low neuroglycopenic (shortage of glucose in the brain) symptoms occur. Such as disorders of mental functions including: difficulty concentrating, drowsiness, and incoordination [16] [17]. These symptoms warn people that immediate action is required to treat the hypoglycaemia and restore blood glucose to normal.

The threshold values at which these processes occur in **healthy** individuals are known and are reproducible. The inhibition of insulin secretion occurs approximately at 4.4 mmol/l, the glucagon secretion at 3.8 mmol/l, epinephrine secretion at 3.7 mmol/l, and disorder of mental function at 2.5 mmol/l [15].

In diabetic patients, this chain of reactions is defect, some counter regulatory processes do not occur, and the onset of symptoms is delayed or hindered. Making it hard for the people with diabetes to detect an oncoming hypoglycaemia [18] [19] [20].

The pathophysiology of glucose counterregulation in T1D is characterized by the loss of the first and second defense mechanisms against hypoglycaemia. First counterregulation mechanism is absent as there is no decrease of insulin in plasma, due to its way of administration, and because of its continuous absorption from the subcutaneous tissue where it was injected. The second counterregulation mechanism is defect as it has been shown that in people with T1D glucagon concentrations cannot increase[15]. The deficient glucagon response seems to be related to the deficiency of insulin secretion [21]. Moreover the epinephrine response, to

the constant glucose decrease, is often deficient [19] [17] [22]. The glycaemic threshold for the epinephrine response is moved to a lower level in people with T1D [17]. Defective glucose counterregulation syndrome is illustrated on 1.3. Hypoglycaemia is most common in people with T1D.

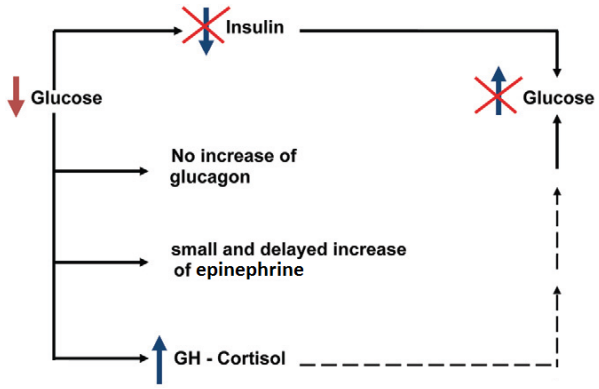

Figure 1.3: Defect Mechanisms of Hypoglycemia Counterregulation in people with diabetes

The mechanism of glucose counterregulation in T2D remains intact during first years of disease. This explains the low number of hypoglycaemia events during this period of disease. Clinical research showed that hypoglycaemia becomes more frequent in people with T2D as the disease progresses. Patients who had T2D for over 15 years present the same deficient counterregulation as seen in people with T1D. [20] [19] [23]

1.6 Hypoglycaemia and cardiovascular system

Acute hypoglycaemia provokes physiological responses affecting cardiovascular system. These responses are produced by the sympatho-adrenal activation and counter regulatory hormone secretion. The goal of these changes is to maintain the supply of glucose to the brain, protecting it from neuroglycopenia (shortage of glucose in brain), and promote glucose production in liver (gluconeogenesis). Blood flow is therefore increased to the myocardium, the splanchnic circulation (to provide precursors of gluconeogenesis to the liver e.g lactate), and the brain. This is achieved by altering regional blood flow; bloodflow to the organs that are not important for counterregulation, such as spleen and skin is decreased. [24] [25] [26] [27] [28]

1.6.1 Physiological changes in the heart

Autonomic activation of the sympathoadrenal system, results in a release of epinephrine which provokes haemodynamic changes. The hemodynamic changes associated with hypoglycaemia include an increase in heart rate, an increase in blood pressure, reduced arterial resistance,

and increased cardiac output [29]. The workload of the heart is therefore markedly increased. This effect promotes the transport of materials to liver and hence gluconeogenesis.

Hypoglycaemia has long been known to affect the electrocardiogram (ECG) [30]. Both insulin-induced and spontaneous clinical hypoglycaemic episodes prolong cardiac repolarization [31]. These changes are reflected by changes in the T wave of the electrocardiogram. During hypoglycaemia the T waves are often flattered [32]. Other changes in ECG have been observed including Q-T interval prolongation [33] and S-T segment depression [34]. Activation of the sympathoadrenal system is thought to be the main factor provoking these changes [31].

Electrophysiological changes are partially related to hypokalemia. It was shown that hypoglycaemia induces a fall in serum potassium via epinephrine release and a direct effect of insulin [35]. Insulin and epinephrine stimulate NA+/K+ ATPase function that results in a fall of potassium concentration in extracellular fluid leading to hypopotassemia [6]. A fall in extracellular potassium concentrations alters the cell's ionic balance and prolongs repolarization rate, illustrated by the flattered T wave. However some studies reported that the changes in potassium concentration do not seem to be major determinants of cardiac repolarization abnormalities during hypoglycaemia [36]. Implying that hypoglycaemia as such has an effect on the repolarization rate.

The changes in the ECG reported by earlier studies are summed in table: 1.1.

Haemodynamic Changes	Response	Cause
Heart Rate	Increase	Epinephrine
ECG change	T-wave flattering	Epinephrine, Hypokalemia
ECG change	QT-segment prolongation	Epinephrine, Hypokalemia
ECG change	ST-segment depression	Epinephrine, Hypokalemia

Table 1.1: Summary of heart changes in response to hypoglycaemia.

1.7 Role of Diabetes Applications

Diabetes management is a time consuming and a difficult task. The developments in IT technology are aimed at making it easier for people with diabetes to manage the disease. Currently the setup of a typical BG monitoring system consists of an insulin pump and a glucose meter. The data between the insulin pump and the glucose meter is typically synced by e.g use of bluetooth. However, this means that two devices have to be carried at all times. Some companies such as "iBG Star" are attempting to exclude the glucose meter as a separate device. The idea of their solution lies in the fact that many people already carry a very capable computer with them, namely a smartphone. They developed a glucose meter that is very small and directly connects to the smartphone. The iBG glucose meter docked with a smartphone is shown on figure 1.4. Such solution seems natural, as carrying less devices is always preferable.

The application allows to view and manipulate the BG data, and allows the user to improve the diabetes treatment.

A typical hypoglycaemia detection system uses a continuous glucose monitor (CGM) together with glucose meter and an insulin pump to manage and monitor the glucose levels. A smartphone to the authors knowledge was never included as the main operating unit in such systems.

In present study the idea is to use the ECG signal which is easy to measure and relay it to the mobile phone, where the signal is filtered processed and analyzed for hypoglycaemia. The practical improvement from current hypoglycaemia detection setup is the idea that the person with diabetes only needs the mobile phone and a set of ECG sensors for the system to work.

Figure 1.4: iBGStar glucose monitor is connected with a smartphone. A test strip is inserted directly into iBGStar.

1.8 Summary

A link between changes in the ECG signal and hypoglycaemia was reported by many studies. Typically, the ECG feature extraction was done by hand, which imposed some limitations. For example, it was impossible to extract a large number of ECG features for mathematical analysis. The decision to include or not include ECG features was based on the researcher's prior knowledge. This means that any connections that were thought not to be highly significant were not included in the analysis of the previous studies. Implying the possibility that significant connections were overlooked. The benefiter of an ECG based hypoglycaemia detection model is the person with diabetes. To make any modeling results available for the people with diabetes, a mobile application that will use the model to alert the person with diabetes of an oncoming hypoglycaemia has to be developed.

Problem statement:
How can an algorithm for prediction of hypoglycaemia based on the ECG signal be developed and implemented in a mobile application?

Part II

Problem solving

2

Problem solving strategy

In order to answer the problem statement three aims were identified. The idea behind dividing the study into different phases was based on the fact that very little was known about the information within the accessible data set. Therefore before any complicated modeling techniques were employed an initial analysis was performed to investigate whether there is any evidence of a change in ECG features during hypoglycaemia. The second aim of the project, given the fact that initial analysis showed that features did change during hypoglycaemia, was to create a mathematical model that could use the ECG features to predict hypoglycaemia. The third aim was introduced in order to make any results obtained during modeling available for the people with diabetes by developing a mobile application.

The problem solving part of the project will start with a description of the data that was available for present study, followed by three chapters each describing methods used to solve one of the study aims. The aims of present study are listed below:

- **Aim 1:** identify statistically significant changes in the ECG features during insulin induced hypoglycaemia
- **Aim 2:** investigate modeling techniques that can be used to detect hypoglycaemia based on ECG features
- **Aim 3:** develop a mobile application that can be used by the person with diabetes for detection of hypoglycaemia

3 Data description

This study used part of the data collected in a case-control study at Steno Diabetes Center by Toke Folke Christensen and Lise Tarnow. The used data contain information on ten male subjects with diabetes Type 1 who were followed during insulin induced hypoglycaemia(BG below 3.9 mmol/l). The data contain information on two events of insulin induced hypoglycaemia an two control visits for each patient, during which BG and ECG were measured. A typical ECG signal can be found on 3.2.

Each patient went through 2 sessions, each containing an experiment and a control visit. Control visits will later be referred to as V2 and V4. Experiment visits will be referred to as V1 and V3. Session 1 is defined as V1 and V2. Session 2 is defined as V3 and V4.

The data collection process during experiment visits (V1,V3) was identical for all patients. Upon patient's arrival at 8.00 BG was measured, then BG was measured (from the ear) every 30 minutes until start of insulin injections at 10.00. Hereafter as the BG levels started to fall BG was measured every 10 minutes, or more frequently. Insulin injections continued until the BG fell below 2.5 mmol/l, where injections ceased and juice was given to restore the BG levels. After the BG levels started to rise BG was measured every 10 minutes or more frequently until BG levels exceeded 3.9mmol/l, then BG was measured every 30 minutes. ECG was measured during the entire procedure 12-derivative ECG was measured. Figure 3.3 shows a typical ECG signal measured using 12 different leads.

The control visits (V2,V4) followed the same procedure as experiment visits, however glucose was being continuously injected to prevent hypoglycaemia and keep the patient's in a BG range of 5 to 10 mmol/l.

Data collected from patient 2, 9 and 10 were not accessible. Data from patient 3 and 7 only contained one visit and was not included in present study. Data from patients 1, 4, 5, 6 and 8 was fully accessible.

Patient exclusion:
Patient 1 was 61 years old at the time of the experiment and was taking Akarin for depression treatment. The daily dosage was 20 mg. As reported by the Food and Drug Administration [37] Akarin may cause abnormal heart rhythms. The prevalence of this side effect was found to increase with the patients age and found to be the highest in patients over 60 years old. Additionally the FDA recommends not to use Akarin in patients with hypoglycaemia as the risk of abnormal heart rhythm is increased in these patients. [38]

In present study all patients including patient 1 were injected with insulin and a hypoglycaemia was induced. The ECG is thought to change in response to hypoglycaemia. However in patient 1 sideeffects of Akarin might disturb these changes. The risk of sideeffects in patient 1 is high as the age is over 60 and as described in section 1.6.1 insulin induced hypoglycaemia might result in hypoglycaemia, which inturn further increases risk of heart rhythm related sideeffects of Akarin. Because of the high risk of heart rhythm related side effects it was chosen to exclude patient 1 from this study.

BG data: BG measurements from all included patients and visits were combined into a single data set. Total number of BG measurements was 295. The maximum time between BG measurements was roughly 30 minutes (low sample rate at high BG values). The minimal time between the two measurements is 10-2 minutes(during hypoglycaemia segments). Meaning that the interesting segments of declining BG were sampled at a much higher rate than segments where BG was high. This is illustrated on the distribution plot of data from all patients 3.1. Number of BG measurements below or equal to 3.9 mmol/l is 93. 202 glucose measurements were above 3.9 mmol/l.

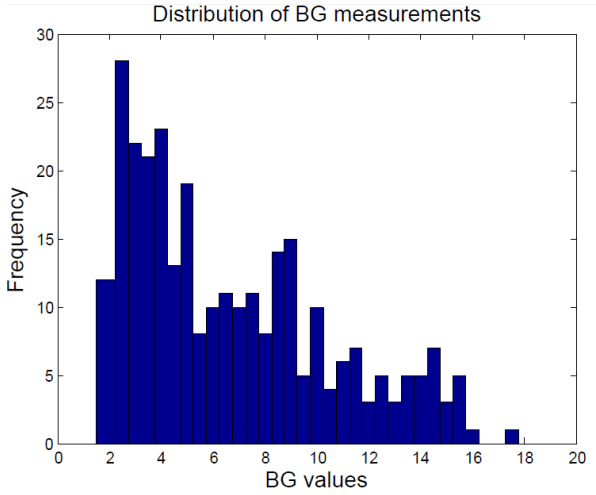

Figure 3.1: Distribution plot of BG measurements, all included patients, all visits

ECG data: The present study has a set of 313 features that were computed based on the original ECG measurements. Intervals between the feature measurements are exactly 10

seconds. The full data set of all accessible patients and visits hold a total of 26748 points for each feature. Many features included solely or above 80% of NaNs. The algorithm used to extract these features was not acessible and information could not be obtained on the nature of missing values(missing at random or not). The features that contained less than 80% NaNs are shown in appendix 10.2. Other features were also included in model creation, however none of them were found to be good predictors.

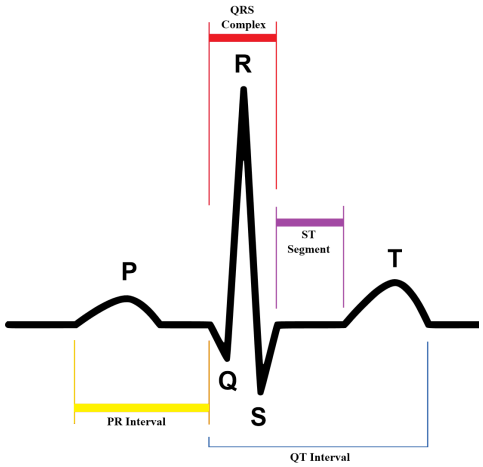

Figure 3.2: Normal ECG signal

[39]

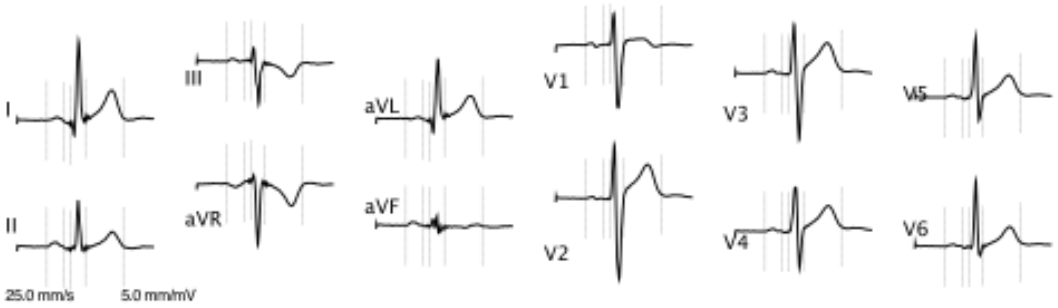

Figure 3.3: 12 ECG leads

[40]

4 Initial Analysis Methods

4.1 Introduction

In this chapter the methods used to achieve first aim of the study, namely to identify statistically significant changes in the ECG features during insulin induced hypoglycaemia, will be described. The goal here was to uncover whether any further attempts to create a model can be justified or whether there is no change in the features and hence a model might not be possible to make. This chapter was written based on [41].

4.2 Data pre-processing

There are fewer BG measurements than ECG measurements available. In order to analyze the link between BG and the ECG features, it was required to link ECG data with BG data. This was done in the following way: ECG measurements +-60 seconds from the time of BG measurement were extracted, then a median value was extracted and used as a feature point at time of BG measurement. The data set used for initial analysis contained 295 BG measurements and 313 ECG features with 295 measurements each.

4.3 Choice of statistical test

To achieve the first aim of this study a statistical test was used. The goal with using a statistical test was to find what features could separate the hypoglycaemia and no hypoglycaemia events. In addition different threshold BG values were used for splitting data into hypoglycaemia and no hypoglycaemia sets. A threshold BG value was identified for each feature that was associated with largest difference in the hypoglycaemia and no hypoglycaemia sets. A variety of different statistic tests exists however given the structure of data, Mood's test was chosen. Mood's test was used based on following:

1. The samples are not matched (one sample is longer than the other).
2. The variances are not equal in hypoglycaemia and no hypoglycaemia samples.
3. The data may contain outliers.

4.4 Mood's Median Test

The Mood's median test is a nonparametric test that is used to test the equality of medians from two or more populations. The test can be used to investigate whether two samples are independent or there exists an association between them. In this study the aim was to investigate whether the ECG signals during hypoglycaemia differ from ECG signals during No hypoglycaemia. Formally the null hypothesis is:

- H0: The population medians in the two samples are equal
- Ha: The population medians are not equal

A significance level a of 5% was chosen, meaning that if p-value was above 0.05 the null hypothesis was not rejected. Suggesting that the difference in medians between the two samples was not statistically significant.

Mood's test has following assumptions:

1. Observations are independent
2. Observations come from populations with continuous distribution functions

A detailed description of Mood's test procedure can be found in appendix 10.3

4.4.1 Implementation

In present study Mood's test was implemented as shown on figure 4.1. A starting threshold value of 2.3 mmol/l was chosen to ensure that a minimum of 10 points would be included in Hypoglycaemia set. Maximum threshold value was chosen based on definition of hypoglycaemia of present study(4 mmol/l). A step size of 0.01 mmol/l was set as low as possible without making the procedure too computationally heavy.

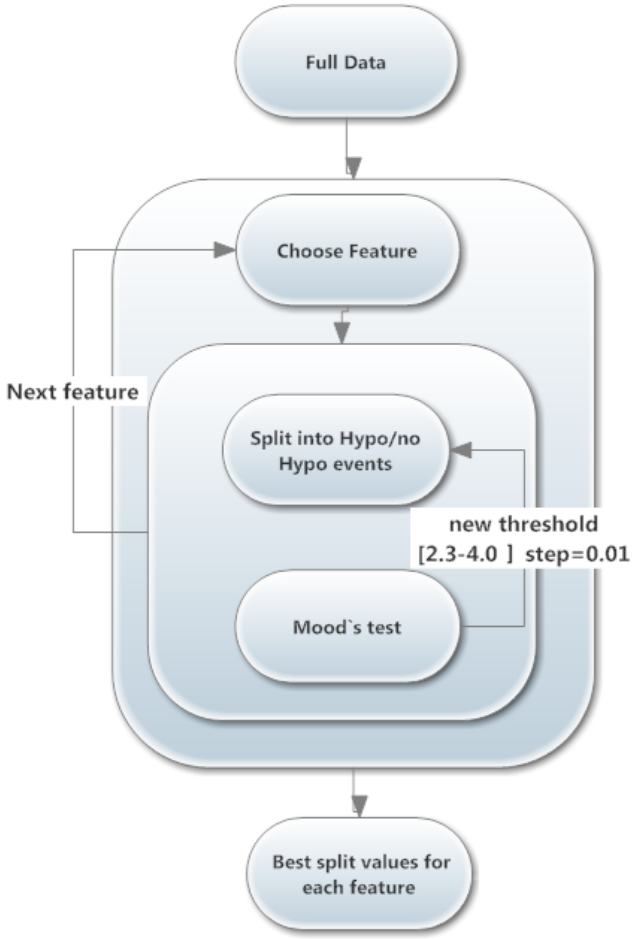

Figure 4.1: Full data set is input to procedure The algorithm starts by picking first feature. Then the feature points are split into Hypoglycaemia/No Hypoglycaemia subsets. First split was made at BG 2.3 mmol/l. Hereafter Mood's test was applied on the two subsets, and a p-value was obtained. Then the data was split using a different threshold and the Mood's test was repeated. The process continued at different thresholds ranging from 2.3 to 4.00 mmol/l with a step size of 0.01. When p-values for all thresholds were computed for a single feature, next feature was chosen and the process was repeated. When all features were analyzed, thresholds that produced lowest p-values for each feature were exported.

5 Model Development methods

5.1 Introduction

The second aim of this study was to investigate modeling techniques that could be used to detect hypoglycaemia based on ECG features. Steps taken to achieve the second aim are described in this chapter.

Patient specific model
For this study it was chosen to create a patient specific model and to limit the model to choosing a single most important feature. The reasoning behind this choice can be found in discussion chapter 8.2.

5.2 Data pre-processing

In this section the methods that were used to pre-process the data before inserting it into the model creation algorithm will be described. The BG was interpolated. The ECG features were filtered and normalized.

5.2.1 Spline interpolation

The dataset contains a large amount of ECG measurements and much fewer BG measurements. In order to fully use the dataset it was chosen to interpolate the BG and thereby obtain a BG measurement for every ECG measurement. For this purpose quadratic spline interpolation was used. Spline interpolation uses low-degree polynomials in each of the intervals, and chooses the polynomial pieces such that they fit smoothly together. The resulting function is called a spline. For this study it was chosen to use quadratic spline interpolation and not

linear interpolation as in liner interpolation the curve becomes discontinuous in the inner points. Linear and quadratic interpolations are shown on figure 5.1. For this study quadratic interpolation was used to interpolate the BG measurements.

When performing an interpolation it is important to keep in mind that interpolation introduces an error that has to be considered. In present study the BG measurements are at most 30 minutes from one another during periods with no hypoglycaemia. During hypoglycaemia the intervals between any two BG measurements are between 10 and 2 minutes. The aim of this study is to detect hypoglycaemia so the model has to clearly distinguish between hypoglycaemia and no hypoglycaemia measurements. For this reason even though the time between each measurement during no hypoglycaemia is large it is possible to interpolate it, as according to the experiment description no hypoglycaemia events were present in those segments. The model is not aimed at detecting a specific value at that range but will use these measurements as no hypoglycaemia measurements. For the hypoglycaemia segments a sample rate of 10-2 samples per minute is arguably the highest sample rate that can be achieved in practice.

A rough estimate of the maximal error in hypoglycaemia segments is based on following: The BG fall and rise rates have a defined value as reported by a study from 2004 [42]. This study included 124 subjects and the purpose of this study was to study rate of change of blood glucose for a diabetic population. The results of this study showed that the rate of change of blood glucose was within +-0.0555 mmol/l/min. The data of current project fit these results. The mean glucose change rate during hypoglycaemia was 0.068 mmol/l/min for the entire patient population. The maximum change rate of glucose was -0.16 mmol/l/min. This hight change rate was only detected between 2 measurements in one of the patients.

For the current study these results mean that if the interval between two BG measurements during hypoglycaemia is 10 minutes and the change rate of glucose is +-0.068 mmol/l/min, then the error can roughly be calculated as follows: if the first BG measurement and the second BG measurement 10 minutes later have same values then the points will be interpolated with a horizontal line with a constant value. In the worst case scenario during this time the BG could have decreased for the first 5 minutes and increased for the last 5 minutes. Meaning that the largest error compared to the interpolated line is achieved at the 5 minute mark. If the change rate was +0.068 for the first 5 minutes and -0.068 for the last 5 minutes than the largest error is $0.068 * 5 = 0.34$. The likelihood of this worst case scenario is discussed in section 8.2.

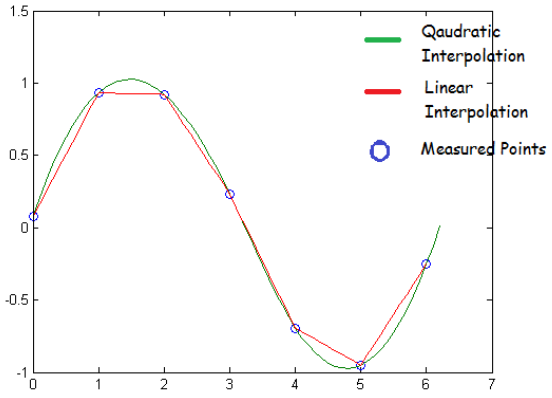

Figure 5.1: Linear interpolation and quadratic spline interpolation of same random measurements

5.2.2 ECG feature filtering

An ECG feature is plotted on figure 5.2. The sharp rises and falls in the feature value are partially caused by the noise which makes detecting changes in the actual feature value much harder. A simple and effective solution for this problem is a moving average filter. As the name implies, the moving average filter operates by extracting average from a number of points from the input signal to produce each point in the output signal. Filtered values of the same ECG feature are shown on figure 5.3. Here a window size of 10 samples was used to compute each point. Meaning that each filtered sample is based on average of the current sample and 9 other samples that came prior to it. Increasing window size will decrease the noise but in return some of the information is lost. It is therefore important to use a window size that effectively removes as much of the noise as possible and at the same time keeps the essential signal information. In addition increasing the window size introduces a delay. In present study time between each ECG measurement is 10 seconds. Meaning that an increase of window size by 1 will increase the delay by 10 seconds.

In present study the optimal window size was determined as follows. The model was created as described in sections 5.3.1 and 5.3 using different window sizes for filtering. The window sizes ranged from 1 (no filtering) to 20. Maximal window size of 20 was chosen based on computational costs and the fact that a larger delay is unwanted. The model was created on train data from a single session and evaluated on resubstitution data from the same session. The misclassification error the model produced was used to evaluate choice of window size. The lower the error the better the chosen window size. The overall error that the model produced using a specific window size was calculated by averaging errors of all patient models using that particular window size. It can be seen on figure 5.4 that the minimal error was achieved when a window of 5 samples was used.

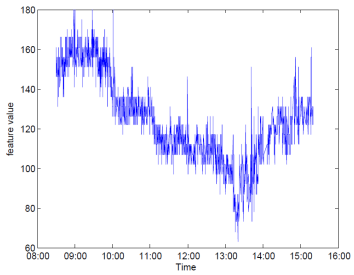

Figure 5.2: Raw feature data

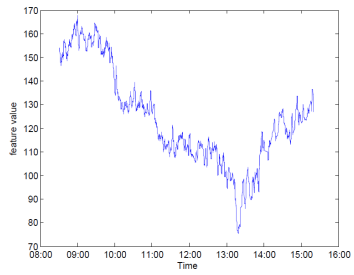

Figure 5.3: Data filtered with moving median filtering using windows size 10.

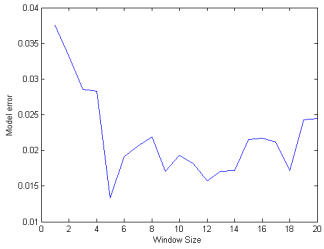

Figure 5.4: System error with different filtering window sizes

5.2.3 Data Normalisation

It was noted that the baseline of the different ECG features prior to the insulin infusion is different from one visit to another. Additionally the peak amplitudes achieved during hypoglycaemia events were also different. Figure 5.5 shows a plot of feature values, measured during visit 1 and visit 3. It can be observed the feature value drops during both visits, however the starting points and the minimal values are different. Assuming that hypoglycaemia occurs during both visits at the lowest point of each chart, a model that uses this information to detect hypoglycaemia can be created. Unfortunately if the original feature values are used such a model will not function. If the model is trained on visit 3 the hypoglycaemia will be associated with a feature value of below 60. When model is provided with the visit 1 data, no

hypoglycaemia will be detected, as the lowest point here is 80, hence the model will produce false negative results.

The possible solution to this problem would be to look at the dynamics of the system. For instance it is possible to look at the change in a feature value in relation to a reference point. Based on the experiment protocol prior to the time of insulin injection all patients are known not to be in hypoglycaemia. For this reason the point just before the insulin injection was chosen as a reference point. All other data points were calculated as a percent deviation from this baseline. Figure 5.5 shows a plot of the normalized feature values. Now the lowest point of both visits is approximately -50%. Meaning that a model trained on one visit will be able to detect hypoglycaemia in another visit.

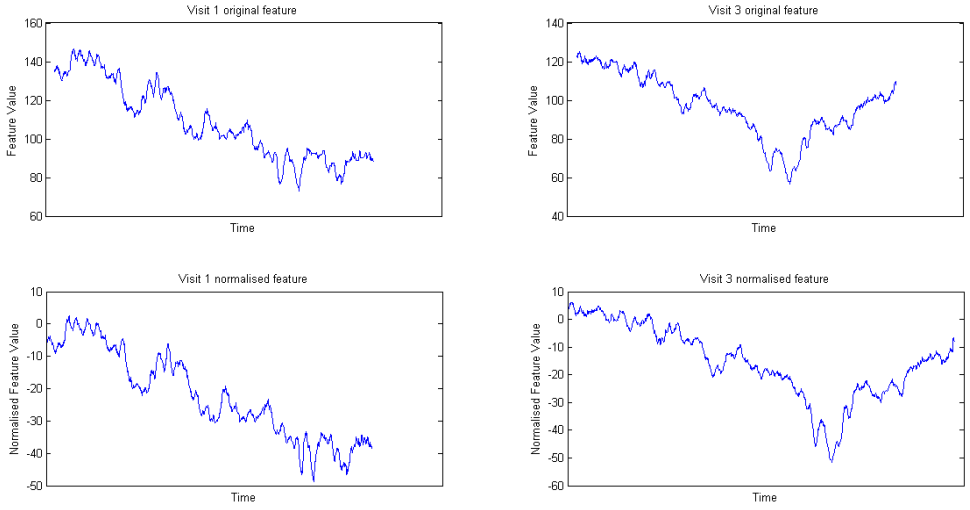

Figure 5.5: Absolute values and normalized values of a feature during visit 1 and 3

5.3 Classification Model

This section will explain how the final model was constructed. The section will start with explaining the overall structure of the model, followed by a description of how the model was validated to ensure its performance on new data. The section is concluded with a description of logistic regression which was used as a classification function.

5.3.1 Model Structure

The final model consists of 2 submodels trained on different data. This was done to make the model detect hypoglycaemia during experimental visits but also to be able to distinct a real

hypoglycaemia from the action of insulin. For more information see discussion section 8.2

Model 1 is the model that is trained on a single experiment visit. The data from this experiment is cut after the lowest point is reached. This is done as it was observed that after the BG level returns to normal after a hypoglycaemia the system continues to detect hypoglycaemia. In practice after a hypoglycaemia was detected the person with diabetes will hopefully take actions to avert it, so it is not essential for the system to detect the point when hypoglycaemia ends. For this reason it was chosen to focus the system on training to detect hypoglycaemia, this was done by cutting all data points after the lowest BS measurement was reached. Model 1 searches after a feature that changes during insulin induced hypoglycaemia and is most predictive of the event.

Model 2 is the model that can separate the real hypoglycaemia from solely action of insulin. Here the model is trained on the control visit data and the data used in model 1. This model will search after a feature that changes during hypoglycaemia but does not change during control visit. A feature that is not affected by the action of insulin is identified by model 2.

Combined model will analyze output from model 1 and 2 and if both detect hypoglycaemia the output is hypoglycaemia. This setup detects the hypoglycaemia based on model 1 and verifies the hypoglycaemia based on model 2.

5.3.2 Ensuring performance on new data

The aim of this study it to detect hypoglycaemia based on the ECG feature values. This is a typical classification problem, where performance of the model is measured with classification error rate: percentage of incorrectly classified instances. The model is created to classify new data. Hence it is important that the model is good on new data. The model is build based on training data, so it might overfit, i.e not generalize to unseen data.

One way to overcome this problem is to not use the entire data set when building a model. Some of the data is removed before training begins(train set and validation set). Then when training is done, the data that was removed is recombined with the training set into resubstitution set and can be used to test the performance of the model on "new" and "old" data. This is the idea of holdout validation. The classification function parameters are fitted using the training set only. Then the classification function is asked to predict the output values for the data in the resubstitution set. The errors it makes are used to evaluate the classification function.

Train and Validation sets have to be representative samples of the data that the model will be applied to. For this study the points were extracted based on the data distribution in the resubstitution sample, meaning that each of the sets contains the same percentage of hypoglycaemia and no hypoglycaemia points. The size of the two sets is important for

performance of the model. The more validation data, the better will the model perform on the new data. However if too little train data is accessible the model performance will suffer due to lack of information. For this study the train set is 9/10 of the data and the validation set is the remaining 1/10 as recommended in [43].

The disadvantage of holdout validation is that the evaluation may depend heavily on which data points end up in the training set and which end up in the validation set, and thus the evaluation may be significantly different depending on how the division is made. During this study it was observed that when using a 1/10 validation set the resulting model error was stable and hence the more complex k-fold cross validation was not needed.

5.3.3 Logistic Regression as a classification function

This section was written based on [44] and [45]. Given the fact that the event we want to predict is binomial (Hypoglycaemia/no Hypoglycaemia); ECG features are not normally distributed; variances of hypoglycaemia /no hypoglycaemia subsets cannot be assumed to be equal- logistic regression model was used.

Logistic regression is a statical tool that attempts to describe the relationship between a response variable(hypoglycaemia) and a single or a set of explanatory variables(ECG features). A detailed description of logistic regression can be found in appendix 10.4. Logistic regression uses a β parameter to link the possibility of hypoglycaemia with the value of a feature. If β is negative it means that an increase in the feature value makes hypoglycaemia less probable, conversely a positive β indicates that an increase in feature value makes hypoglycaemia event more probable. If β for a feature is 0 then a change in the value of a feature has no effect on the possibility of hypoglycaemia. Formally the null hypothesis is as follows:

- H0: $\beta_i = 0$, significance level a was set to 0.05.
- Ha: β_i is different from 0

5.4 Implementation

In present study Logistic Regression was used to construct model 1 and model 2 described in section 5.3.1. Two slightly different model creation algorithms were used to construct these models. Algorithm for creation of model 1 is shown on figure 5.6, and algorithm for model 2 is shown on figure 5.7.

Algorithm for model 1 feature selection:
The algorithm is provided with visit 1 data as described in 5.3.1. Multiple models are constructed by iterating through a range of BG values for definition of hypoglycaemia for a single feature. Each model is created based on model 1 train set and evaluated on model 1 resubstitution set. The model that produced the smallest error using a feature is stored together with

the BG threshold used to split the BG points. This process is repeated with all ECG features. Finally a feature that produced the smallest error is chosen for model 1. This feature and BG threshold used for splitting are stored.

Algorithm for model 2 feature selection:
This algorithm is provided with data for model 2 as described in 5.3.1. In this algorithm the BG measurements are split into Hypoglycaemia/No Hypoglycaemia groups based on the threshold identified in model 1. The algorithm then iterates through all features and creates a model based on model 2 train data for each feature. The models are then evaluated on the re-substitution set of model 2. Feature that produced that smallest error was chosen for model 2.

Final model creation and evaluation:
When the best features were identified model 1 and model 2 are created based on same data as used during feature selection for both models however now full data sets are used to train models (no validation set). The created models are then combined into final model and tested on data from a different session: experiment and control visits. The model was evaluated based on spline interpolated BG measurements and the ECG features that were filtered and normalized. Model sensitivity and specificity were calculated. In order to maximize the use of the data, the model was first trained on visits 1,2 and tested on 3,4. Then the model was trained on visits 3,4 and tested on visits 1 and 2. This was done for each patient. Results obtained can be found in results section 6.

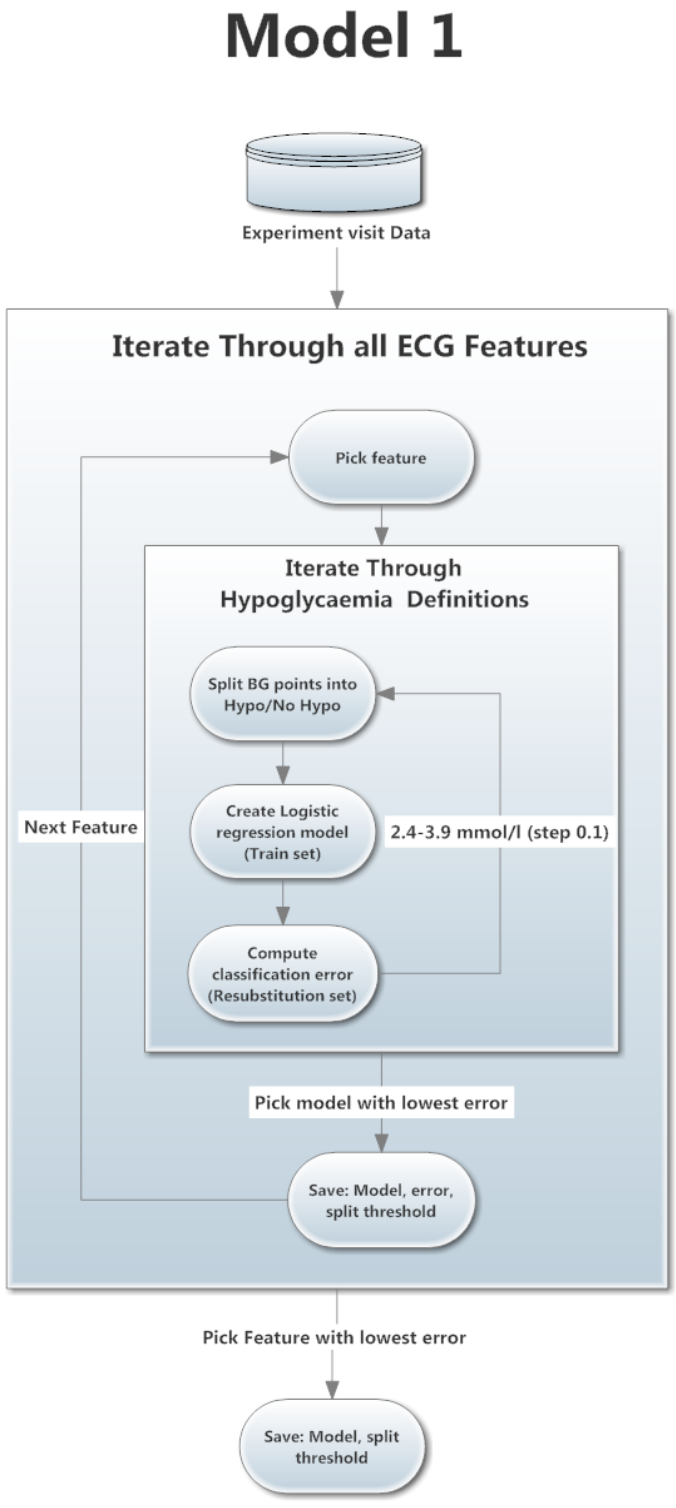

Figure 5.6: Data from visit 1 is input to this algorithm. The algorithm starts by selecting first feature then it iterates through the BG thresholds and creates a model for each threshold. Each model is tested on resubstitution set. The best model for a feature is stored. The algorithm repeats with another feature until all features were evaluated. Finally the best feature is picked based on the minimal error on the resubstitution set. BG threshold used for this feature is also stored.

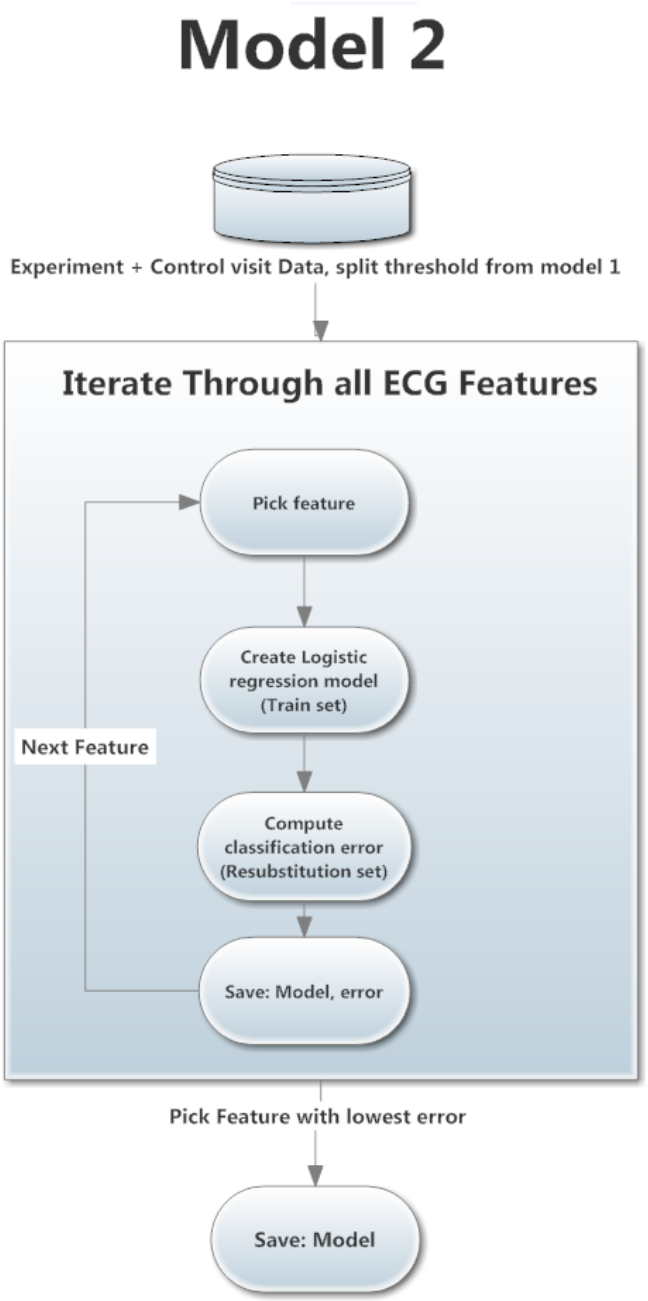

Figure 5.7: Data from both visits and the BG threshold identified in model 1 are input to this algorithm. The algorithm iterates through all features and computes a model for each of them. The BG threshold from model 1 is used to split BG measurements into Hypoglycaemia/No Hypoglycaemia points. Each model is then tested on a resubstitution set. Finally the feature that produced a model with the lowest error is selected for model 2.

5.5 Error Sources

ECG measurements:
The correct electrode placement is essential to get the correct and accurate ECG signal. This is also true for a 12 lead ECG where the heart is viewed from different angles. Any slightly misplaced electrodes might show an incorrect view of the heart or pick up more noise from the surrounding muscle tissue. For this study "SpiderView Plus Digital Holter monitor" was used to measure ECG. This monitor samples the ECG signal with a frequency of 1000 Hz and performs no data processing. The ECG was filtered and the features were extracted using an external algorithm. The resulting features had a frequency of 6 Hz. The algorithm used for feature extraction was not accessible and the error could not be evaluated.

In the present study the features were filtered as a part of data pre-processing described in section 5.2.2. Filter used a window size of 5. The time between ECG measurements was 10 seconds. This introduces a delay of 50 seconds (each point is calculated by extracting average of a point at time i and the 4 points before it: i-1, i-2, i-3, i-4).

BG measurements:
The BG measurements were collected by sampling the capillary blood from the ear, using "HemoCue Glucose 201+" glucose monitor. The trueness of this glucose monitor was assessed in a study from 2005 [46]. This study reported that the difference between the actual capillary BG and the monitor measurements was $+-0.4\%$, additionally this difference was found insignificant by the test statistics.
In resent study the BG measurements were interpolated. Every interpolation introduces an error term, however given the hight frequency of BG samples in the critical hypoglycaemia range this error is thought to be small. A more detailed description on error estimates associated with interpolation can be found in section 5.2.1.

6 Results

6.1 Results of initial analysis

The overall results of initial analysis are summed in the following text.

T-wave Analysis of T-wave related features by means of Mood's test provided some statistically significant results, which can be viewed as trends. It was shown that T- wave peak amplitude was affected, the most profound difference in T-wave peak amplitude was achieved when BG was at 2.49 mmol/l. In addition it was shown that the T-wave morphology score (morphology score measure describes the differences in T-wave morphology between normal values and the observed values, the higher the score the less normal the observed T-wave is) showed the most profound difference when split into hypoglycaemia and no hypoglycaemia sets at a BG threshold value of 3.345 mmol/l.

R wave R wave analysis showed that changes in amplitude occurred at 2.49 mmol/l and duration at 2.345 mmol/l.

TP interval Another find was a change in the TP-interval interval occurring at a BG of 2.8164 mmol/l.

S wave Changes in the S wave peak amplitude and duration were found to be most clear at a BG level of 2.44 mmol/l.

RP interval Change of RP interval peak amplitude was found to be most profound at level of 2.325 mmol/l BG.

P wave Changes in the p wave peak amplitude were most profound at BG level of 3.345 mmol/l.

ST segment A significant change in the ST segment was identified. It was found that at BG of 2.5 mmol/l these changes were most profound.

6.2 Results of model development

This section will describe the results obtained from model testing. The model was evaluated for each patient individually, the specificity, sensitivity, features used for model, hypoglycaemia threshold used to define hypoglycaemia and the delay from the time when actual BG falls below Hypoglycaemia threshold until the system detects hypoglycaemia are listen in table 6.1. As described in section 5.4 the model was evaluated by training on V3, V4 and then tested on V1 and V2. Then the model was trained on V3,V4 and tested on V1, V2. The results obtained for each patient are summed in following text.

Patient 4:

Figure 6.2 shows a plot of data used for testing and the detected hypoglycaemia events for patient 4. It can be seen that when the model was trained on V3 and V4 it failed to identify any hypoglycaemia measurements during visit 1 (sensitivity = 0). When the model was trained on V1 and V3 the model was able to identify hypoglycaemia 180 seconds after the BG fell below hypoglycaemia threshold of below 2.4 mmol/l. During the control visits the model did not produce any false positives (specificity = 1).

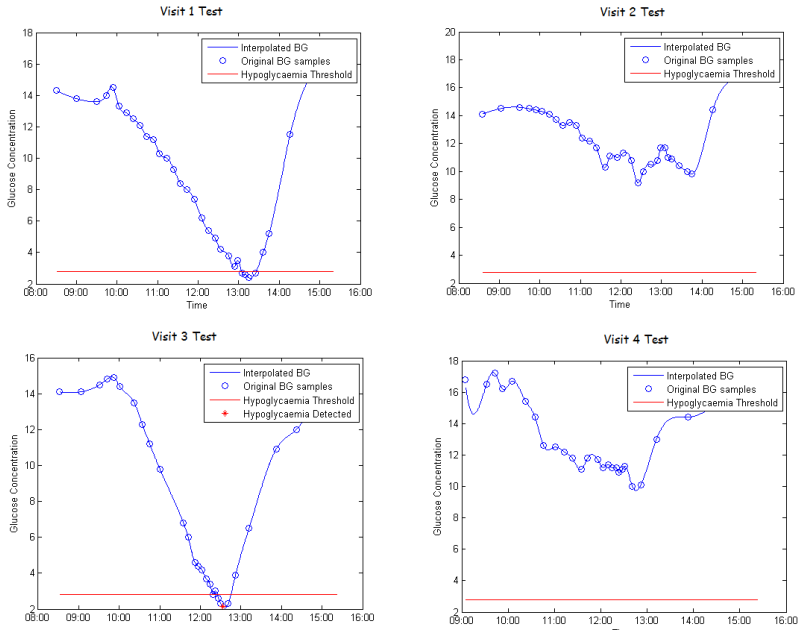

Figure 6.1: Plots of data used for model testing and detected hypoglycaemias of patient 4. Visit 1,2,3 and Test figures show the BG measurements during these visits. The red line indicates the hypoglycaemia threshold and red dot indicate the first point that is classified as hypoglycaemia.

Patient 5:

Testing results of patient 5 model are shown on figure 6.2. When the model was tested on visit 1 hypoglycaemia was detected 500 seconds after the actual BG fell below hypoglycaemia threshold of 3.6 mmol/l. Tests on Visit 3 showed that the hypoglycaemia was detected 19

minutes after the actual BG fell below 3.5 mmol/l. Tests on both control visits produced false positive results.

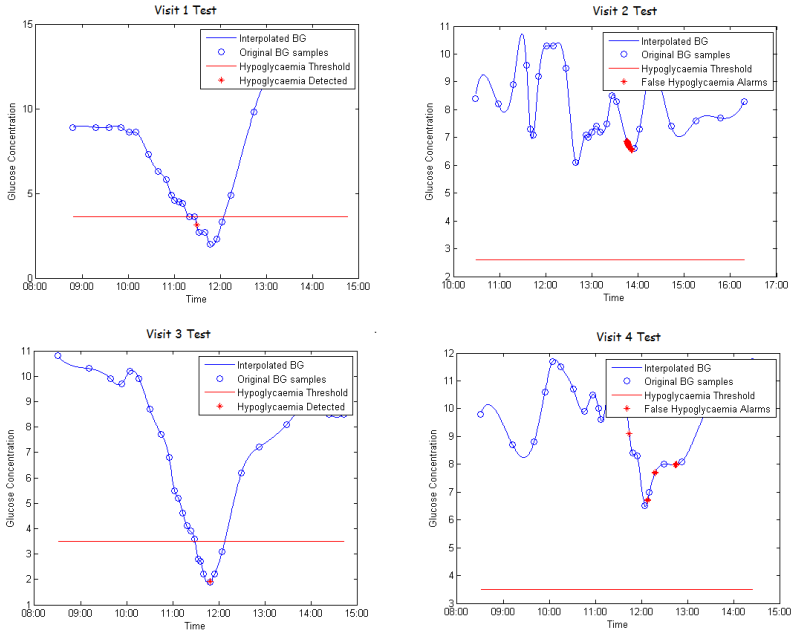

Figure 6.2: Plots of data used for model testing and detected hypoglycaemias of patient 5. Visit 1,2,3 and Test figures show the BG measurements during these visits. The red line indicates the hypoglycaemia threshold and red dot indicate the first point that is classified as hypoglycaemia on experiment visit. On control visits all points that were classified as hypoglycaemia are marked red.

Patient 6:

Model tests for patient 6 are shown on figure 6.3. The hypoglycaemia was detected during both experiment sessions. During visit 1 hypoglycaemia was detected 420 seconds after BG fell below 3.0 mmol/l (hypoglycaemia threshold), while during visit 3 hypoglycaemia was detected 80 second after the hypoglycaemia threshold of 2.5 mmol/l was crossed by the actual BG. During both control visits the model produced no false positives (specificity = 1).

Patient 8:

Model tests for patient 8 are shown on figure 6.4. Tests on visit 3 showed that model could detect hypoglycaemia already 10 seconds after the actual BG fell below hypoglycaemia threshold of 2.5 mmol/l. Tests on V4 Control visit produced no false positives (specificity = 1). When tested on V1 the model could detect hypoglycaemia 440 seconds after the BG fell below the hypoglycaemia threshold of 2.7 mmol/l. However during tests on V2 the model produced false positives.

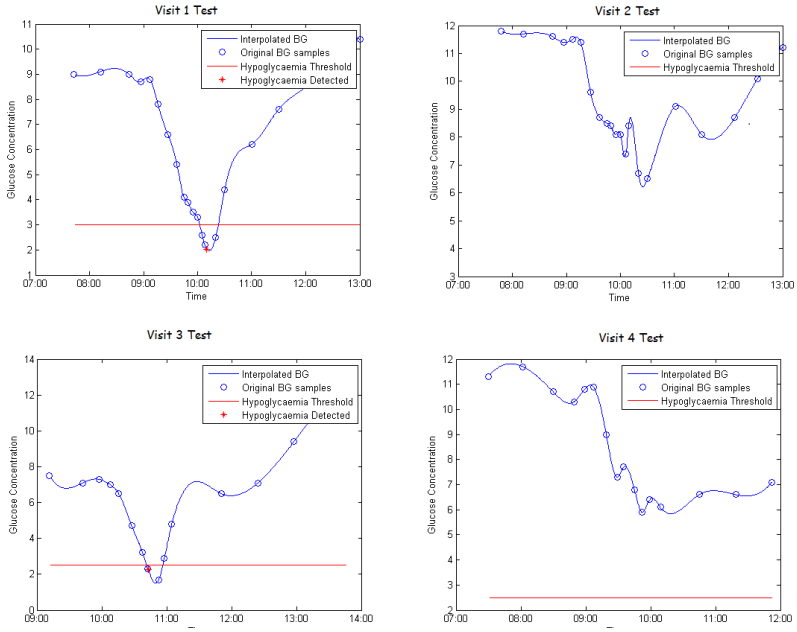

Figure 6.3: Plots of data used for model testing and detected hypoglycaemias of patient 6. Visit 1,2,3 and Test figures show the BG measurements during these visits. The red line indicates the hypoglycaemia threshold and red dot indicate the first point that is classified as hypoglycaemia.

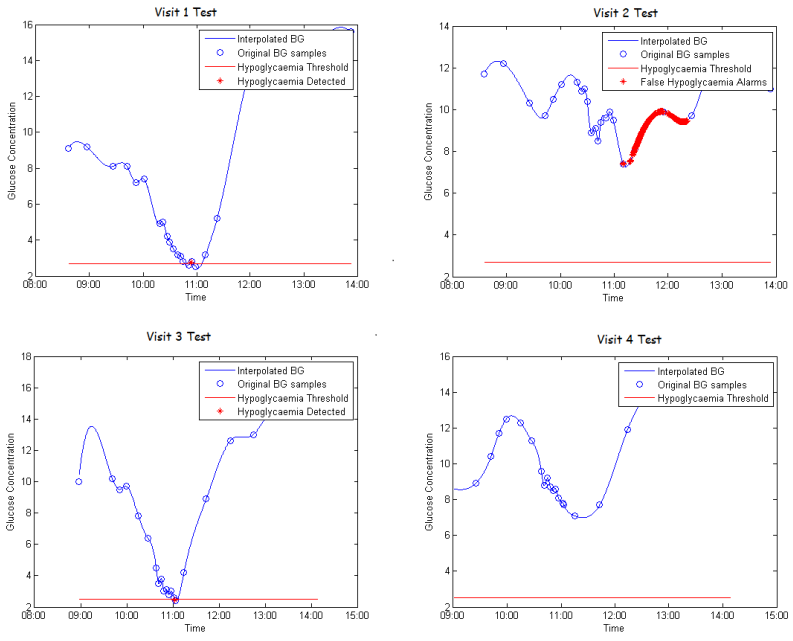

Figure 6.4: Plots of data used for model testing and detected hypoglycaemias of patient 8. Visit 1,2,3 and Test figures show the BG measurements during these visits. The red line indicates the hypoglycaemia threshold and red dot indicate the first point that is classified as hypoglycaemia during experiment visit. During control visits all points that were classified as hypoglycaemia are marked red.

	Test On V1			Test On V2		Model Information	
Patient	Sensitivity	Specificity	Delay	Sensitivity	Specificity	Features	Hypo Threshold
4	0%	100%	no detected	NaN	100%	$10; 2$	$2.8 mmol/l$
5	89%	100%	$500 sec$	NaN	98%	$263; 88$	$3.6 mmol/l$
6	13%	100%	$420 sec$	NaN	100%	$7; 263$	$3.0 mmol/l$
8	32%	100%	$440 sec$	NaN	84%	$56; 163$	$2.7 mmol/l$

	Test On V3			Test On V4		Model Information	
Patient	Sensitivity	Specificity	Delay	Sensitivity	Specificity	Features	Hypo Threshold
4	23%	100%	$180 sec$	NaN	100%	$63; 263$	$2.4 mmol/l$
5	9%	100%	$19 min$	NaN	99%	$263; 238$	$3.5 mmol/l$
6	52%	100%	$80 sec$	NaN	100%	$213; 38$	$2.5 mmol/l$
8	67%	100%	$10 sec$	NaN	100%	$55; 99$	$2.5 mmol/l$

Table 6.1: This table shows the performance of the model when trained on visit V3, V4 / V1, V2 and tested on V1, V2/ V3, V4. And Features column indicates features that were chosen for each patient for model1 and model2. Hypo Threshold shown the threshold value that is used to define hypoglycaemia for a specific patient. The Delay column indicates the time that passed from the point when actual BG falls below Hypo threshold until the system detects hypoglycaemia. Sensitivity and Specificity of the model on V1, V2, V3, V4 is also listed.

7 Application Development

7.1 System vision

The system should sound an alarm, alerting the person with diabetes of a hypoglycaemia event. Hypoglycaemia should be detected based on the ECG signal. This alarm should give the person with diabetes adequate time to react by simply taking in some rapid acting glucose such as a juice, hence averting severe hypoglycaemia.

For such a system to work it will need a mathematical hypoglycaemia detection model that was described in chapter 5. The system should allow the medical worker responsible for diabetes management to compute such model.

7.2 System description

The system is supposed to give people with diabetes a chance to avert a severe hypoglycaemia. The system is divided into two separate parts; Mobile application- used by the person with diabetes; PC application- used by the medical worker.

Mobile application

In order to identify the requirements for the Mobile application interviews with two individuals with diabetes were conducted, resulting in a set of usecases. The interview questions can be found in appendix 10.1. During these interviews people with diabetes were asked what the hypoglycaemia detection system should do. The main requirement was that a hypoglycaemia alarm should be sounded when the blood sugar falls to dangerously low levels. In addition interviewees stated that the system should store history of all hypoglycaemia events that can be viewed by the person with diabetes to better understand the pattern of these events and possibly take actions to avert them. This was needed due to the fact that both

interviewees did not systematically make a log of their hypoglycaemia events, due to inconvenience. However both interviewees expressed a wish to have such information available. The third function that the system should be able to perform is to provide helpful hints during a hypoglycaemia event. Interviewees stated that under normal conditions they know exactly how to act during hypoglycaemia, however "during hypoglycaemia everything can get very confusing"- stated one of the interviewees. Therefore an action guide should be provided, implicitly stating that the blood sugar is, indeed, critically low and glucose should be taken in immediately. In addition it was stated that the surrounding people can be unable to help if a severe hypoglycaemia occurs. The interviewee expressed the need for the system to be able to aid the people that are trying to help the person with diabetes. Finally for the system to function it should download the prediction model created in PC application. The specific model is chosen based on Patient ID.

PC application

The second part of the system that is operated by medical worker should give the ability to load hypoglycaemia data into the system, compute the model and store the model. Secondly the system should be able to make the data accessible for download for the person with diabetes using Mobile application. The uploaded model is tagged with a Patient ID.

7.3 Requirements

In this section the objective is to describe what the system should do and the means to obtain this information. Functional requirements describe what the system should do. The functional requirements were identified using use cases and actors that in turn were identified based on interviews. First step in this process is to identify the actors and secondly the use cases. 7.1 illustrate identified actors and use cases.

7.3.1 Identification of actors

Considering the system description three actors can be identified:

- **Person with diabetes:** using mobile application.
- **Medical worker:** Creating and uploading a hypoglycaemia detection model through PC application.
- **Helper:** A person trying to help the person with diabetes through mobile application.

The actors are drawn as "‘stick-men"’ in 7.1. The "person with diabetes" interacts with the mobile application by accessing the systems functions and setting up the user while the "medical worker" interacts with PC application by computing and uploading the mathematical model necessary for the prediction algorithm to work on the mobile application. The third user "helper" interacts with a small part of the mobile application by viewing the guides. The two parts of the program are different systems as indicated by the system boundary.

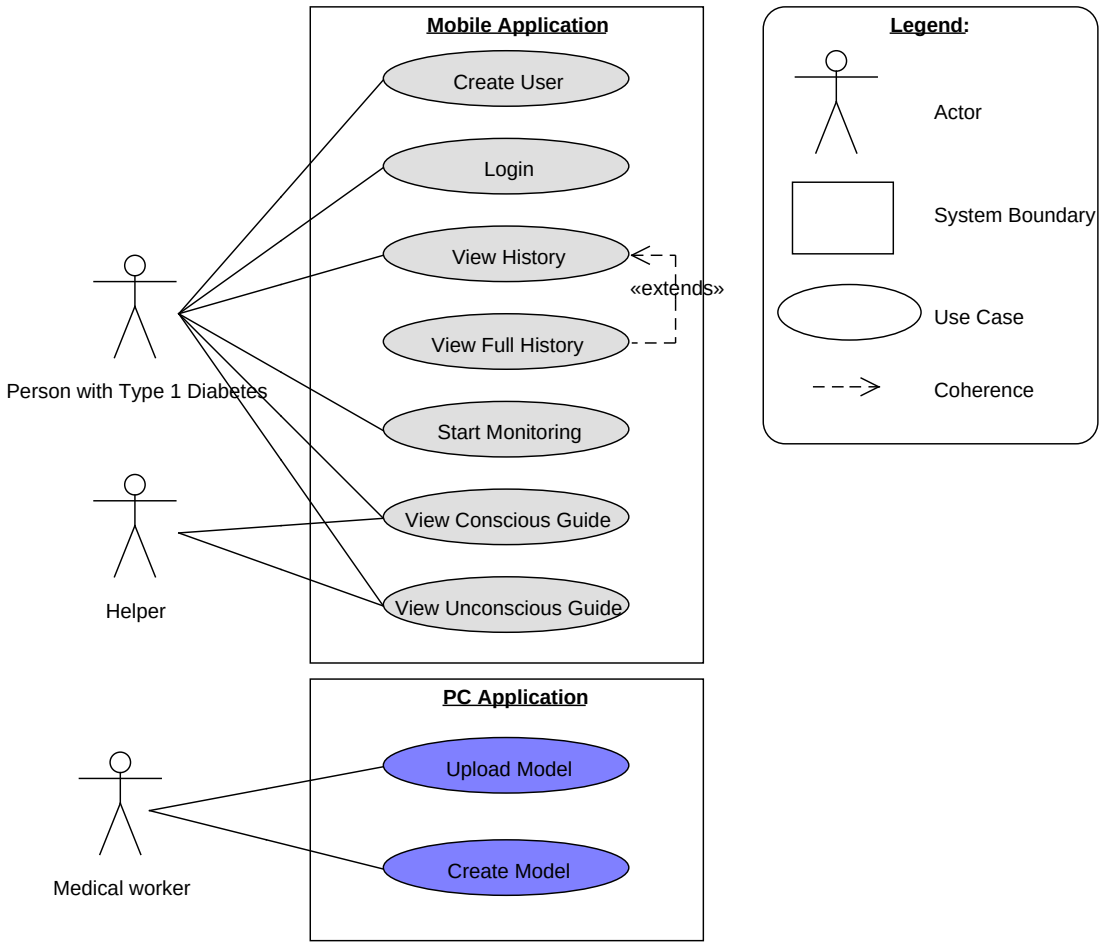

Figure 7.1: Use case diagram for the system

7.3.2 Identification of use cases

Use cases are descriptions of functionality of the system. To identify use cases one can ask: how, what and finally check if it is a use case [47]:

- **How** does each actor use the system?
- **What** does the system do for each actor?
- **Check** Does the function produce something of measurable value to the actors?

In the following text the usecases will be identified.The system is divided in two parts the mobile application and PC application.

Mobileapp

The first part of the system is the mobile system, it contains following use cases:

Create user use case. In order for the "person with diabetes" to start using the mobile application a new user must be crated. During this use case a link to the ECG monitoring device is established; access to a server holding the patients personal model is established and the model created in PC application is downloaded.

Login usecase allows "person with diabetes" to access mobile applications main functions.

View history use case gives the "person with diabetes"' the ability to view a bar chart of hypoglycaemia events.

View Full history use case gives the "person with diabetes" the ability to view a list of all hypoglycaemia events with times and dates attached.

Start monitoring use case starts analysing the incoming ECG data and in case of a hypoglycaemia an alarm message is displayed. This use case is activated by "person with diabetes"'

The **view conscious guide** use case is activated by "person with diabetes" or "helper" in case of a hypoglycaemia if guidance is needed in treating mild or moderate hypoglycaemia.

The **view unconscious guide** use case is activated by "helper"' in case of a hypoglycaemia if guidance is needed in treating severe hypoglycaemia.

PC application

The other part of the system is the PC application used by the "medical worker" and contains 2 usecases.

Upload model usecase is activated by the "medical worker" to upload the ECG prediction model so that it can be downloaded by mobile application (Create user use case).

Create model use case is activated by "medical worker", the system is provided with patient data and the model for prediction of hypoglycaemia is the output of this usecase.

7.3.3 Use case descriptions

Figure 7.1 illustrates the 9 identified use cases. This section will elaborate the description of each use case. It is common to include activity diagrams in the descriptions of the use cases. This report does not include activity diagrams because "Flow of events" is already a simple sequential description of the use case. The naming convention for use cases is: Use Case#: name, e.g "UC1: "Create Model".

Use case: Create Model
ID: UC1
Description: This use case allows the medical worker using PC application to specify a path to the patient's hypoglycaemia data and receive a model as output.
Actor: medical worker
Preconditions: The actor has started the system; the data from the experiment is stored on the computer as .ecg files.
Main flow: 1. The actor Enters a Patient ID. 2. The actor selects a path to the experimental patient specific data stored on the computer. 3. System checks the data. 4. Data valid: a. System computes a model based on the provided data. b. System attaches the Patient ID to the Model and patient name and stores the file.
c. When the model is created the actor is informed.
Post-conditions: The model was created and a patient ID is attached to the model.
Alternative flow: 4. Data invalid a. The user is asked to specify a different .ecg file

Table 7.1: Use case-specification: Create Model

Use case: Upload Model
ID: UC2
Actor: medical worker
Description: This use case lets the medical worker upload data model so that it can be downloaded by the mobile application. PC application stores the patient's model on a web server and tags it with patient ID.
Preconditions: The actor started the system. At least one patient ID tagged model was created (UC1).
Main flow: 1. The actor is presented with a list of patients whose models are computed. 2. The marked patient is presented in detail 3. When the actor the presses upload button the system accesses the web server. 4. If webserver is accessed, the system uploads the patients id and computed model.
Post-conditions: The model was created and a patient ID is attached to the model.
Alternative flow: 4. If the system could not establish link to the server an error message is displayed.

Table 7.2: Use case-specification: Upload Model

Use case: Login
ID: UC3
Actor: Person with diabetes
Description: This use case allows person with diabetes to access the Mobile applications main functionality by entering a password and a username. If the security check passes then the systems main screen is displayed.
Preconditions: The actor started the system. At least one user was created (UC4).
Main flow: 1. Actor is presented with the password and login field. 2. When the actor enters password and username the system checks these for validity. 3. If both the username and password are correct the system proceeds to load the user data and displays the main screen.
Post-conditions: The user related data are loaded and main screen is displayed.
Alternative flow: 3. If the username or password are incorrect an error message is displayed.

Table 7.3: Use case-specification: Login

Use case: Create User
ID: UC4
Actor: Person with diabetes
Description: This use case allows the "person with diabetes" to create a new Mobile application user. During this usecase the "person with diabetes" enters the required data, including the patient specific ID that is used to access the model on the web server. The result of this usecase is a new user file that includes users ECG model. Password and username are associated with this file and can be used in login usecase.
Preconditions: The actor started the system.
Main flow: 1. The system displays input field where patients Patient ID, Username and Password have to be entered. 2. When all the fields are filled and create button is pressed by the user the system attempt to access the web server and compare the specified patient id with existing files. 3. If the ID could be matched and connection was established the model begins downloading. Upon completed download the model is integrated into the program and the user is informed of successful user creation.
Post-conditions: The model is stored on the mobile device and integrated into the system and a new user is created.
Alternative flow: 3.a If the connection fails an error is displayed. 3.b If the patient ID could not be matched to any existing on the online database an error is displayed.

Table 7.4: Use case-specification: Create user

Use case: View History
ID: UC5
Actor: Person with diabetes
Description: This use case allows the "person with diabetes" to access the history of the previous hypoglycaemia events. A bar chart is presented displaying history of the hypoglycaemia events.
Preconditions: The user is logged in (UC3), main screen is displayed.
Main flow: 1. The actor chooses to view history by pressing the "View history" button. 2. A containing the date of last hypoglycaemia, total number of hypoglycaemias and a bar chart showing the hypoglycaemia events by month displayed.
Post-conditions: The hypo history information and bar chart is displayed on a history screen.
Alternative flow: None

Table 7.5: Use case-specification: View History

Use case: View FullHistory
ID: UC6
Actor: Person with diabetes
Description: This use case allows the "person with diabetes" to access the detailed log of previous hypoglycaemia events. All the previous hypoglycaemia events are displayed as a list with a day of the week, date and time of each event. This use case extends View history use case(UC5).
Preconditions: The history screen is displayed (UC5).
Main flow: 1. The actor chooses to view full history of hypoglycaemia events by pressing the full history button. 2. The system displays a full list of hypoglycaemia events with date and time attached to each event.
Post-conditions: A list with all hypoglycaemia events is shown.
Alternative flow: None

Table 7.6: Use case-specification: View FullHistory

Use case: Start monitoring
ID: UC7
Actor: Person with diabetes
Description: This use case allows the person with diabetes to start monitoring the live ECG signal. If a hypoglycaemia is detected an alarm is activated.
Preconditions: The actor is logged in (UC3), main screen was displayed.
Main flow: 1. The actor chooses to start monitoring by pressing the start monitoring button. 2. The system analyses the incoming ECG signal and computes the risk of hypoglycaemia 3. If hypoglycaemia is detected an alarm is sounded and an alarm screen is displayed.
Post-conditions: An alarm is sounded and an alarm screen is displayed.
Alternative flow: 3. If no hypoglycaemia is detected nothing happens

Table 7.7: Use case-specification: Start Monitoring

Use case: View Conscious Guide
ID: UC8
Actor: Person with diabetes, helper
Description: This use case allows the "person with diabetes" or "helper" to access a guide that will help to treat a mild or moderate hypoglycaemia.
Preconditions: Hypoglycaemia was detected, alarm screen displayed after start monitoring use case (UC7).
Main flow: 1. The actor chooses to view the guide by pressing the "person conscious guide" button. 2. The system turns off the sound alarm and displays the first step of the guide. 3. The actor presses next after completing the step 4. System shows the next step of the guide. 5. Steps 3-4 repeat until the end of the guide is reached. 6. The system exists to the main screen.
Post-conditions: The system returns to the main screen.
Alternative flow: 3. The actor presses reset system button and exits to the main screen

Table 7.8: Use case-specification: View Conscious Guide

Use case: View Unconscious Guide
ID: UC9
Actor: helper
Description: This use case allows the "helper" to access a guide that will help to treat a severe hypoglycaemia.
Preconditions: Hypoglycaemia was detected, alarm screen displayed after start monitoring use case (UC7).
Main flow: 1. The actor chooses to view the guide by pressing the "person unconscious" button. 2. The system turns off the sound alarm and displays the first step of the guide. 3. The actor presses next after completing the step. 4. System shows the next step of the guide. 5. Steps 3-4 repeat until the end of the guide is reached. 6. The system exists to the main screen.
Post-conditions: The system returns to the main screen.
Alternative flow: 3. The actor presses reset system button and exits to the main screen

Table 7.9: Use case-specification: View Unconscious Guide

7.3.4 Specifying requirements

The functional requirements for the system were identified by analyzing the use case descriptions. The following requirements were identified:

Requirements for Mobile application:

R0: The system **shall** allow the user to protect personal information.

R1: The system **shall** analyse live ECG signal for risk of hypoglycaemia

R2: The system **shall** sound an alarm if hypoglycaemia is detected.

R3: The system **shall** provide an action guide for treating hypoglycaemia.

R4: The system **shall** show history of hypoglycaemia events as a chart.

R5: The system **shall** show full log of all hypoglycaemia events with dates and times attached.

R6: The system **shall** allow user to create a new user and download the patient specific model from a webserver

Requirements for PC application:

R7: The system **shall** allow user to analyse the experiment data and extract a model based on these.

R8: The system **shall** allow user to view all patient models currently accessible.

R9: The system **shall** allow user to upload a patient specific model to a webserver.

7.4 Analysis and Design

An analysis model describes what the system should do but not how it is done. An analysis model consists of analysis classes for the system and use case realization. Analysis classes model key concepts of the problem domain. Use case realizations describe how instances of the analysis classes are related.

The purpose in design is to specify how the functionality will be implemented.

For the purpose of this project Analysis and design will not be described since the focus of this project is not to develop a fully functioning piece of software but merely a prototype. Instead this section will include a deployment diagram showing how the Mobile Application and the PC application should be deployed, which is usually a part of analysis section.

7.4.1 System deployment

The deployment diagram of the system is sown on figure 7.2. The system is based on three separate units. The system will allow users of Mobile and PC applications to access the webdatabase. This can be implemented using PHP and MySQL. PHP as the scripting language which is called (via http) from the applications, and handles all the database access. MySQL is the database. PHP is a wide used language supported by android and windows. This setup works as follows: The database is installed on the webserver holding information on all patients that were stored from all the different PC applications. In addition a webarea is created on webserver which has access to the database. The PHP scripts are placed within the webarea. The web scripts access the database, with functions for fetching and storing information.

The mobile and PC applications use a httpclient to send http GET or http POST requests to the PHP scripts on the webserver, either sending model information or asking for model information. Each model is tagged with a Patient ID which is specified in the PC application during http POST. The Mobile application has to specify a Patient ID during http GET to access the model information.

It's also possible to write database access Java code directly in the application. However accessing database server directly is a security risk.

Mobile and PC applications store data locally by using serialisable objects. This format for data storage was chosen due to the fact that the data that has to be stored locally is very limited and there is no need for a database.

For the system to be fully functional some measure of security other than the ID tagging has to be implemented. This might be a concern if the data on the webserver will hold sensitive patient related information. If only using the patient ID the database might be accessed by brute forcing to pick a valid patient ID. The possible solution is to implement a user manager on the webserver. In such a configuration the http requests to the PHP from the applications will include patient ID, Username and password. The user manager will only allow http requests to go through for registered users. This will make the system accessible only for registered users. The Patient ID can still be bruteforced, however the risk of such an event is diminished as access is granted to fewer people.

To make the system even more secure following steps need to be taken: The "medical worker" will upon upload create patient ID, model parameters and a One Time Key (OTK). This key can be communicated to the patient f.ex by means of an SMS. The patient in turn upon creation of user enters the password username, name and the OTK. This implementation will insure that the data on the webserver will not be accessible if the phone security becomes compromised. This is achieved because the patient ID is not used in the mobile application. Instead a OTK links to the Patient ID which in turn links to the patient model data. The OTK becomes inactive after the model was downloaded.

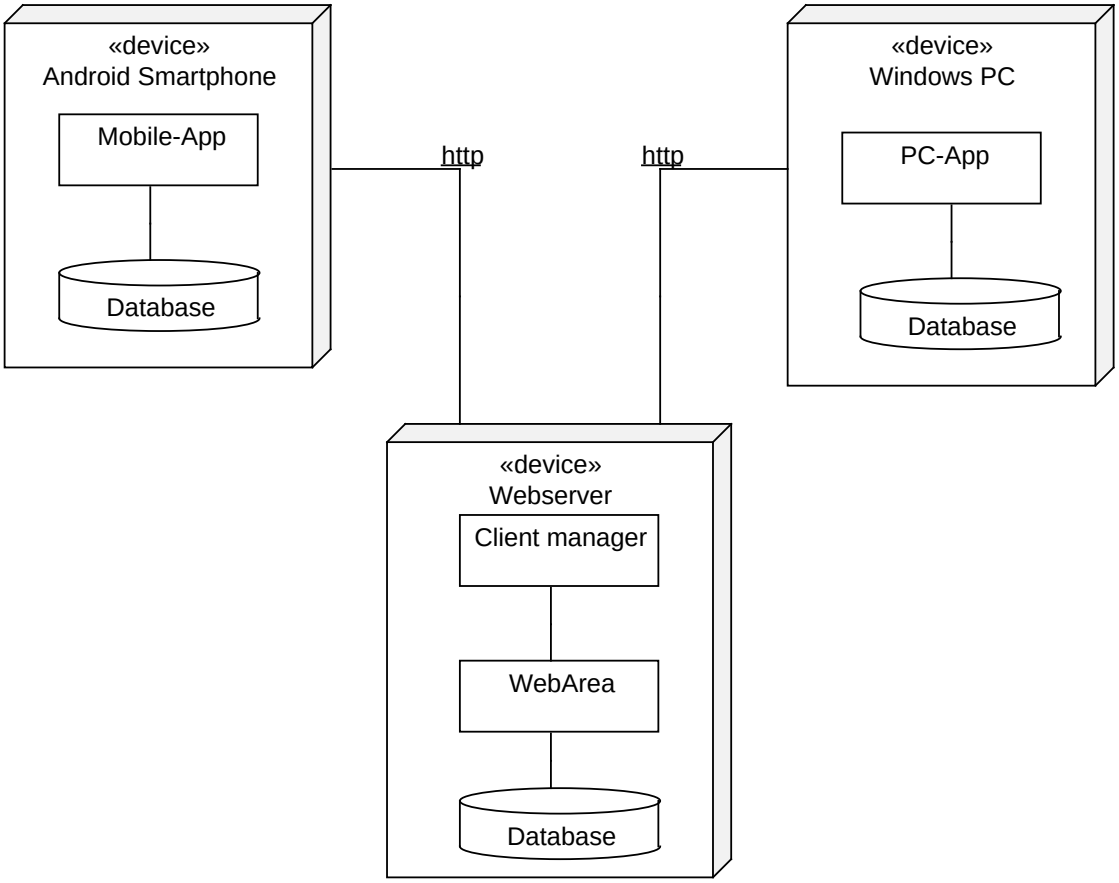

Figure 7.2: Deployment diagram

7.4.2 Limiting the scope

The time related limitations of this study make it necessary to reduce the scope of the system. In order for the system to be fully functional the ECG signal has to be sampled and feature extraction has to be implemented live. This was not implemented due to the time limits of the present study. In addition a web server would have to be established for the mobile application to be able to get the model and for the PC application to be able to upload model. The web server as well as the PC application was not implemented. The resources available were focused on designing a prototype of the Mobile application. The model was stored locally in a file as if it was downloaded from the webserver. The input ECG signal was simulated for the mobile application. For this project the Mobile Application was be programmed in Java. Taking study limitations into consideration the original system requirements were modified. Requirement 7-9 were removed and requirement 6 was modified to use internal database to get the model from. The modified requirement list is listed below.

Requirements for Mobile application:

R0: The system **shall** allow the user to protect personal information.

R1: The system **shall** analyse live ECG signal for risk of hypoglycaemia

R2: The system **shall** sound an alarm if hypoglycaemia is detected.

R3: The system **shall** provide an action guide for treating hypoglycaemia.

R4: The system **shall** show history of hypoglycaemia events as a chart.

R5: The system **shall** show full log of all hypoglycaemia events with dates and times attached.

R6: The system **shall** allow user to create a new user and load the patient specific model.

7.5 Implementation

The Mobile application prototype was implemented in Java, using the libraries supported by most android devices. The application consists of a set of serialized objects used to store user data and the Java code itself. The prediction algorithm was implemented as a java class which needs a set of patient specific parameters to function. The parameters are for the purpose of this study hardcoded as input during model class creation.

I/O streams

The system uses Javas IO stream to store and retrieve information. This is used to store and retrieve the hypoglycaemia events as well as patient data on user creation. In this study file I/O object streams were used (ObjectInputStream, ObjectOutputStream) for storing and retrieving users personal information and hypoglycaemia event history. ObjectInputStream and ObjectOutputStream classes have the readobject() and writeobject() methods that can be used o either store or retriever an object. Below is a code example of storing and retrieving patients name.

```
\\When saving
              \\Open a file to write to, named SavedObjects.sav
              FileOutputStream saveFile = new FileOutputStream("SavedObjects.sav");

              \\Create an ObjectOutputStream to put the objects into the save file.
              ObjectOutputStream save = new ObjectOutputStream(saveFile);

              \\Now save
              save.writeObject(name);

\\When loading
              \\Open file to read from, named SavedObjects.sav.
              FileInputStream saveFile = new FileInputStream("SavedObjects.sav");
```

```
\\Create an ObjectInputStream to get the objects from the save file
ObjectInputStream load = new ObjectInputStream(saveFile);

\\Now load.
name = (String) load.readObject();
```

Threads

In the developed prototype threading was used. Multithreading refers to two or more tasks executing at the same time within a single program. A thread is an independent path of execution within a program. Threads are often used when visualisation has to be accompanied by other tasks. This is useful in the case when the hypo history is loading during view history usecase. Here two different tasks have to be executed, namely data associated with a user has to be loaded and a visualisation has to be created. The thread first needs to be instantiated and then a thread.start() method is called to start the thread. Here 2 processes run simultaneously. Later thread.join() is called to halt the main process until the thread has finished computing.

Singleton design pattern

The singleton design pattern is another design principle implemented in the system. The idea behind singleton is to ensure that a class has only one instance, and provide a global point of access to it. The way to ensure only ONE instance of the same class exists, is to declare the constructor private so that the class can only be instantiated by itself. The instance of the class is declared to be a class variable by using the java-keyword: "static". When the class has to be instantiated, the method checks if there already exists an instance of this class, and if not, the method will create a new instance, otherwise an existing instance is returned. This was implemented in the system when simulating and analyzing the ECG signal. Singleton design pattern insured that only one simulation could exist.

```
public class SimulateECG {
       private static SimulateECG instance = null;

       private SimulateECG() {
        \\create a simulation
           }
       public static SimulateECG getInstance() {
              if (instance == null) {
                 instance = new SimulateECG();
                 }
              return instance;
       }
}
```

7.6 Graphical User Interface development

The graphical user interface or GUI was designed and developed in three iterations. The first iteration took place prior to the interviews with people with diabetes. The initial mockup developed in first iteration was used during interviews as an example of what the system could look like. During second iteration the information gained from the interviews and the designed use cases was used to expand the system with additional screens to house new functions and make changes to the existing ones. The third and last iteration of the GUI design took place after a user test. The test results was used to apply final changes to the GUI.

First iteration
A mock-up of interface was developed, for the Mobile application based on the developers understanding of the problem domain. The system mock-up can be seen on figure 7.3. This mock-up was developed prior to the interviews with the diabetes patients. The mock-up contains 4 different screens (labeled A, B, C, D on figure 7.3).

Screen A is a login screen displaying fields where username and password have to be entered. The two buttons on this screen are :login and Setup coloured green and blue.

Screen B is a view of the main screen. Here the date of last hypoglycaemia, system status and ECG signal status are shown. The system and ECG signal status as well as the date of last hypo are colour coded. Below the last hypo date is a blue "Hypo History" button, which opens the Hypo history screen (screen C on figure 7.3) . The green "start monitoring" button is the largest button on the screen as it will be used the most. In the upper left corner is a red "logout" button.

Screen C is the hypoglycaemia history screen which shows the date of last hypoglycaemia event, the total number of hypoglycaemia events as well as a bar chart showing hypoglycaemia events by month. In the right upper corner is a red "Back to main" button.

Screen D is the hypoglycaemia alarm screen, containing a warning message stating that hypoglycaemia is present and that actions have to be taken to stop the hypoglycaemia. The instructions are coloured red. Largest button on the screen is the green "Reset System" button.

The main design idea behind this initial mock-up was to make the most used buttons the largest and to be consistent with the placement of exit and back buttons. In addition the buttons were colour coded, where red was used for back or logout functions, blue was used for utility buttons and green was used for performing major tasks such as login, start monitoring and reset system. The information provided to the user was also colour coded, Any negative events or not functioning system parts are marked red, while functioning system parts are marked green. This primary design was subject to large changes after the interviews, f.ex. colour choices were reconsidered.

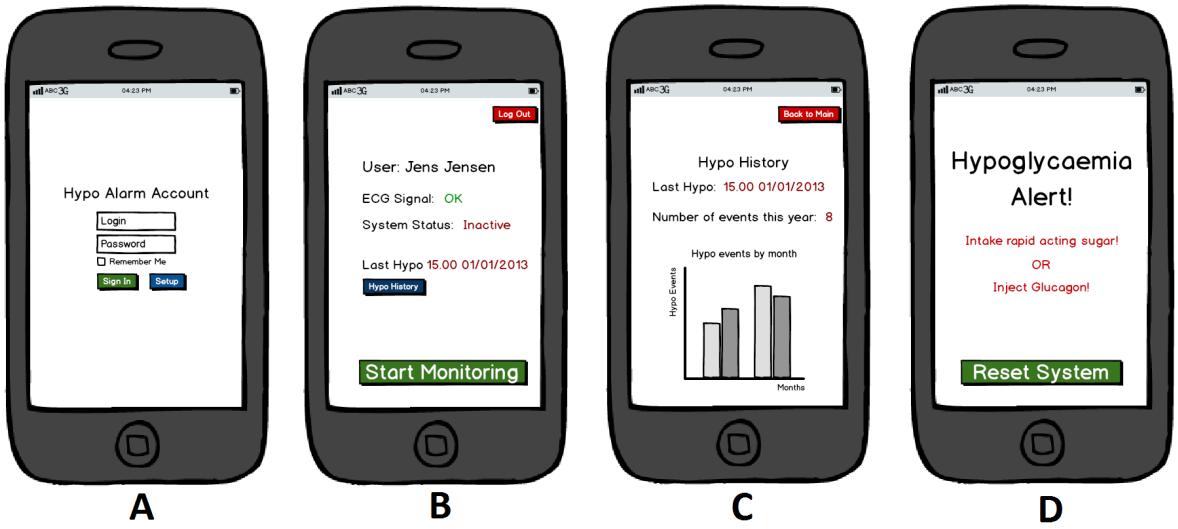

Figure 7.3: System Mock-up, first iteration. A: Login screen, B:main screen, C: hypoglycaemia history screen, D:hypoglycaemia alarm screen

Second Iteration

The second iteration of GUI development took place after the interviews where the use cases were identified. The new functions were incorporated into the GUI. During this iteration attention was given to make the system user friendly by following the "10 Usability Heuristics for User Interface Design" developed by Jacob Nielsen in 1994 [48].

The 10 Usability Heuristics are:

1. Visibility of system status- keeps the user informed on what is going on.
2. Match between system and the real world- use the users language, no system terms.
3. User control and freedom - Support emergency exit.
4. Consistency and standards- Use same words for same things
5. Error prevention- eliminate error prone conditions
6. Recognition rather than recall- minimize the users memory load.
7. Flexibility and efficiency of use- support expert users to speed up process, tailor frequent actions
8. Aesthetic and minimalist design- only show the relevant information
9. Help users recognize, diagnose, and recover from errors- error messages should be understandable, no codes
10. Help and documentation- provide help when needed

Figure 7.4 shows the GUI design that was created during second iteration of GUI development.

Screen A shows the login screen, the minor change was applied by making the primary action "Login" slightly bigger than the "Setup" secondary action.

Screen B was developed. Here the user can enter the One Time Key and the additional information needed to download the model and create a user. Help button will in accordance with principle 10 explain the user what the One Time Key is and what will happen once the user is created. The primary action "create" is marked green and is bigger than the secondary "cancel" action marked with red. Cancel was introduced to fulfill principle 3. In case of an error during user creation an error message is displayed, stating in everyday language what went wrong and how to fix it in accordance with principle 9.

Screens C and D show the main screen in monitoring state and inactive state. The colour marking was only kept on the items that show the state of the system. This was done to make the most important information stand out. The system status information was also separated from the general information and moved down on the screen. Once the system is in active state an animation of an ECG signal is shown and the system status changes to "monitoring". The "start monitoring" button changes colour to red and label to "stop monitoring". This was done to make it clear for the user what the system status is and fulfill principle 1.

Screen E shows the hypo history screen that can be accessed from the main screen The colour marking of text was removed as on main screen. A new button "full history" was added to support the functionality requirements identified in UC6 "View Full History". This button is coloured blue.

Screen F shows the full history screen.

Screen G shows the alarm screen that activates when a hypoglycaemia is detected. The colour coding of the text was removed as it is used only to communicate system status on the other screens and the system needs to be consistent in accordance with principle 4. Two new buttons were added "person unconscious guide" and "person conscious guide". These were added to fulfill the requirements identified in use case 9. The "Reset System" button is primary action on this screen and will be used the most; however the button is made small and coloured red. This change was added after the interviews. It was noted that if a hypoglycaemia event occurs and the primary user(person with diabetes) is unable to take action himself, help from others is needed. The application must provide guidance to the secondary user (helper). Helper is not familiar with the system and the functions that he might use need to stand out. "If I would pick up a screaming phone, I might automatically press the biggest green button, before reading the text" - stated one of the interviewees. Therefore the view guide buttons on this screen are much bigger and marked green. The smaller reset system button is still available for the primary user, but it is made clearly the "wrong choice" for the secondary user. In addition when the "reset system" button is pressed a confirmation message pops up asking to confirm the action in accordance with principle 5.

Screen H shows the Guide Screen that can be accessed from the alarm screen. The screen contains a picture showing what has to be done and explanatory text. The Guide contains a number of steps, the current step as well as the total number of steps is visible at all time. In the bottom of the screen there are "next" "previous" and "cancel" buttons. Cancel is the

smallest button marked red (principle 5). The other 2 buttons allow the user to navigate through the steps (principle 3).

In general the system uses no terms that are not known by the user in accordance with principle 2. The system has few functions and therefore meeting the principles 6 per default as the memory load is minimal. Principle 7 is also achieved per default as all system functions can be accessed in two clicks or less.

Third iteration
The third phase of the GUI design was based on results from a Heuristic Evaluation. This process involved having a single evaluator examine the interface and judge its compliance with the 10 heuristic principles. The evaluator used in this study was also interviewed during the use case identification process. The process of testing and the subsequent changes in GUI design are described in the test section.

7.7 System validation

The purpose of system validation is to determine if the designed system meets requirements defined in subsection 3.2.4. For easy reference the requirements are listed below:
R0: The system **shall** allow the user to protect personal information.
R1: The system **shall** analyse live ECG signal for risk of hypoglycaemia.
R2: The system **shall** sound an alarm if hypoglycaemia is detected.
R3: The system **shall** provide an action guide for stopping hypoglycaemia.
R4: The system **shall** show history of hypoglycaemia events as a chart.
R5: The system **shall** show full log of all hypoglycaemia events with dates and times attached.
R6: The system **shall** allow user to create a new user and load the patient specific model.

The functional requirements describe what functions the system should support. The system is validated by checking if the required functionality is supported. The list below validates each requirement. V1 is validation of R1, V2 of R2 and so on.

- **V0:** The user information is protected with username and password, that have to be entered before the main functions of the system are accessed.
- **V1:** The ability to analyse the risk of hypoglycaemia is provided by the prediction algorithm integrated into the system.
- **V2:** The system sounds an alarm and opens an alarm window when hypoglycaemia is detected.
- **V3:** From the alarm window 2 buttons are accessible that can be pressed by the user to access the guides.
- **V4:** From the main system window hypoglycaemia history window can be opened and the bar chart of the hypoglycaemia events is displayed together with additional information.

- **V5:** From the history window the full history log can be accessed, which is displayed in a new window, with date, and time of each event.
- **V6:** The system allows creation of a new user from the system setup window which can be accessed from the login window.

7.8 System Test

The testing process of the System was divided into two main parts. The firs part of testing was performed during the system development process. These tests included Unittest and Integrationtest. Unititest verifies the functionality of a single code section. Integration test verifies the correct function of components when they are combined. These two test are white box tests. During this study the tests performed during system development were not documented. [49]

The second part of testing consisted of System Testing. Here the complete system was tested to verify that it meets the requirements. This test falls within the scope of the black box. Typically system testing contains functionality test, usability and stability test [49]. For the purpose of this study the functionality test was not performed, instead the functionality was verified through system validation. The stability test was not performed as the system is a very early prototype. The main focus was given to the usability testing. For the purpose of this study it was chosen to conduct a Heuristic Evaluation. The main focus of this test was to assess the interface and uncover any usability problems. The following section will describe the method used for acceptance testing and the results that it provided.

7.8.1 Heuristic Evaluation

This study uses the methodology for conducting a heuristic evaluation developed by Jacob Nielsen [50]. Heuristic evaluation falls under the usability testing group and is directed at finding the usability issues in user interface design. Heuristic evaluation does not provide a systematic way to fix to the usability problems or a way to assess the quality of redesigns. However, heuristic evaluation explains the uncovered usability problems with reference to usability principles. These problems can then be corrected by following the guidelines for the violated principle. It is recommended to include at least 3 evaluators in order to identify the majority of usability problems as one person will never be able to find all the usability problems in an interface. However for this project only one evaluator was used. This was done due to the limited time frame of the project and the fact that the developed system is an early prototype.

Heuristic evaluation is conducted by having an evaluator inspect the interface alone. Typically the results of the evaluation session can either be verbalized by the evaluator or written as reports. For this study the evaluator was asked to comment the interface while going

through the system. The evaluator was not given help until he was clearly in trouble and have commented on the usability problem in question.

The heuristic evaluation session lasted for 30 minutes, due to the simplicity of the system. During the evaluation the evaluator went through the interface several times and checked if all the parts of the system interface fulfill the 10 heuristic principles. The evaluator was provided with a list of the 10 principles and examples of how these principles were successfully implemented on other applications [51]. During the session the evaluator was encouraged to consider any additional usability principles that come to mind.

The system could be evaluated by the evaluator with no assistance however such evaluation might be time consuming. It was chosen to aid the evaluator by providing him with typical task. These tasks were based on the Use cases identified in section 7.3.3. The tasks stated what had to be done, however did not list the steps necessary to get there. A list of tasks and principles provided to the evaluator can be found in appendix 10.2.

The output from using the heuristic evaluation method was a list of usability problems in the interface with references to those usability principles that were violated by the design in each case in the opinion of the evaluator. It was explained to the evaluator that it is not sufficient to simply say that he does not like something; he should explain why he does not like it with reference to the heuristics or to other usability principles.

7.8.2 Test Results

The heuristic evaluation revealed a set of usability issues in the GUI design. The uncovered issues as well as the implemented fixes are listed below.

Issue 1
On the login screen A on figure 7.4 usability principle 4 was violated. It was noted by the evaluator that during Create user task it was unclear that the "setup" button would trigger user creation procedure. It was expressed that most applications use a different term for this function. While "setup" is often used for utility functions such as sound adjustment.
Fix: Change button label from "setup" to the widely used "Create User".

Issue 2
On the Setup screen B on figure 7.4 usability principle 10 was violated. The evaluator expressed confusion on whether the username and password should be provided by the medical worker or if they need to be created by the evaluator. The help section contained no information on this subject.
Fix: Explain that username and password need to be created by the user himself.

Issue 3
On the main screen in inactive state C on figure 7.4 principle 4 was violated. Here under

system status "inactive" was used to state that the system is not currently monitoring the ECG signal. According to the evaluator the term "inactive" as a System status did not implicitly explain what was inactive. The user thought that this referred to the connection to the Internet. It was only clear that the system status referred to the ECG monitoring once start monitoring button was pressed and the screen switched to monitoring state.
Fix: Change label from "inactive" to "Not Monitoring ECG". To be consistent with the use of terms for states main screen in monitoring state D on figure was also modified. The label was changed from "monitoring" to the more explanatory "Monitoring ECG".

Issue X
The last issue of uncovered during the heuristic evaluation could not be linked to any of the 10 heuristics as it is a functionality issue. This issue was uncovered by the evaluator due to the fact that the evaluator is a domain expert, working with diabetes as a nurse as well as having diabetes. It was noted that the system detects hypoglycaemia at different levels for different persons; therefore it might be useful to inform the user what the hypoglycaemia detection threshold is.
Fix: Add a new information line "Hypoglycaemias are detected at: XX mmol/l" on the main screen.

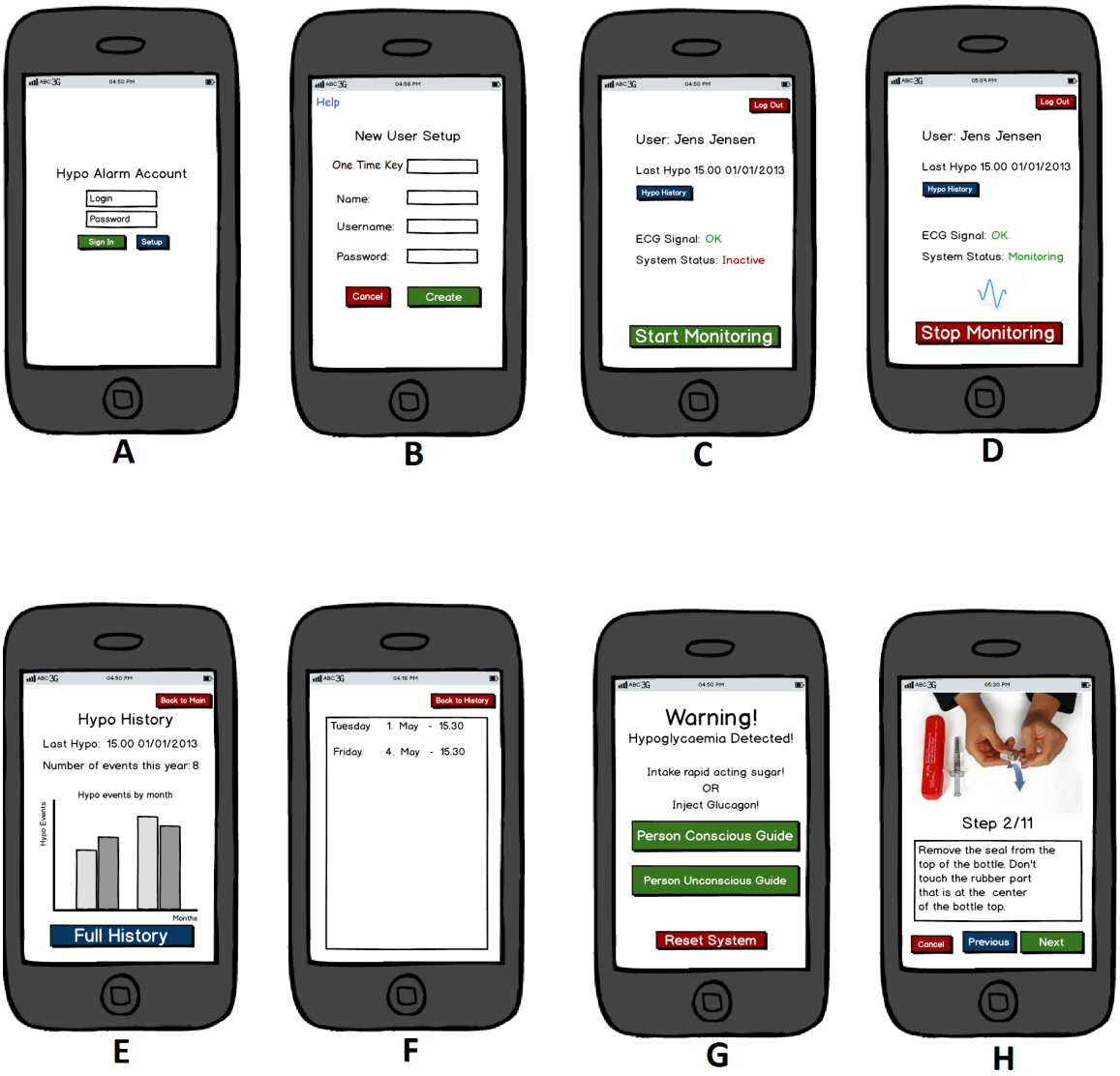

Figure 7.4: System Mock-up, second iteration. A:Login Screen, B: Setup screen, C:main screen(inactive), D:main screen (active), E: hypoglycaemia history screen, F: full hypoglycaemia history screen, G: hypoglycaemia alarm screen, H: guide screen.

Part III

Synthesis

8 Discussion

The discussion chapter is divided into three separate sections : initial analysis discussion, model discussion and application discussion. In these three sections the main design decisions will be discussed.

8.1 Initial analysis discussion

For the data pre-processing BG measurements and the ECG features were linked together. This can be achieved by interpolating the BG measurements or by filtering the ECG measurements. The first possibility is to interpolate the BG measurements and extract a BG point for every ECG point. The advantage of this approach is that all the ECG data is used. However interpolation introduces an error term. Another possibility is to take a BG measurement and specify a time range around the time of BG measurement. Then extract the median of the ECG feature measurements that fall within this time range, obtaining a single ECG point that corresponds to a single BG point. The main drawback of this approach is that a large portion of ECG signal is excluded. However the advantage is that an error term is not introduced. Additionally interpolation is generally more complex than filtering. For the initial analysis filtering approach was used as it was faster to implement and a large amount of BG measurements was available to analyse on as all the patients and visits were included in the analysis sample.

8.2 Model discussion

Quadratic spline interpolation
For the hypoglycaemia prediction model the ECG and BG points were linked by means of quadratic interpolation. Number of BG measurements for a single patient was small and therefore filtering approach was not a viable option in this case. It was chosen to use quadratic

interpolation and not linear interpolation as discontinuity of the linear interpolation might not fit with the behavior of BG which changes smoothly as a result of gradual absorption of insulin or sugar or action of other hormones. Interpolation introduces an error. The maximal error was estimated in section 5.2.1 to 0.34mmol/l. However the maximal error scenario might not be possible as the immediate shift in BG change rate from +0.068 to -0.068 is improbable. Additionally the spline created will not be a horizontal line as the splines on each side will effect its shape, making this maximal error scenario even less probable. For a spline to be horizontal four BG measurements need to have same values. Finally the time between measurements in the hypoglycaemia range is less than 10 minutes for the majority of measurements.

Patient specific model
Tests were performed to see whether the variance of ECG features for BG ranges was higher for the entire subject group than for a single patient over two sessions. It was observed that the intra patient variance in the features was much smaller than the variance of the combined data set. Based on this observation, it was chosen to concentrate the efforts on designing a patientspecific model.

Train model on one session
Model performance was found to differ depending on what session was used as a training set. The model detected hypoglycaemia and produced no false positives regardless of the session that was used to train the model on only for patient 6. For patient 5 the model produced false negatives during both control visits. And for the remaining patients the model only performed well when using one session to train on and not the other one. Additionally different features were picked for different sessions of the same patient. This means that the features that are best predictors in one of the sessions are not necessarily the best predictors for another session, suggesting that the model is overfitting to a single visit. Given current data such problem could be solved by using both experiment and both control visits to train the model on, however, then there will not be any data to test the model on. The data of a single visit should be viewed as a single sample (ECG and BG measurements that are close to each other cannot be assumed to be uncorrelated). For this study we have only two control and two experiment visits. The model should be able to detect a completely unseen hypoglycaemia event. If points of both sessions are included in the model training set, even though a portion of the points is reserved for testing, the model performance on the test set might be overoptimistic. This is because during model training points that are very close to the "new" points are seen by the algorithm. If the model is trained and validated on points of the same visits, which are very similar, a multi feature model will produce a perfect split even on the "new" measurements from the same set. For this reason the decision was made to only include one session to train on.

Limit model to one feature
It was chosen not to use multiple features in a model. This was done due to the observation that a combination of two features will often provide a model that perfectly separates the hypoglycaemia measurements from no hypoglycaemia measurements even when using a

validation set that is larger than the train set (this is true only when training the model on a single session). When a multifeature model is trained on one visit e.g V1 and tested on a different visit e.g V3 the performance is very poor. Limiting to 1 feature lowers the chance of overfitting to a single visit. To be able to create a multifeature model it is necessary to have at least 1 more session to train on (3 sessions in total). Then the model can be allowed to select more than 1 feature. Such a model can be trained on two sessions validated on a holdout sample and finally tested on the third session. When the model is trained on two visits the risk of overfitting is much lower, as the model will most likely choose features that fit both visits. There is still a risk that these two visits are not representative so the model must use a leave one session out approach. Here each of the visits is used in succession to test the model on. Ideally a fourth session is desired that could be used as a validation set.

Use two models: Model 1 and Model 2

For this study it was chosen to use two models. Model 1 was designed to be a good hypoglycaemia detector. Model 2 was designed to detect ECG changes produced mainly by the hypoglycaemia and not insulin. Ideally, Model 2 alone can be used to detect hypoglycaemia induced ECG changes. For this to work Model 2 should be allowed to include multiple features and be trained on multiple sessions. Then a combination of features that change mainly in response to hypoglycaemia will be identified. However in present study Model 2 was limited to selecting a single most important feature and was trained on a single session (these decisions are addressed earlier in the discussion). The resulting model produced a very small amount of false positives on experiment data shortly after the start of insulin infusion and before the BG fell below the hypoglycaemia threshold. This suggests that Model 2 features are to some degree affected by insulin. For the purpose of this study it was chosen to develop Model 1 that was good at detecting ECG changes during insulin induced hypoglycaemia. Model 1 produced no false positives during experiment visits and hence the combined model did not produce false positives.

Chosen ECG features

Features that were chosen for the models, of different patients, that detected hypoglycaemia during experiment visits and produced no false positives during control visits are summarized in the following text. It was found that feature 263(aVR: TP-interval peak amplitude), 388 (I: TP-interval peak amplitude) and 99(V1: RP-interval peak amplitude) could be used to distinguish between a real hypoglycaemia and purely the action of insulin, these features were used for model 2. These features changed during control visits and experiment visits, however, during experiment the change was much more profound than during control. This implies that model 2 features are effected by both the action of insulin and the hypoglycaemia as such. For model 1 features were identified that could identify the hypoglycaemia, these features might be affected by both insulin and hypoglycaemia, however the action of insulin seems to be predominant; if model 1 is used to classify control visit measurements false positives will be produced. The model 1 features are : 7(PR interval), 61(II: Maximum S-wave peak

amplitude), 213(V5: TP-interval peak amplitude),55(II: ST-level at the E point).

Model performance
The overall model performance is unclear as the model was trained on events of insulin induced hypoglycaemia , the ECG changes during spontaneous hypoglycaemia might or might not be detected by the current model features, however if more sessions are included it might be possible to identify features that change mainly in response to hypoglycaemia. Such a model can be based on Model 2 which will be trained on multiple sessions and allowed to select multiple features.

8.3 Application discussion

The developed mobile application prototype holds all the major functions that were identified in the requirement specification process of present study. For the purpose of this study the system was programmed in Java; Swing was used to develop a GUI. Such code can easily be converted to Android: the system logic can remain the same, and only Swing GUI has to be converted to Android code. It was decided not to use Android code from the start of the project due to the time limitations.

Typical system tests include usability, functionality and stability tests (see section 7.8). Application developed during this study should be viewed as an early prototype and therefore the system testing was aimed at discovering usability problems. The complete system was envisioned to have three parts: a mobile application, a PC application and a webserver. Due to the time limitations of the current study only the mobile application developed.

The final system usability test included a single evaluator. This is not sufficient as a single evaluator will almost never be able to find all the usability problems in a system. However given the small amount of system functions and the limited time of the study, a decision was made to include a single evaluator. The application functionality was tested by validating the functions specified in the requirements. The aim with testing was to see if there are any usability problems associated with the design of mobile application. The results obtained from performed test uncovered some usability problems that were fixed immediately. If given more time the system development would enter the next iteration where a PC application and a server would be developed.

9 Conclusion

The problem that the present study attempted to solve was to develop a hypoglycaemia detection model based on the ECG and integrate it into a mobile application that can be used by people with diabetes. To solve the problem statement three aims were identified. For easier reference the three aims are listed below:

- **Aim 1:** identify statistically significant changes in the ECG features during insulin induced hypoglycaemia
- **Aim 2:** investigate modeling techniques that can be used to detect hypoglycaemia based on ECG features
- **Aim 3:** develop a mobile application that can be used by the person with diabetes for detection of hypoglycaemia

Aim 1 conclusion: Features that change during hypoglycaemia were identified using Mood's test. It was noted that a step size of 0.01 was overexcessive as the resulting the p-values remained the same over a range of 0.1 mmol/l, meaning that a step size of 0.1 mmol/l is more appropriate. The results of this initial analysis suggested that changes in ECG features might be used to create a hypoglycaemia detection model.

Aim 2 conclusion: The developed algorithm performed well in some conditions but produced false positives and false negatives in other conditions. The reasons behind this performance were discussed in discussion section together with the possible solutions. In conclusion it can be said that given more data a reliable hypoglycaemia detection model might be developed.

Aim 3 conclusion: The developed application received good comments from the evaluator and only a few usability issues were identified which were fixed shortly after the evaluation. Great attention was given to make the system comply with the 10 design principles. However some usability problems might have been overlooked during development process by the developer and during system evaluation by a single evaluator. Therefore if given adequate time more evaluators should be included to screen the system for additional usability problems.

10

Bibliography

[1] International Diabetes Federation. Diabetes statistics. http://www.idf.org/node/23640, 2011.

[2] Diabetesforeningen. Diabetes i danmark. http://www.diabetes.dk/Rundt_om_diabetes/Diabetes_i_tal/Diabetes_i_Danmark.aspx, 2011. Downloaded: 10-10-2012.

[3] International Diabetes Federation. The economic impacts of diabetes. http://www.idf.org/diabetesatlas/economic-impacts-diabetes, 2009. Downloaded: 14-10-2011.

[4] PE. Cryer. Hypoglycaemia: The limiting factor in the glycaemic management of type i and type ii diabetes. *Diabetologia.*, 45(937-948), 2002.

[5] Tim Holt and Sudhesh Kumar. *Diabetes fra A til Z.* Fadl's Forlag, 2010.

[6] Stefan Silbernagl & Agamemnon Despopoulos. *Color Atlas of Physiology 4th ed.* Thieme, 1991.

[7] Jannik Hilsted, Knut Borch-Johnsen, and Jens Christiansen. *Diabetes.* Munksgaard Danmark, 2007.

[8] World Health Organization. About diabetes. http://www.who.int/diabetes/action_online/basics/en/index.html, 2011. Downloaded: 14-10-2011.

[9] Mark A. Creager Joshua A. Beckman and Peter Libby. Diabetes and atherosclerosis: epidemiology, pathophysiology, and management. *JAMA*, 287:2570–2581, 2002.

[10] IRL B. HIRSCH. Type 1 diabetes mellitus and the use of flexible insulin regimens. *American Family Physician*, 2(3), 1999.

[11] Clouse RE Lustman PJ IAnderson RJ, Freedland KE. The prevalence of comorbid depression in adults with diabetes. *Diabetes Care*, 24(1069-1078), 2001.

[12] The Diabetes Control and Complications Trial Research Group. The effect of intensive treatment of diabetes on the development and progression of long-term complications in insulin-dependent diabetes mellitus. *N Engl J Med*, 329(977-986), 1993.

[13] Kerr D and Richardson T. *Hypoglycaemia in Clinical Diabetes*. Wiley, 2007.

[14] Philip E. Cryer, Stephen N. Davis, and Harry Shamoon. Hypoglycemia in diabetes. *DIABETES CARE*, 26(6):1902–1912, 2003.

[15] Spyridon Bakatselos. Hypoglycemia unawareness. *Diabetes Research and Clinical Practice*, 93:1902–1912, 2011.

[16] Frier BM. McAulay V, Deary IJ. Symptoms of hypoglycaemia in people with diabetes. *Diabet Med*, 18:690–705, 2001.

[17] Karamitsos DT er al Bacatselos SO. Hypoglycemia unawareness in type i diabetic patients under conventional insulin treatment. *Diab Nutr Metab*, 8:267–275, 1995.

[18] Schwartz NS et al Boyle PJ. Plasma glucose concentrations at the onset of hypoglycemic symptoms in patients with poorly controlled diabetes and in nondiabetics. *N Engl J Med*, 318:1487–92, 1988.

[19] Sherwin RS Amiel SA. Effect of intensive insulin therapy on glycemic thresholds for counterregulatory hormone release. *Diabetes*, 37:901–907, 1988.

[20] Cryer PE. Segel SA, Paramore DS. Hypoglycemia-associated autonomic failure in advanced type 2 diabetes. *Diabetes*, 51:724–733, 2002.

[21] Cryer PE. Raju B. Loss of the decrement in intraislet insulin plausibly explains loss of the glucagon response to hypoglycemia in insulin-deficient diabetes. *Diabetes*, 54:757–764, 2005.

[22] Cryer PE. Dagogo-Jack SE, Craft S. Hypoglycemia-associated autonomic failure in insulin-dependent diabetes mellitus. *Clin Invest*, 91:819–28, 1993.

[23] The United Kingdom Prospective Diabetes Study Group. U.K. prospective diabetes study. 16. overview of 6 yearsâĂŹ therapy of type ii diabetes: a progressive disease. *Diabetes*, 44:1249–1258, 1995.

[24] Bryan RM. Cerebral blood flow and energy metabolism during stress. *Am J Physiol*, 259:269–280, 1990.

[25] Braatvedt GD et al. Splanchnic blood flowinman: evidence for mediation via a beta-adrenergic mechanism. *Clin Scil*, 84:201–207, 1993.

[26] Hensel H Allwood MJ. Muscle and skin blood flow in the human forearm during insulin hypoglycaemia. *J Physiol*, 147:269–273, 1959.

[27] Craig KJ Thompson I Swainson CD Frier BM. Patrick AW, Hepburn DA. The effects of acute insulin-induced hypoglycaemia on renal function in normal human subjects. *Diabet Med*, 6:703–708, 1989.

[28] Hepburn DA Dargie HJ Barnett E Frier BM Fisher BM, Gillen G. Splenic responses to acute insulin-induced hypoglycaemia in humans. *Clin Scil*, 78:469–474, 1990.

[29] Frier BM. Wright RJ. Vascular disease and diabetes: is hypoglycaemia an aggravating factor? *Diabetes Metab Res Rev*, 24:353–363, 2008.

[30] Hollander W Judson WE. The effects of insulin-induced hypoglycemia in patients with angina pectoris; before and after intravenous hexamethonium. *Am Heart J*, 52:198–209, 1956.

[31] Ireland RH Macdonald IA Heller SR Robinson RT, Harris ND. Changes in cardiac repolarization during clinical episodes of nocturnal hypoglycaemia in adults with type 1 diabetes. *Diabetologia*, 47:312–315, 2004.

[32] Salmela PI et al. Koivikko ML, Karsikas M. Effects of controlled hypoglycaemia on cardiac repolarisation in patients with type 1 diabetes. *Diabetologia*, 51:423–426, 2008.

[33] Ireland RH Lee S Newman C Heller SR. Robinson RT, Harris ND. Mechanisms of abnormal cardiac repolarization during insulin-induced hypoglycemia. *Diabetes*, 52:1469–1474, 2003.

[34] Fazekas JF Parrish AE, Sugar SJN. A relationship between electrocardiographic changes and hypokalemia in insulininduced hypoglycemia. *Am Heart J*, 43:815–820, 1952.

[35] Kerp L Peterson K-G, Schluter KJ. Regulation of serum potassium during insulin-induced hypoglycaemia. *Diabetes*, 31:615–617, 1982.

[36] P I Salmela J S Tapanainen A Ruokonen T Seppanen H V Huikuri J S Perkiomaki M L Koivikko, M Karsikas. Effects of controlled hypoglycaemia on cardiac repolarisation in patients with type 1 diabetes. *Diabetologia*, 51:426–435, 2007.

[37] FDA. Celexa (citalopram hydrobromide) tablets and oral solution. http://www.fda.gov/Safety/MedWatch/SafetyInformation/ucm271275.htm, 2012.

[38] FDA. Celexa (citalopram hydrobromide) tablets and oral solution. http://www.fda.gov/Drugs/DrugSafety/ucm269086.htm, 2012.

[39] wikipedia. Ecg signal. http://upload.wikimedia.org/wikipedia/commons/9/9e/SinusRhythmLabels.svg, 2012. Downloaded: 10-12-2012.

[40] wikipedia. Ecg leads. http://upload.wikimedia.org/wikipedia/commons/7/70/ECG_12derivations.png, 2012. Downloaded: 10-12-2012.

[41] J Chambers J Crawshaw. *A concise advanced levek statistics with worked examples*. 2002.

[42] Janet Tamada Tomothy Dunn, Richard Eastman. Rates of glucose change measured by blood glucose meter and the glucowatch biographer during day, night, and around mealtimes. *Diabetes Care*, 27(61-66), 2004.

[43] Tom M. Mitchell. *Machine Learning*. McGraw-Hill Science, 1997.

[44] F. E. Harrell. *Regression Modelling Strategies With Applications to Linear Models and Logistic regression and Survival Analysis*. Springer, 2001.

[45] A. Azzalini. *Statistical Inference Based on the likelihood*. Chapman & Hall, 2002.

[46] Gary L. Myers Hubert W. Vesper, Enada Archibold. Assessment of trueness of glucose measurement instruments with different specimen matrices. *Clinica Chimica Acta*, 358(68-74), 2005.

[47] Craig Larman. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process (2nd Edition)*. Prentice Hall PTR, 2002.

[48] Jakob Nielsen. 10 usability heuristics for user interface design. http://www.nngroup.com/articles/ten-usability-heuristics/, 2005.

[49] Lee Copeland. *A practitioner's guide to software test design*. Artech House, 2004.

[50] Jakob Nielsen. How to conduct a heuristic evaluation. http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/, 1995.

[51] Theresa Neil. Designing web interfaces. http://designingwebinterfaces.com/6-tips-for-a-great-flex-ux-part-5, 2009.

Part IV

Appendix

10.1 Interview questions

The interviews were conducted in danish. Listed below are the interview questions, which are divided into two parts: clinical and technical.

Klinisk:

1. Hvor længe havde du diagnosen Diabetes Type 1?
2. Hvor mange hypoglykaemiske anfald får du om året?
3. Hvor og hvornår får du typisk hypoglykaemiske anfald?
4. Har du nogensinde haft et hypoglykaemisk anfald hvor du skulle bruge hjælp af andre?
5. Er en hypoglykaemisk anfald noget du er "bange" for?
6. Hvis du får en svær hypoglykæmisk anfald, ved din familie/venner hvad de skal gøre for at hjaelpe dig?
7. Hvad gør du, nu for at undgå en hypoglykæmi?

Teknisk:

1. Bruger du nogle diabetes apps på din mobil?
2. Hvorfor/ hvorfor ikke?
3. Hvad synes du om ideen at lave en hypoglykæmi alarm der kører fra din mobil?
4. Systemet vil overvåge ECG og give en alarm hvis hypoglykæmi er på vej. Er der andre funktioner som du synes skal med i systemet. F.eks en logbog der indeholder tidspunkter for alle hypoglykæmiske anfald; eller en lille guide der popper op når man får hypoglykæmi, som kan bruges af den persom der skal give en glucagon indsprøjtning.

10.2 Documents used during heuristic evaluation

10.2.1 Opgaver

1. Opret ny bruger
2. Login med ny bruger
3. Start hypoglykæmi monitorering
4. Åbn en guide til behandling af svær hypoglykæmi på en bevidsløs person
5. Åbn en guide til behandling af mild hypoglykæmi.
6. Kig på hypoglykæmi historie
7. Kig på detaljeret hypoglykæmi log.

10.2.2 10 design Principper

1. Visibilitet af systemets status
Systemet skal altid informere brugeren om hvad der sker, hva godt timet feedback.
Eksempel: Picnic. En loadingbar vises mens application loader.

Figure 10.1: Picnic loadingbar

2. Forbindelse mellem systemet og det virkelige verden.
Systemet skal bruge brugerens sprog; ord, vendinger og begreber skal være en del af brugerens hverdag. Ingen system orienterede termer må bruges. Information skal præsenteres på en naturlig måde.
Eksempel: ITunes. er organiseret som et "library" der indeholder mediebibliotek: musik, film, shows, audibooks. Under "library" er der en ŤstoreŤ hvor man kan købe media.

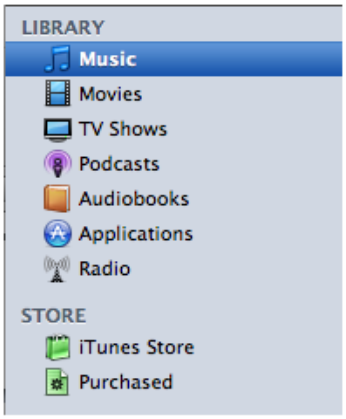

Figure 10.2: Itunes library bar

3. Brugerstyring og frihed
Brugere kan tit vælge en systemfunktion ved en fejl. Systemet skal derfor have tydelige Ťnø-dudgangŤ knapper.
Eksempel: CollabFinder. Search function kan enten køres eller slukkes ved at trykke på cancel.

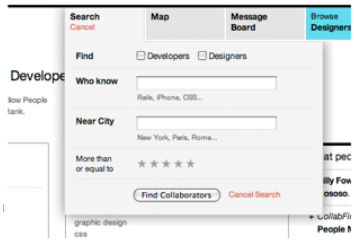

Figure 10.3: CollabFinder navigation

4. Sammenhæng og standarder

Brugere bør aldrig være i tvivl, om forskellige ord i forskellige situationer betyder det samme eller ej. Systemet skal være konsekvent med design, knappeplacering osv.

Eksempel: Word, Excel, og PowerPoint alle burger same toolbar, same menuer: menu options: Home, Insert, Page LayoutĚ

Figure 10.4: CollabFinder navigation

5. Genkendelse frem for hukomelse

Det er vigtigt at have et god interface design der forebygger fejl. Man skal dobbelttjekke vigtige handlinger f.eks hvis brugeren trykker på en delete all files knap, er det vigtingt at spørge brugeren om han er sikker.

Eksempel: Yammer. Her er Ťprimary actionŤ knap meget større end Ťsecondary actionŤ knap.

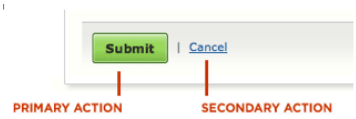

Figure 10.5: Yammer primær og sekundær aktioner

6. Fejl prevention

Belastning på brugerens hukomelse bør minimeres. Man skal ikke kunne huske hvad der skete

i step 1 af processet for at udføre step 10.

7. Fleksibilitet og effektivitet
Systemet skal understøtte eksperbrugere. (genveje der kan bruges af ekspertbrugere til at arbejde hurtigere)
Eksempel: Trykke på Ctrl+S for at gemme isteted for at navigere igennem menuer.

8. Æstetisk og minimalistisk design
Systemet skal ikke vise mere information end der er brug for.

9. Hjælp brugere med at spotte og rette fejl.
Fejl meddelelser skal srives i hverdagssprog (ingen kode). Der skal stå præcist hvad problemet er og hvordan den kan rettes.
Eksempel: Digg 9.0 giver feedback med instruktioner hvis noget går galt.

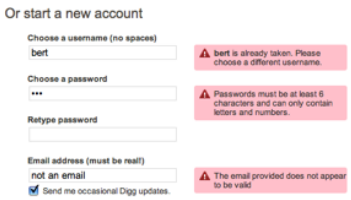

Figure 10.6: Yammer primær og sekundær aktioner

10. Hjælp og dokumentation
En god system bør kunne bruges uden en manual. Men i nogle tilfælde er det en fordel at have en nem tilgængelig hjælp function.
Eksempel: Skype har en help knap på toolbaren der altid er synlig.

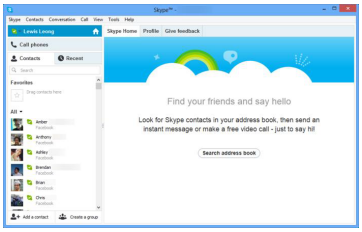

Figure 10.7: Skype hjælp funktion

10.3 Mood's test procedure

The test statistic is calculated in following steps:

1. The sample median M of combined samples is calculated
2. Number of observations above and below M in each sample is counted and inserted into table 10.3.
3. The expected cell counts are computed using the formula:
$$\mathbb{E}_{ij} = \frac{(R_i - C_j)^2}{N}$$
where E_{ij} =the expected frequency for the cell in row i, column j. R_i =the sum of frequencies in row i. C_i =the sum of frequencies in column j. N =the sum of frequencies of all cells.
4. The chi-squared statistic is calculated by taking each observation count O_i, subtracting the expected value E_i, then squaring this difference. The differences are divided by E_i and summed:
$$\chi^2 = \sum_i \frac{(O_i - E_i)^2}{E_i}$$
If number of observations in any of the table cells is below 5, Yate's correction is applied
$$\chi^2 = \sum_i \frac{(\mid O_i - E_i \mid -0.5)^2}{E_i}$$
The larger the difference between observed and expected, the larger the test statistic becomes and the more likely we are to reject the null hypothesis.
5. Degrees of freedom are calculated by using this formula: df=(Number of rows-1)*(Number of columns-1)
6. Reject the H0 if the obtained p value is below 0.05.

10.4 Logistic Regression

Before describing the logistic regression, the main idea about the general linear regression model will be described. Let Y denote the random response variable and let $\boldsymbol{X} = [X_1, \ldots, X_k]$ be a vector containing the k explanatory variables $X_1, \ldots X_k$.

	y_1	y_2	Total
$< M$	a	b	$a+b$
$>= M$	c	d	$c+d$
Total	$a+c$	$b+d$	N

Table 10.1: Two-by two contingency table.

The idea in general linear regression models is to link the mean value of the response variable to a weighted sum of the explanatory variables through some function:

$$\mathbb{E}[Y|\boldsymbol{X}] = g^{-1}(\boldsymbol{X}\beta)$$

where $\beta = [\beta_0, \ldots, \beta_k]^\top$ is a set of parameters to be estimated and g is a known function called the *link function*, such that

$$g\big(\mathbb{E}[Y|\boldsymbol{X}]\big) = \boldsymbol{X}\beta = \beta_0 + \beta_1 X_1 + \cdots + \beta_k X_k$$

Logistic regression is used to describe a binomial event based on set of predictor variables. Let Y be the binomial random outcome of the event, with $Y = 1$ denoting the occurrence of the event and $Y = 0$ denoting the absence of the event. This means that $Y|\boldsymbol{X} \sim \text{Bin}(1, \pi)$ where $\pi = Prob(Y = 1|\boldsymbol{X})$.

The logistic regression model is defined by the link function given as the *logit* function,

$$\text{logit}(\pi) = \log\left(\frac{\pi}{1-\pi}\right) = \boldsymbol{X}\beta \tag{10.1}$$

Solving for π gives logistic regression model

$$\pi = Prob(Y = 1|\boldsymbol{X}) = \frac{1}{1+\exp(-\boldsymbol{X}\beta)} = g(X) \tag{10.2}$$

$g(X)$ is called the logistic function, see figure 10.8. This function has an unlimited domain but the range is restricted to the interval $(0; 1)$, meaning that the logistic function is well suited to evaluate the probability of a binomial event. Parameters in logistic regression have an intuitive effect. Let $\tilde{\boldsymbol{X}} = [X_1, \ldots, X_j + 1, \ldots, X_k]$. So that that $\tilde{\boldsymbol{X}}$ contains the same predictors as $\boldsymbol{X}$ except for a unit change in X_j. Let

$$\tilde{\pi} = \frac{1}{1+\exp(-\tilde{\boldsymbol{X}}\beta)},$$

then the following hold

$$\begin{aligned}\text{logit}(\tilde{\pi}) - \text{logit}(\pi) &= \tilde{\boldsymbol{X}}\beta - \boldsymbol{X}\beta \\ &= \beta_j(X_j + 1) + C - (\beta_j X_j + C)\end{aligned}$$

provided the predictors $X_1, \ldots, X_{j-1}, X_{j+1}, \ldots, X_k$ are held constant and C is defined as unchanged parameters and predictors(constant). For each predictor X_j, the corresponding parameter β_j is the change in the probability of the event $Y = 1$ occuring per unit change in X_j.

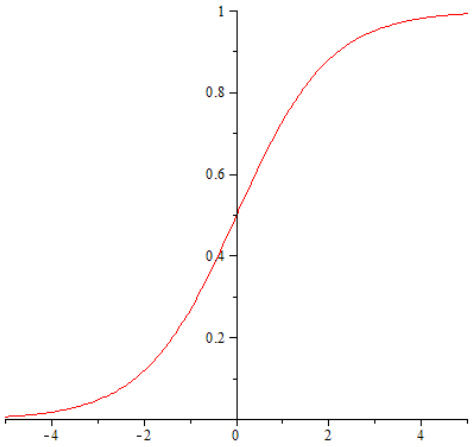

Figure 10.8: The logistic function

10.4.1 Parameter estimation

Suppose a set of n samples $y_1, \ldots, y_n$ are observed, each sampled from a random binomial distribution where $Y_i \sim \text{Bin}(1, \pi_i)$. In addition assume that for each y_i a set of explanatory variables $X_{1i}, \ldots, X_{ki}$ are known. To estimate the regression parameters β, maximum likelihood method method can then be applied. The density function of a binomial random variable is given as

$$f(y) = \binom{1}{y} \pi^y (1-\pi)^{(1-y)}$$

where π is given as in equation (10.2) and $y = 1 \vee y = 0$, this gives a likelihood equation as follows:

$$L(\beta|\mathbf{X}, \mathbf{Y}) = \prod_{i=1}^{n} \pi^{y_i} (1-\pi)^{1-y_i}. \tag{10.3}$$

The techniques employed to find the maximum likelihood estimates fall under the general label numerical analysis. There are several methods of numerical analysis, but they all follow a similar series of steps. First, some initial estimates of the parameters are picked. Then the likelihood of the data given these parameter estimates is computed. Then the parameter estimates are slightly improved and the likelihood of the data is recalculated. This process is repeated the parameter estimates do not change much (a change of 0.001 is small enough for a stop criterion). The optimization method used by MATLAB is Newton-Raphson.

To assess the maximum likelihood estimates of the regression parameters various test statistics exist. A test statistic which by default is reported by MATLAB is the Wald statistic. The Wald statistic is defined by

$$z = \frac{\hat{\beta}_i}{\hat{\text{SE}}(\hat{\beta}_i)}$$

where $\hat{\mathrm{SE}}(\hat{\beta}_i)$ is the estimated standard error of $\hat{\beta}_i$, defined by means of the observed Fisher information. Null hypothesis is that $\beta_i = 0$, significance level a was set to 0.05.

Table 10.2: Features included in present study

Feature Nr.	ECG feature name	Units
2	P-wave offset	mV
4	QRS duration	ms
5	QT interval	ms
6	QTc Bazett	ms
7	PR interval	ms
9	Avg RR interval	ms
10	QTc Fredericia	ms
11	QTc Framingham	ms
12	Morphology Combination Score x 1000	score
13	T-wave Morphology score	score
14	I: P-wave peak amplitude	mV
31	I: Minimum ST-level	mV
32	I: Maximum ST-level	mV
35	I: Maximum R-wave peak amplitude	mV
38	I: TP-interval peak amplitude	mV
46	II: R-wave duration	ms
47	II: S-wave peak amplitude	mV
55	II: ST-level at the E point	mV
56	II: Minimum ST-level	mV
57	II: Maximum ST-level	mV
58	II: Minimum ST-level	mV
60	II: Maximum R-wave peak amplitude	mV
61	II: Maximum S-wave peak amplitude	mV
63	II: TP-interval peak amplitude	mV
85	III: Maximum R-wave peak amplitude	mV
88	III: TP-interval peak amplitude	mV
99	V1: RP-interval peak amplitudee	mV
107	V1: Maximum ST-level	mV
110	V1: Maximum R-wave peak amplitude	mV
112	V1: T-wave peak amplitude	mV
135	V2: Maximum R-wave peak amplitude	mV
146	V3: R-wave duration	ms
147	V3: S-wave peak amplitude	mV
148	V3: S-wave duration	ms
149	V3: RP-interval peak amplitude	mV
155	V3: ST-level at the E point	mV
156	V3: Minimum ST-level	mV
157	V3: Maximum ST-level	mV
158	V3: Minimum ST-level	mV
160	V3: Maximum R-wave peak amplitude	mV
161	V3: Maximum S-wave peak amplitude	mV
162	V3: T-wave peak amplitude	mV
163	V3: TP-interval peak amplitude	mV
171	V4: R-wave duration	ms
172	V4: S-wave peak amplitude	mV
173	V4: S-wave duration	ms
174	V4: RP-interval peak amplitude	mV
181	V4: Minimum ST-level	mV
182	V4: Maximum ST-level	mV
183	V4: Minimum ST-level	mV
185	V4: Maximum R-wave peak amplitude	mV
186	V4: Maximum S-wave peak amplitude	mV
187	V4: T-wave peak amplitude	mV
188	V4: TP-interval peak amplitude	mV
196	V5: R-wave duration	ms
197	V5: S-wave peak amplitude	mV
207	V5: Maximum ST-level	mV
210	V5: Maximum R-wave peak amplitude	mV
211	V5: Maximum S-wave peak amplitude	mV
213	V5: TP-interval peak amplitude	mV
221	V6: R-wave duration	ms
222	V6: S-wave peak amplitude	mV
235	V6: Maximum R-wave peak amplitude	mV
236	V6: Maximum S-wave peak amplitude	mV
238	V6: TP-interval peak amplitude	mV
256	aVR: Minimum ST-level	mV
257	aVR: Maximum ST-level	mV
260	aVR: Maximum R-wave peak amplitude	mV
262	aVR: T-wave peak amplitude	mV
263	aVR: TP-interval peak amplitude	mV
282	aVL: Maximum ST-level	mV
285	aVL: Maximum R-wave peak amplitude	mV
296	aVF: R-wave duration	ms
297	aVF: S-wave peak amplitude	mV
307	aVF: Maximum ST-level	mV
310	aVF: Maximum R-wave peak amplitude	mV
311	aVF: Maximum S-wave peak amplitude	mV
313	aVF: TP-interval peak amplitude	mV