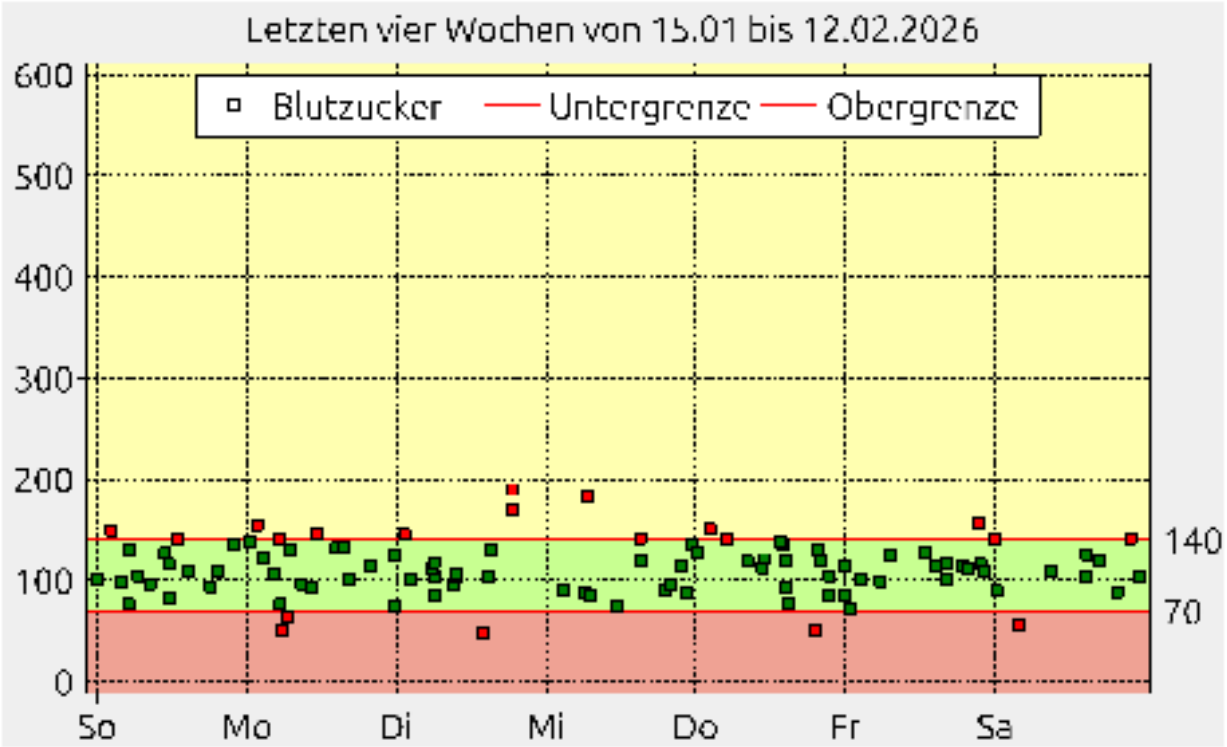


GlucoReader

Tool zum Auslesen und Anzeigen von Messdaten aus dem Glucometer

Grafik nach Tag gruppiert



Wochentag	Anzahl	Mittelwert	Std.-Abw.	Minimum (mgdl)	Maximum (mgdl)
Montag	18	111	29	50	155
Dienstag	14	115	34	49	190
Mittwoch	12	109	29	74	182
Donnerstag	17	114	24	50	151
Freitag	15	113	19	72	156
Samstag	9	104	23	57	142
Sonntag	15	111	20	76	150

Inhaltsverzeichnis

Allgemein.....	3
Übersicht.....	3
Version.....	3
Datensicherheit.....	3
Lizenz.....	3
Daten am Glucometer.....	3
Installation.....	5
Basis.....	5
Mittels GIT.....	5
Aus den Zip's.....	5
FPC-USB-HID.....	5
GlucoReader.....	6
Installation in Lazarus.....	6
Projekt in Lazarus kompilieren.....	7
Verwendete Verzeichnisse.....	7
Erster Start.....	8
Erstellen der Datenbank.....	8
Mit Glucoreader arbeiten.....	11
Anzeige – Analyse.....	12
Messgerät – Daten einlesen.....	14
Datenbank.....	15

Allgemein

Wichtiger Hinweis

Diese Software ist „as is“ und es wird keine Gewähr in irgendeiner Hinsicht geleistet. Zudem ist sie im Alpha – Zustand. Das bedeutet das alles noch experimentell ist.

Das Projekt wurde von mir gestartet, weil ich keine Software gefunden habe, mit deren Hilfe ich auf die Daten aus dem Glucose Messgerät unter Linux zugreifen konnte. Und auch diese Daten längerfristig verwalten kann.

Warum ausrechnet Lazarus und Pascal. Einfache Antwort, weil ich damit die letzten 25 Jahre mit Pascal gearbeitet habe. Warum Lazarus, weil es Open Source ist und auf allen wichtigen Plattformen vorhanden ist.

Übersicht

GlucoReader ist ein Tool zum grafischen Anzeigen der Daten vom Acensia Contour next ONE Geräten über USB. Aktuell funktioniert die Software nur unter Linux Betriebssystemen.

Die Daten lassen sich dann aufbereitet Anzeigen bzw. Drucken.

Version

Das Programm ist aktuell eine frühe Alpha Version. Das bedeutet das die grundlegenden Funktionen vorhanden sind, das Programm aber noch in den Kinderschuhen steckt.

Aktuell muß das Programm selbst aus den Quellen mithilfe von Lazarus und den FPC kompiliert werden. Es gibt keinen Installer.

Benötigte externe Softwarepakete müssen selbst besorgt werden.

Datensicherheit

Die Daten werden NUR lokal, das bedeutet auf ihren Rechner gespeichert. Das heißt auch das keine Datensicherung durchgeführt wird. Auf externe Geräte muss man die Daten selbst sichern.

Keine Daten in der Cloud, keine Daten verlassen mit diesem Programm ihren Rechner. Das Programm selbst hat keine Updatefunktion.

Der Quellcode besteht aus Pascal und ist jederzeit einsehbar.

Lizenz

Die Lizenz des Programmes selbst ist GPL V3.

Es werden externe Sourcecoden mit ihren jeweiligen Lizenzen im Programm verwendet.

Pakete:

- LazReport: in Lazarus enthalten
- Printer4Lazarus: in Lazarus enthalten

- DateTimeCtrls: in Lazarus enthalten
- TACHartLazarusPkg: in Lazarus enthalten
- FpcUSBHIDLinux: extern Projekt FPC-USB-HID¹ aber angepasstes Paket

Daten am Glucometer

Wenn die Daten aus dem Glucometer gelesen werden, wird dabei nichts am Glucometer verändert. Es sind ca. 800 Datensätze am Glucometer selbst gespeichert. Diese lassen sich jederzeit und wiederholt auslesen. Auch über die originalen Programme. Auch mehrere Glucometer lassen sich ohne Probleme auslesen. Es wird jedesmal geprüft, ob es den Datensatz – Datum, Zeit und Wert – bereits vorhanden ist, wenn vorhanden, dann wird der Wert nicht neuerlich in die Datenbank eingetragen.

¹ Quelle: Original <https://github.com/LongDirtyAnimAlf/FPC-USB-HID> hier verwendet <https://github.com/afriess/FPC-USB-HID> Branch andi_wip

Installation

Auf die Installation von FPC und Lazarus wird hier nicht eingegangen. Bitte die Wiki² oder das deutsche Forum³ zu Rate ziehen. Wie gesagt, das Programm ist aktuell Alpha.

Basis

Bei Lazarus müssen in der Paketverwaltung, zu finden unter Package → Install/Uninstall Packages folgende Pakete nachinstalliert werden, soweit nicht schon vorhanden.

- LazReport 1.0
- Printer4Lazarus 0.5
- DateTimeCtrls: 1.5.1
- TChartLazarusPkg 1.0

Aktuell ist das mit FPC 3.2.2 und Lazarus 4.x getestet. Beim FPC main gab es ein Problem mit dem Inline, sollte seit 11.2.2026 gefixt sein.

Mittels GIT

Installieren der Sourcen mittels GIT. Dazu bitte die Dokumenttaion von GIT zu Rate ziehen.

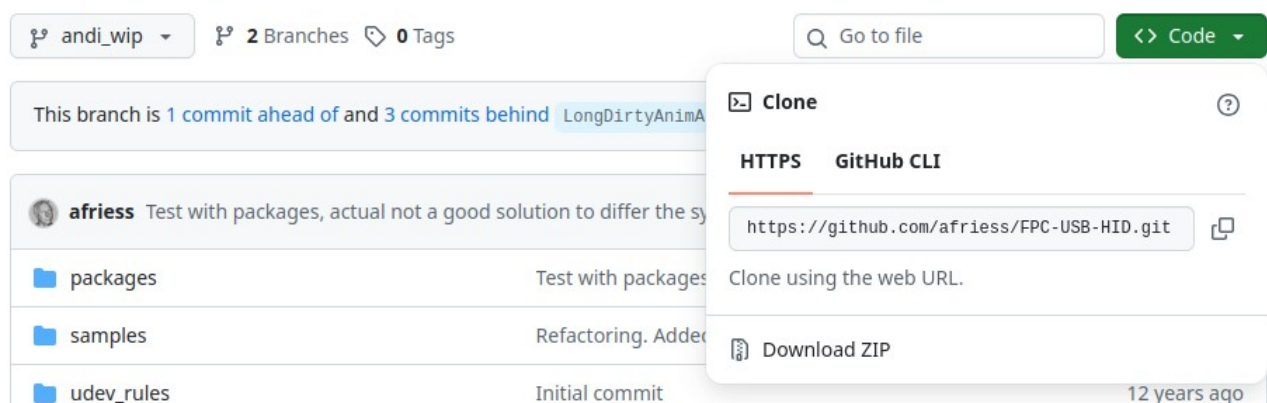
Quellen:

https://github.com/afriess/FPC-USB-HID/tree/andi_wip (wichtig der Zweig andi_wip !)

<https://codeberg.org/afriess/GlucoReader>

Aus den Zip's

FPC-USB-HID

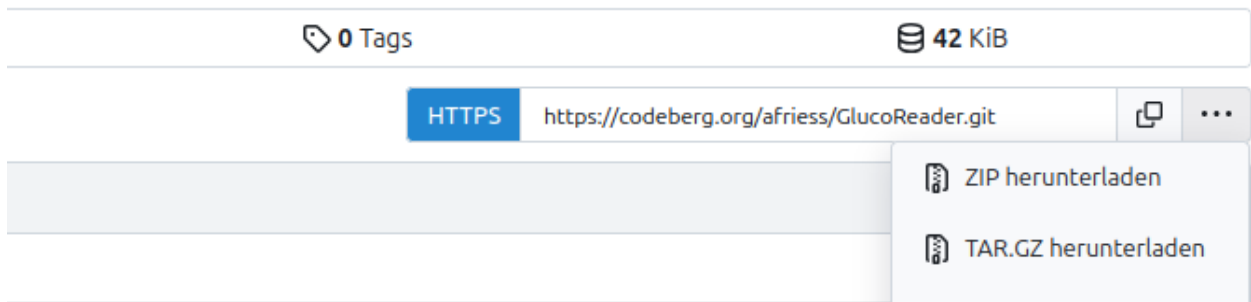


Oder einfacher bei <https://github.com/afriess/FPC-USB-HID.git> zuerst links Branch „andi_wip“ anwählen und dann rechts auf „Code“ gehen. Dort das „Download ZIP“ anwählen. Diese Zip-Datei dann in seinen Arbeitsbereich für Pascalcode entpacken.

² Zum Beispiel https://wiki.lazarus.freepascal.org/Installing_Lazarus/de

³ <https://www.lazarusforum.de/>

GlucoReader

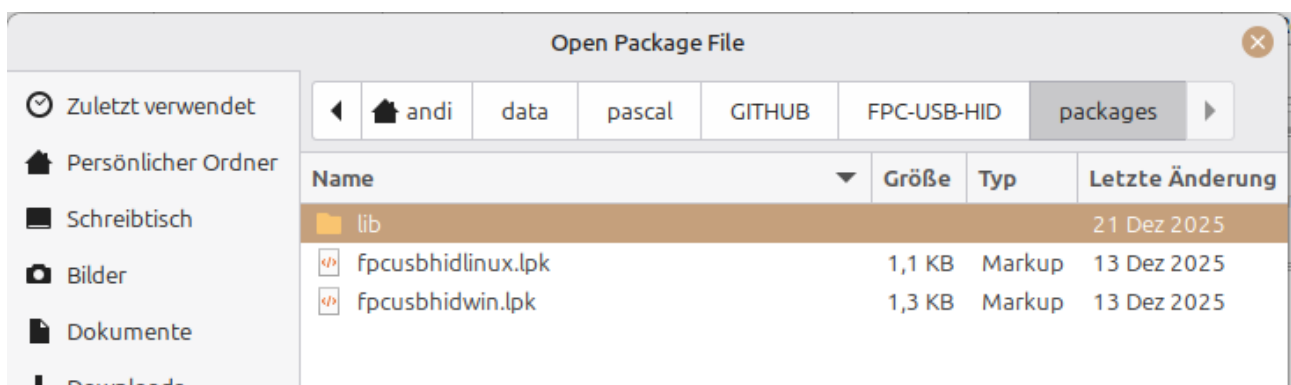


Beim <https://codeberg.org/afriess/GlucoReader> rechts auf die drei (...) Punkte gehen und auch das Zip herunter laden auswählen. Diese Zip-Datei dann ebenfalls in seinen Arbeitsbereich für Pascalcode entpacken. Bitte für den FPC-USB-HID und Glucoreader zwei verschiedene Verzeichnisse verwenden.

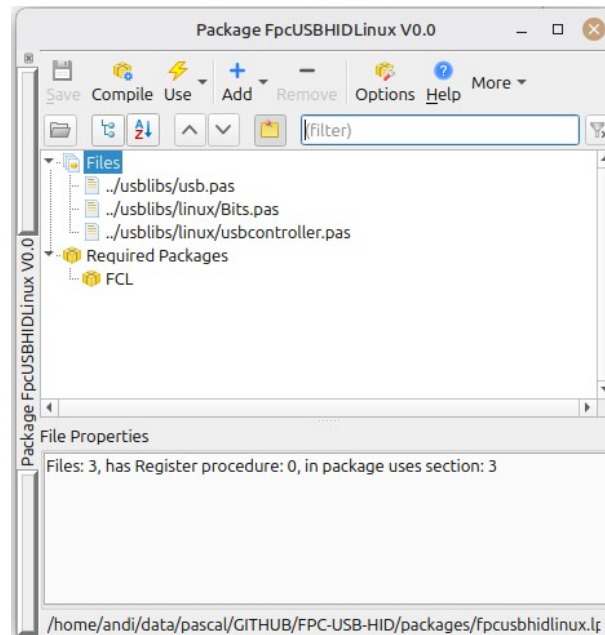
Installation in Lazarus

Es wird nur der FPC-USB-HID als Paket in Lazarus installiert. Es ist kein visuelles Paket, das bedeutet, Lazarus muß nicht neu kompiliert werden.

In Lazarus im Menü Package → Open Package File anwählen und dort fpcushhidlinux.lpk öffnen



Dort dann einfach auf „Compile“ drücken und das Paket wird kompiliert und ist somit auch Lazarus bekannt und kann verwendet werden.



Falls man eine neue Version installieren will, so ist der Vorgang einfach zu wiederholen.

Es braucht nicht in Lazarus installiert werden !

Projekt in Lazarus kompilieren

Dazu das Projekt in Lazarus öffnen und im Menü Projekt → Projekt Options dort in die Compiler Options gehen und oben den gewünschten Build Modus auswählen.

Build Mode:

- **Debug_Testdata:** Es wird mit Testdaten gearbeitet, deswegen ist bei der Anzeige der Daten der Bereich Daten holen rot hinterlegt. Es werden keine realen Messdaten zur Anzeige verwendet. Es wird ein spezieller Dateinamen der ausführbaren Datei verwendet.
- **Debug:** Alles funktioniert normal, es sind nur alle Debugfunktionen und Überprüfungen aktiviert. Bei einem Problem, bleibt das Programm mit einer Meldung in Lazarus stehen. Der Modus ist für die Suche nach Fehlern gedacht.
- **Release:** Alles funktioniert normal, es sind keine Debugfunktionen aktiv.

Wenn im Modus Release kompiliert wird, so kann man die ausführbare Datei auch direkt aus dem Verzeichnis aufrufen aber auch in ein beliebiges Verzeichnis kopiert werden.

Verwendete Verzeichnisse

Die ausführbare Datei ist im selben Verzeichnis wie die Projektdatei

Die Datenbank wird im Homeverzeichnis des Benutzers unter „<home>.local/share/<Programmname>/<Programmname>.sq3“ abgelegt.

Konfigurationen werden im Homeverzeichnis des Benutzers unter „<home>.config/<Programmname>/“ abgelegt. Derzeit noch nicht verwendet.

Für ein einfaches Backup, einfach die Daten in den Verzeichnissen sichern.

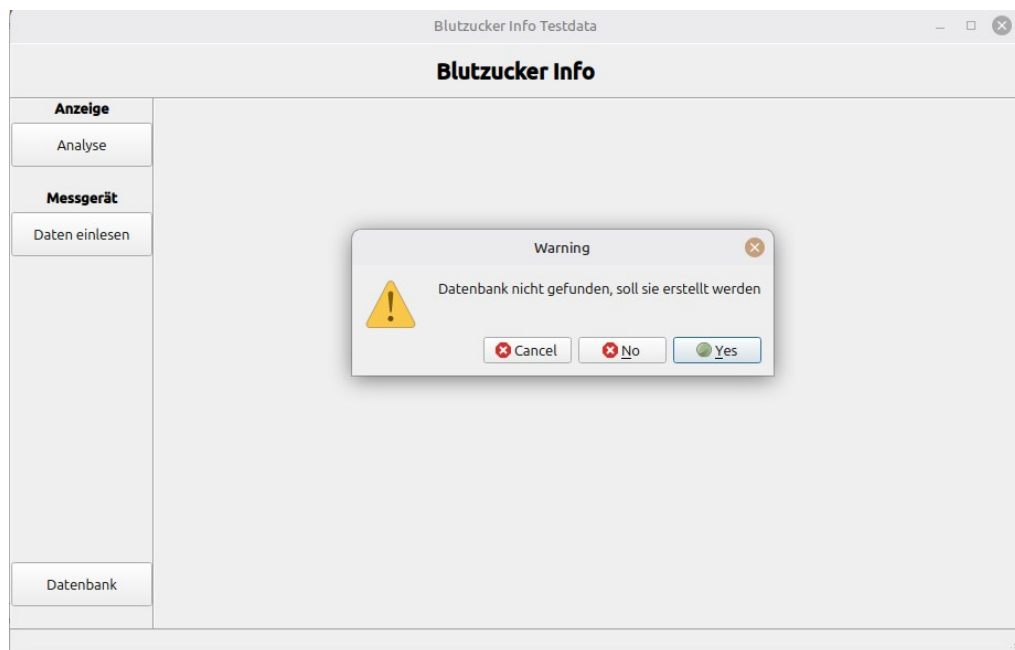
Erster Start

Achtung: Die Vorgangsweise kann sich in den nächsten Versionen ändern !

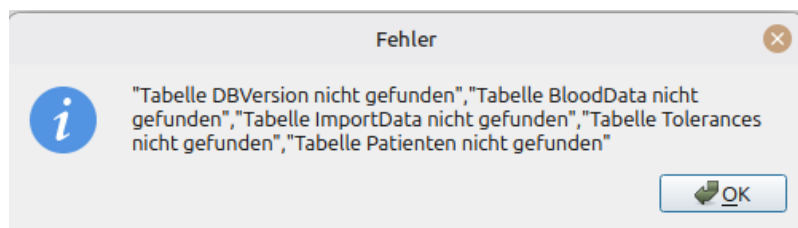
Bei mir sind die Meldungen teilweise auf englisch, da mein Lazarus komplett auf englisch eingestellt ist.

Erstellen der Datenbank

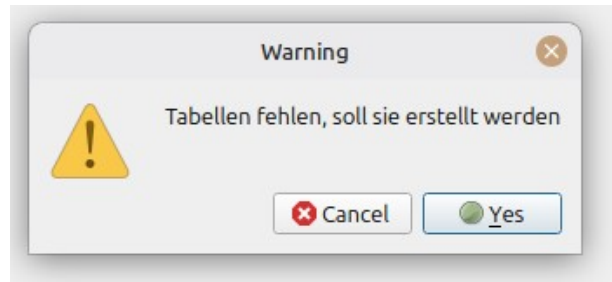
Es wird erkannt das die Datenbank (die SQLite Datei) nicht vorhanden ist und man wird gefragt ob diese erstellt werden soll → mit „Yes“ bestätigen



Die nächste Meldung ist ein Hinweis, das Tabellen noch nicht gefunden wurden, einfach mit „Ok“ bestätigen.



Die nächste Meldung ist die Frage ob man die fehlenden Tabellen erstellen will – mit „Yes“



bestätigen.

Es öffnet sich anschließend der Dialog für die Verwaltung der Patienten. Anhand der Anzeige „Anz“ links, sieht man das die Patiendaten leer sind. Es muß für die korrekte Funktion zumindest ein Patient angelegt sein.

frmPatient			
Patient bearbeiten			
Bearbeiten	Neuer Eintrag	Eintrag Löschen	Anz=0

Wir wählen für das Anlegen eines Patienten einmal „Neuer Eintrag“ aus.

frmPatientEdit			
ID	0	SV Nr	1234012345
☒ Default Benutzer			
Namen	Max Mustermann		
Diabetestyp	2 - Insulinresistenz	Therapie	BOT - Basalinsulin unterstützte OAD
Persönliche Grenzen (mg/dl)			
Untergrenze Blutzucker	75	Obergrenze Blutzucker	135
Speichern		Verwerfen	

Grundlegend zu den Daten, diese werden für den Ausdruck verwendet. Legt man die Ausdrücke dann beim Arzt vor, so sollten die Daten korrekt sein. Ansonsten kann man ausfüllen was man will.

- ID wird NICHT ausgefüllt.
- SV Nr wird die Sozialversicherungsnummer eingetragen
- Default Benutzer WICHTIG !! Unbedingt beim ersten Benutzer anwählen
- Namen
- Diabetes Typ meisten 1A oder 2 (Siehe Diagnose vom Arzt)
- Therapie welche Therapieart angewendet wird (meisten OAD oder BOT, siehe Therapievorslag vom Arzt)

- Persönliche Grenzen – Grenzen die in den Grafiken verwendet werden um den „normalen“ Bereich der Therapie anzuzeigen. Meist vom Arzt definiert. Sollten mit den Grenzen die man im Glucometer definiert hat zusammenstimmen.

Anschließend mit „Speicher“ übernehmen. Jetzt ist der erste (meist der einzige) Patient definiert.

Patient bearbeiten							
Bearbeiten		Neuer Eintrag		Eintrag Löschen		Anz=1	
ID	Name	SVNr	Typ	Therapie	OG	UG	default
1	Max Mustermann		2	BOT	135	75	*

Das Fenster wird durch den „X“ schließen Knopf am rechten Fensterrand (kann ja nach Betriebssystem und Fenstermanager etwas anders aussehen) geschlossen.

Damit ist die Ersteinrichtung abgeschlossen.

Mit Glucoreader arbeiten

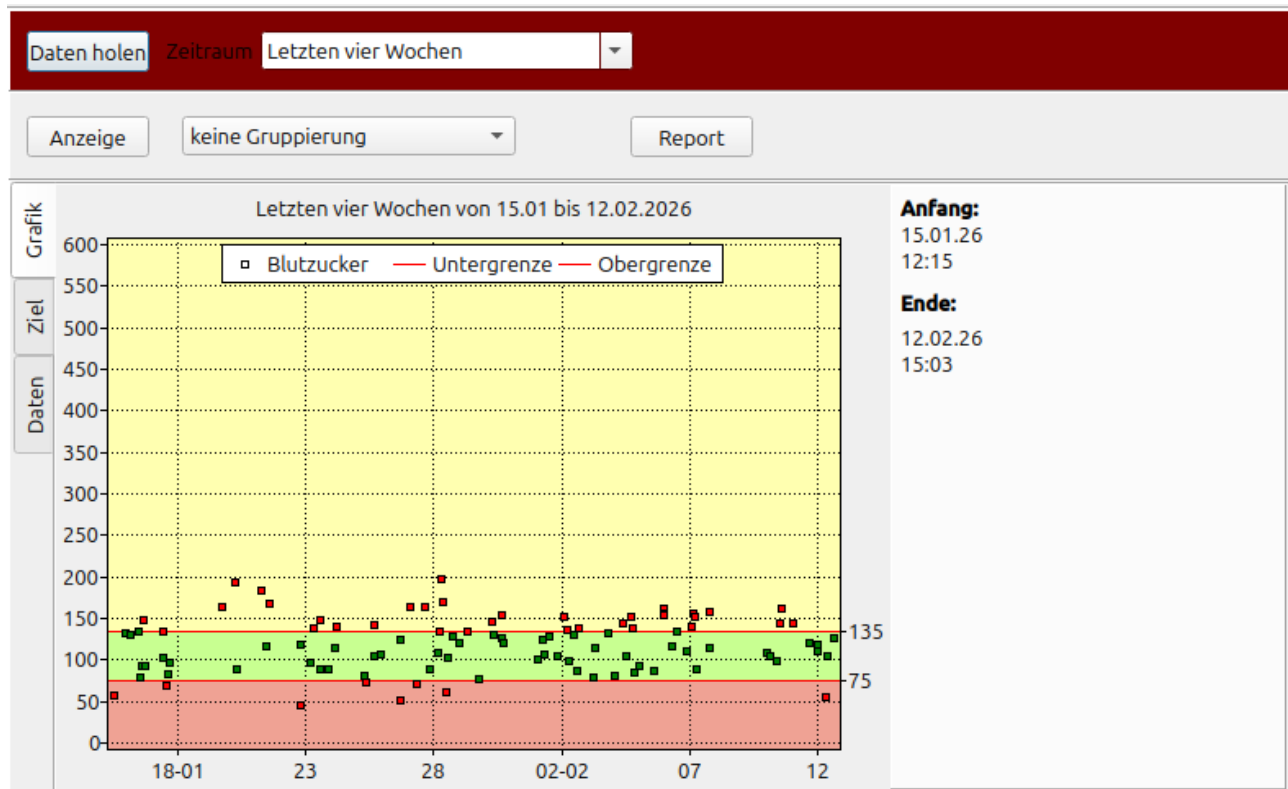
Es gibt in dem Programm aktuell 3 Bereiche

- Anzeige mit Analyse
- Messgerät mit Daten einlesen
- Datenbank mit verschiedenen Funktion

Die ersten beiden Punkte werden in der normalen Verwendung benötigt. Im Bereich Datenbank sind Sonderfunktion wie der Import aus anderen System, die Patientenverwaltung und weitere Funktionen, die eher in Sonderfällen benötigt werden, untergebracht.

Anzeige – Analyse

Hier befinden sich die grafische Aufbereitung der Daten und das Drucken dieser Analysen.



Hinweis: Das obere Panel ist im normalen Betrieb NICHT rot⁴.

Bedienelemente:

- Daten holen: Laden der Daten aus der Datenbank nach dem angewählten Zeitraum
- Zeitraum: Vordefinierte Zeiträume aus welcher die Daten aus der Datenbank gelesen werden
- Anzeig: Auffrischen der Anzeige falls es Darstellungprobleme gibt (normalerweise nicht mehr verwendet)
- Gruppierung: Standard ist keine Gruppierung, sondern die Daten werden laut Datum aufsteigend angezeigt. Weiters kann man die Daten nach Stunden des Tages oder Wochentagen gruppieren.
- Report: Erstellt den Report anhand der Einstellungen und zeigt diesen an. In der Vorschau kann dann der Druck ausgelöst werden.
- Grafik: Anzeige des Verlaufs entsprechend der Gruppierung
- Ziel: Tortengrafik, anhand der man sieht, wie gut man in der Therapie gelegen ist
- Daten: Liste der Rohdaten, die für die Grafik herangezogen worden sind.

⁴ Rot bedeutet, das die Daten nicht aus Messwerten stammen. Ist das Feld rot, dann ist der falsche „Build Modus“ beim kompilieren verwendet worden.

Messgerät – Daten einlesen

Lesen der Daten aus dem Messgerät „Contour next One“.

The screenshot shows a software window with a light gray background. At the top, there are five buttons: 'USB Activate', 'USB Deactivate', 'Start', 'Stop', and 'Store'. Below these buttons, there are several data fields: 'Zeile: 0', 'Wert: 0', 'mg/dl', 'Datum: 0000.00.00', 'Zeit: 00:00:00', and 'Device: -----'. To the right of these fields is a checkbox labeled 'Anzeigen'. Below the data fields is a large text area labeled 'Log'.

Bedienelemente:

- USB Active: Schaltet die Erkennung des Messgerätes am USB ein
- USB Deactivate: Schaltet die Erkennung wieder aus
- Start: Startet das Einlesen der Daten, kann länger dauern
- Stop: Abbruch des Einlesens (wird normal nicht verwendet)
- Store: Speichern der Daten in der Datenbank
- Anzeigen: Zeigt die aktuellen Daten an (Verlangsamt das Einlesen deutlich)
- Zeile: Aktuelle Zeile die eingelesen wird
- Wert der gelesene Wert
- Datum und Zeit
- Device: Messgerät was erkannt wurde

Vorgang:

Das Messgerät wird mittels USB-Kabel mit dem Rechner verbunden, dann drückt man „USB-Activate“ und wartet bis der „Start“ Button aktiv wird und ein Messgerät erkannt wird.

Dann kann man, wenn man will, das „Anzeigen“ aktivieren und auf „Start“ drücken.

Das löst das Einlesen der Daten aus. Das kann eine Zeitlang dauern, das das aktuelle Messgerät bis zu 800 Datensätze einliest.

Nach dem Einlesen sind die Daten NOCH NICHT in der Datenbank gespeichert. Dieser Vorgang wird erst durch das Drücken von „Store“ ausgelöst. Dabei wird auch kontrolliert ob die Daten bereits in der Datenbank vorhanden sind. Vorhandene Daten werden nicht überschrieben.

Datenbank

Patientendatenverwaltung

Die Patientendatenverwaltung haben wir schon beim Einrichten nach dem ersten Start gesehen.

Patient bearbeiten							
Bearbeiten		Neuer Eintrag		Eintrag Löschen		Anz=1	
ID	Name	SVNr	Typ	Therapie	OG	UG	default
1	Max Mustermann		2	BOT	135	75	*

Bedienelemente:

- Bearbeiten: Bearbeiten des aktuell ausgewählten Benutzers
- Neu Eintrag: neuen Benutzer erstellen
- Eintrag löschen: Der aktuell ausgewählte Benutzer wird gelöscht. Es muss mindestens einen Benutzer geben.

Meistens gibt es nur einen Benutzer, falls man möchte kann man mehrere Benutzer anlegen. Die Messdaten werden dann getrennt für die Benutzer behandelt. Sind aber weiterhin in einer Datenbank gespeichert.

Wicht ist der „default“ Benutzer. Alle Aktionen werden auf diesen Benutzer bezogen. Auswählen kann man diesen nur hier in der Patienvverwaltung. Dazu muss der Datensatz bearbeitet werden und das Häkchen beim richtigen Benutzer gemacht werden. Es kann NUR EINEN default benutzer geben. Wird der geändert, werden automatisch die Berechtigungen bei anderen Benutzern zurückgezogen. Der default Benutzer ist beim Programmstart auch automatisch aktiv.

frmPatientEdit

ID0SV Nr1234012345☒ Default Benutzer

NamenMax Mustermann

Diabetestyp2 - InsulinresistenzTherapieBOT - Basalinsulin unterstützte OAD

Persönliche Grenzen (mg/dl)

Untergrenze Blutzucker75Obergrenze Blutzucker135

SpeichernVerwerfen

Grundlegend zu den Daten, diese werden für den Ausdruck verwendet. Legt man die Ausdrücke dann beim Arzt vor, so sollten die Daten korrekt sein. Ansonsten kann man ausfüllen was man will.

- ID wird NICHT ausgefüllt.

- SV Nr wird die Sozialversicherungsnummer eingetragen
- Default Benutzer WICHTIG Ist das Häkchen gesetzt wird es bei anderen Benutzern automatisch gelöscht
- Namen
- Diabetes Typ meisten 1A oder 2 (Siehe Diagnose vom Arzt)
- Therapie welche Therapieart angewendet wird (meisten OAD oder BOT, siehe Therapievorschlag vom Arzt)
- Persönliche Grenzen – Grenzen die in den Grafiken verwendet werden um den „normalen“ Bereich der Therapie anzuzeigen. Meist vom Arzt definiert. Sollten mit den Grenzen die man im Glucometer definiert hat zusammenstimmen.

Anschließend mit „Speicher“ übernehmen.

Import Bayer

Mit dieser Funktion kann man Daten aus dem Windows Programm Glucofacts-Deluxe übernehmen. Dazu wird die Datenbankdatei von Glucofacts benötigt.

The screenshot shows the 'frmBayer' application window. It features a title bar with the text 'frmBayer' and a close button. Below the title bar, there are four buttons: 'File Select', 'Start', 'Stop', and 'Store'. Below these buttons, there is a status bar with the following labels: 'Zeile: 0', 'Wert: 0', 'mg/dl', 'Datum: 0000.00.00', 'Zeit: 00:00:00', and 'Comment'. There is also a checkbox labeled 'Anzeigen'. Below the status bar, there is a large text area labeled 'Log'. At the bottom of the window, there is a label 'Database'.

Man braucht die Datenbankdatei von Glucofacts nur auf den Linux Rechner kopieren, egal welches Verzeichnis. Sinnvoll ist es die Datei im Home Verzeichnis abzulegen.

Bedienelemente:

- File Select: Damit kann man die Datenbankdatei von Glucofacts auswählen.
- Start: Startet den Leseprozess der Daten, kann länger dauern, da hier tausende an Daten gespeichert sein können.
- Stop: Damit kann man, wenn es nötig sein sollte den Leseprozeß stoppen
- Store: Speichert die Daten in der Datenbank
- Database: Hier wird die aktuelle Datenbank aus der gelesen wird angezeigt

Es können kleinere Fehler auftreten, die meistens von doppelten Datensätzen in der Glucofacts Datenbank herrühren, bzw. lassen sich manche Daten nicht richtig konvertieren.

Die Glucofacts Datenbank ist ebenso eine SQLite3 Datenbank, somit kann man den Inhalt auch notfalls mit eine SQLite Tool inspizieren.

Mir liegt nur eine Version der Importdatenbank vor, somit kann es nicht ausgeschlossen sein, das sich das interne Format geändert hat und das Einlesen deswegen fehl schlägt.

Import CSV

Nicht implementiert

Export CSV

Nicht implementiert