

mHealth a nositelná elektronika

TEORIE A PRAXE

Martin Úbl

2021

A solid orange horizontal bar at the bottom of the page.

Přibližný obsah

1. Představení, kontext, léčba diabetu
2. DIY vs. certifikovaná zařízení
3. mHealth a požadavky na zařízení
4. Nositelná elektronika a sběr dat
5. SmartCGMS

Představení a kontext

Diabetes

- Typ 1, 2 a ostatní

Zaměření na typ 1

- Autoimunitní
- Projeví se často již v dětství
- Léčba inzulinem
- V současnosti vyžaduje mnoho nositelné elektroniky

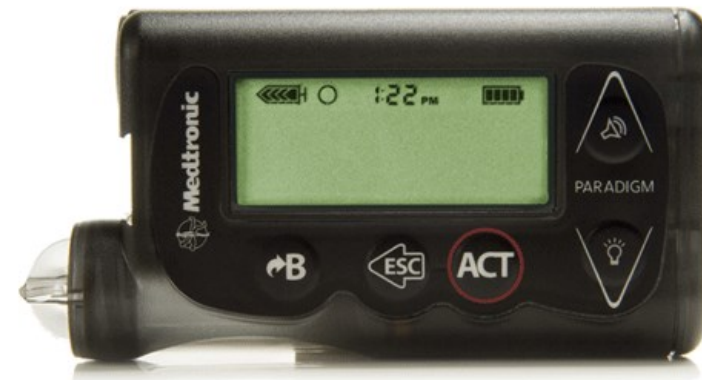
<https://diabetes.zcu.cz/>

Léčba diabetu

Inzulín (typ 1, nově i typ 2)

- Inzulínové pero
- Inzulínová pumpa
 - Podkožní
 - Intradermální

Antidiabetika (typ 2)



Dávkování inzulínu

Bolusové

- Ruční

Bazální

- Ruční
- Automatické
 - Jak?

Měření

Koncentrace glukózy

- V krvi
 - Glukometr
 - Sporadicky
- V podkoží
 - CGM senzor
 - „kontinuálně“



Typický „setup“

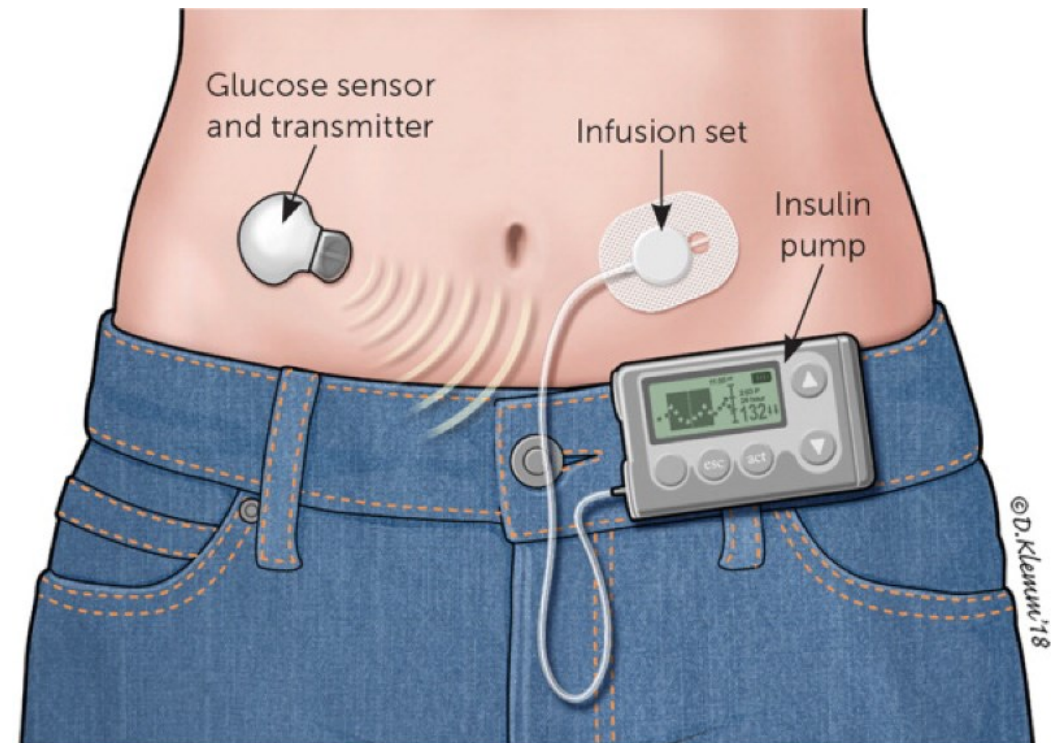
Senzor

Pumpa

Infúzní set

Řídicí zařízení

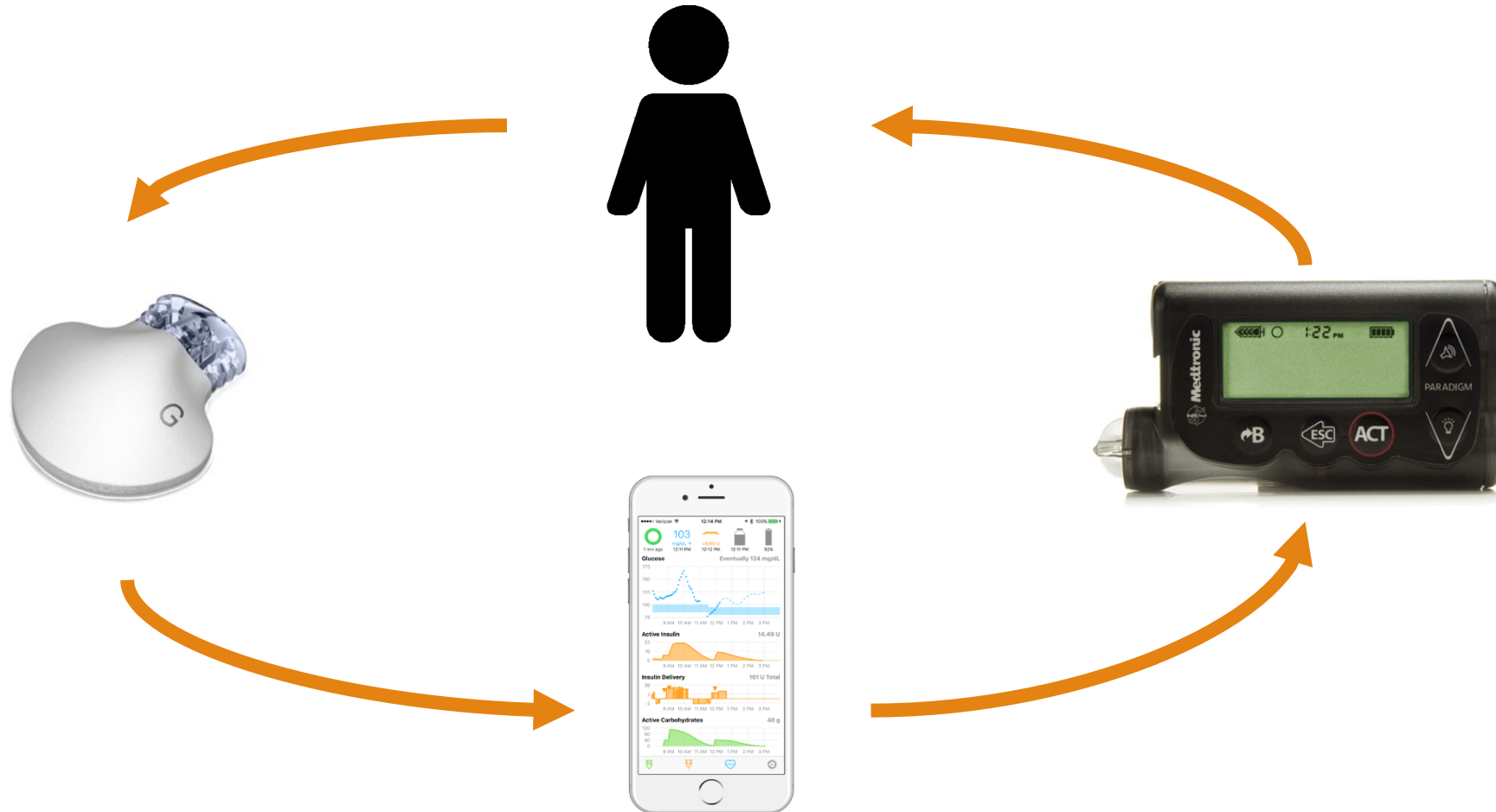
- Nepovinné
- Mezi senzorem a pumpou
- Mobilní telefon, hodinky



Diskuse: spousty zařízení na člověku – jak to ovlivní psychiku, např. u dětských pacientů?

Zdroj: <https://diabeteson.com/technical-devices-that-improve-risk-factors-care-and-quality-of-life/>

Uzavřená smyčka → umělá slinivka



Vývoj regulátorů

Certifikovaný

- „ten správný“

DIY

- „ten rychlý“

DIY

Trend posledních cca 15 let

Pacienti si sami vyvinou léčebný aparát

- Lepení komponent
- Prototypování algoritmů

Nepodléhá certifikaci

Rizika vs. výhody?

DIY v léčbě diabetu

OpenAPS

- open-source, **JavaScript**, **Python**
- Algoritmus oref0

AndroidAPS

- open-source, **Java**
- Pro regulaci spouští oref0 (**JavaScript**)

Loop

- iOS verze, **Objective-C** (později Swift)

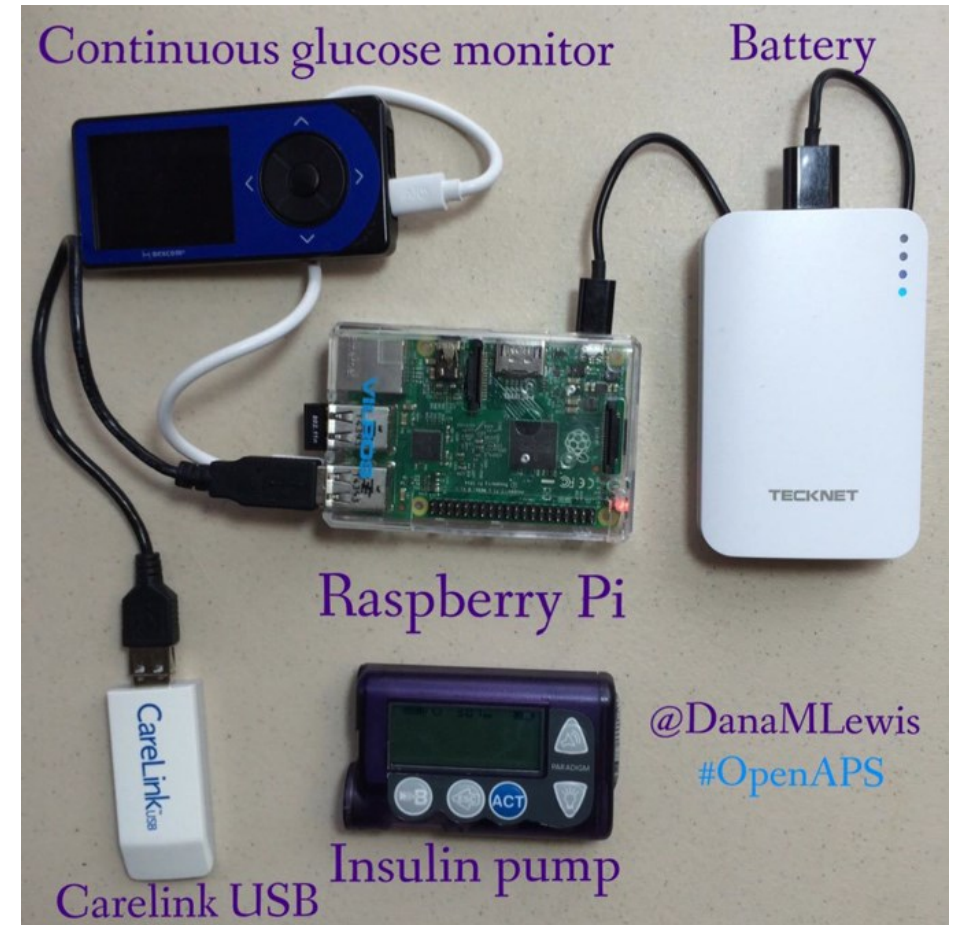
OpenAPS

mHealth?

Nositelná zařízení?

Bezpečnost?

- Zastaralá zařízení
- JavaScript
- Co když dojde k selhání?



OpenAPS - selhání

Jak řešit selhání na některém ze zařízení?

V OpenAPS to úplně ideálně nevypadá...

```
// 38 is an xDrip error state that usually indicates sensor failure
// all other BG values between 11 and 37 mg/dL reflect non-error-code BG values, so we should zero temp for those
if (bg <= 10 || bg === 38 || noise >= 3) { //Dexcom is in ??? mode or calibrating, or xDrip reports high noise
  rT.reason = "CGM is calibrating, in ??? state, or noise is high";
}
if (minAgo > 12 || minAgo < -5) { // Dexcom data is too old, or way in the future
  rT.reason = "If current system time "+systemTime+" is correct, then BG data is too old. The last BG data was read "+minAgo+"m ago at "+bgTime;
// if BG is too old/noisy, or is changing less than 1 mg/dL/5m for 45m, cancel any high temps and shorten any long zero temps
} else if ( bg > 60 && glucose_status == 0 && glucose_status.short_avgdelta > -1 && glucose_status.short_avgdelta < 1 && glucose_status.long_avgdelta > -1 &
  if ( glucose_status.last_cal && glucose_status.last_cal < 3 ) {
    rT.reason = "CGM was just calibrated";
  } else {
    rT.reason = "Error: CGM data is unchanged for the past ~45m";
  }
}
if (bg <= 10 || bg === 38 || noise >= 3 || minAgo > 12 || minAgo < -5 || ( bg > 60 && glucose_status == 0 && glucose_status.short_avgdelta > -1 && glucose_;
```

Logování do konzole

Bez pokusu o zotavení

```
try {
  iobArray.forEach(function(iobTick) {
    ...80 řádek kódu...
  } catch (e) {
    console.error("Problem with iobArray. Optional feature Advanced Meal Assist disabled");
  }
}
```

OpenAPS – neseleže?

„OpenAPS nemůže selhat“

- Fakt?
- Konstatováno na základě „desetitisíců hodin běhu“

Selhání lze částečně předejít formální verifikací

- Tou kód OpenAPS jistě neprošel
- Verifikovatelnosti musí jít kód trochu naproti
 - „spaghetti“ kód o 1600 řádcích v JavaScriptu to nedělá

```
rT.predBGs = {};  
IOBpredBGs.forEach(function(p, i, theArray) {  
    theArray[i] = round(Math.min(401,Math.max(39,p)));  
});  
for (var i=IOBpredBGs.length-1; i > 12; i--) {  
    if (IOBpredBGs[i-1] !== IOBpredBGs[i]) { break; }  
    else { IOBpredBGs.pop(); }  
}  
rT.predBGs.IOB = IOBpredBGs;  
lastIOBpredBG=round(IOBpredBGs[IOBpredBGs.length-1]);  
ZTpredBGs.forEach(function(p, i, theArray) {  
    theArray[i] = round(Math.min(401,Math.max(39,p)));  
});  
for (i=ZTpredBGs.length-1; i > 6; i--) {  
    // stop displaying ZTpredBGs once they're rising and above target  
    if (ZTpredBGs[i-1] >= ZTpredBGs[i] || ZTpredBGs[i] <= target_bg) { break; }  
    else { ZTpredBGs.pop(); }  
}  
rT.predBGs.ZT = ZTpredBGs;  
lastZTpredBG=round(ZTpredBGs[ZTpredBGs.length-1]);  
if (meal_data.mealCOB > 0) {  
    aCOBpredBGs.forEach(function(p, i, theArray) {  
        theArray[i] = round(Math.min(401,Math.max(39,p)));  
    });  
    for (i=aCOBpredBGs.length-1; i > 12; i--) {  
        if (aCOBpredBGs[i-1] !== aCOBpredBGs[i]) { break; }  
        else { aCOBpredBGs.pop(); }  
    }  
}  
if (meal_data.mealCOB > 0 && ( ci > 0 || remainingCIpeak > 0 )) {  
    COBpredBGs.forEach(function(p, i, theArray) {  
        theArray[i] = round(Math.min(401,Math.max(39,p)));  
    });  
    for (i=COBpredBGs.length-1; i > 12; i--) {
```

AndroidAPS

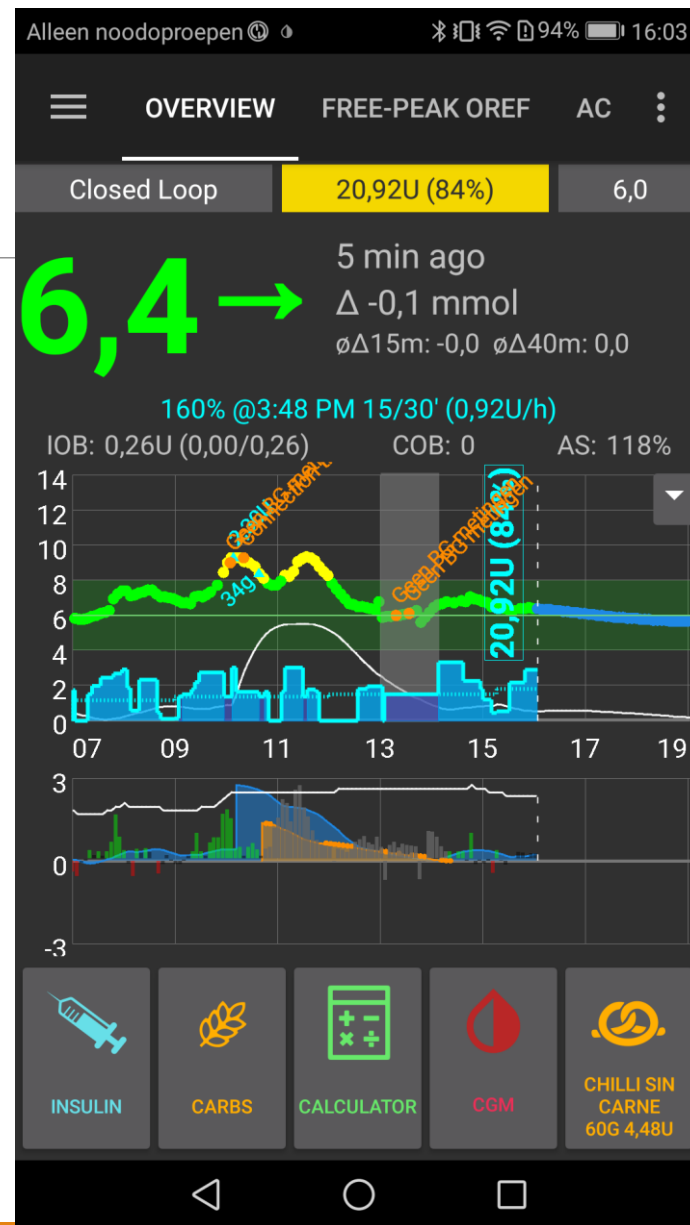
Podobná situace

- **Java** aplikace pro **Android**
- Spouští **JavaScript** kód pro výpočet inzulínu

Autor svému software věří natolik, že ho v režimu uzavřené smyčky použil na svou vlastní dceru (10 let)

Kód na tom není o moc lépe

Autor sám přiznává, že „není moc programátor“



Všechny systémy

Nositelná elektronika

- Vyžaduje komunikační protokol

DIY systémy využívají zastaralý hardware

- S již prolomeným protokolem
- Děravé
- Dávno bez záruky
- Kdo nese odpovědnost za újmu na zdraví?
 - Pacient si naordinuje sám
 - Diabetolog toleruje, občas i podporuje

DIY systémy selhávají

A ne zrovna ojedinele

- Výběr z posledních pár aktivních issue OpenAPS



tzachi-dar commented on 2 Dec 2018

Describe the bug

It seems that today, too much insulin was dosed in the morning, (at 07:55) for example.

To Reproduce

Not possible, but I have the data needed I think.

Expected behavior

A much smaller basal rate

Screenshots

Will be attached. As one can see:

at 07:55 BG was 82, (and going up in jumps of 9).

target bg is 110.

basal = 0.4

sensitivity factor = 100.

So, very naive, since we are below target no basal should be issued.

Protichůdná pravidla pro regulaci

Dávkováno moc inzulínu kvůli nedetekovanému selhání senzoru



scottleibrand commented on 18 May 2019

Contributor

We received an Adverse Event report today (via private message to @AdrianLxM) of a user running OpenAPS oref0 version 0.6.2, with CGM data from Libre/LimiTer/xDrip+, who received excess insulin due to failure to detect a failing sensor. The event in question did not result in hypoglycemia or any other negative health outcome, but could have done so if they had not noticed the problem and removed the failing sensor.

Nedostatečné testování (CI/CD)



stevebell117 commented on 12 Mar 2019 • edited

Contributor

Describe the bug

Installing the newest oref0, the ns-status that's being uploaded to NS is breaking the "devicestatus" collection due to including a "uploader: {}" empty hash on a non-uploader entry.

Chyby v kódu kvůli jeho nepřehlednosti



guimkwon commented on 1 Jan

Describe the bug

The communication between rig and pump fails when BG level goes below 68 mg/dl. I see BG levels below 68 mg/dl on the pump screen fine. However, no BG reported on nightscout and putty shows 'BG too old' error. When BG goes up above 70 mg/dl, everything goes back to normal; looping works and BG reported on nightscout. Please see nightscout screen shot below.



dcacklam commented on 14 Apr

Contributor

Describe the bug

When OpenAPS attempts to set a basal of 6u/hr or higher, the actual command sent to the pump sets the basal to 0.

Tiché selhání samotného OpenAPS

Podle čeho je DIY „bezpečné“?

As users of a patient-driven technology, OpenAPS users are self-reporting improved A1C, day-to-day glucose levels, and quality of life. Safety features important to individuals with diabetes are perceived to be embedded into OpenAPS technology. Twitter analysis provides insight on a patient population driving an innovative solution to improve their quality of diabetes care.

<https://doi.org/10.1177/1932296818795705>

hours. In this highly selective population, user self-reporting suggests OpenAPS is much safer than standard pump with CGM therapy, measured by time spent in hypo- and hyperglycaemia, with no self-reports of severe hypo- or hyperglycaemic events [34](#).

<https://doi.org/10.1111/dme.13816>

OpenAPS is designed to be, and has been, far safer than standard pump/CGM therapy, as measured by duration of hypoglycemia and hyperglycemia, with no reports of severe hypo or hyperglycemic

<https://openaps.org/2016/06/11/real-world-use-of-open-source-artificial-pancreas-systems-poster-presented-at-american-diabetes-association-scientific-sessions/>
<https://dx.doi.org/10.1177%2F1932296816665635>

Discussion and Conclusions: Closing the loop with OpenAPS in people with T1D is effective in decreasing A1c and %T1Hypo, without any serious adverse event. Of note, these results were obtained with people who showed a good baseline metabolic control (A1c of 7.17%). However, we need to study OpenAPS implementation on a larger sample of people with T1D and with a

<https://doi.org/10.2337/db18-993-P>

DIYPS¹⁶ and the #OpenAPS project.¹⁷ The dangers posed to patients from the do-it-yourself artificial pancreas may not be from individuals with malice, but rather from users with an excess of enthusiasm and a shortage of knowledge and experience.

<https://doi.org/10.1177/1932296815583334>

Lékaři DIY vítají

Velká část lékařů vidí jen výsledky

- Ty jsou povětšinou dobré

Psychologický aspekt?

„účel světí prostředky“, cílená ignorace technických nedostatků

Požadavky na mHealth zařízení

Správnost algoritmů

- Verifikace
- Důkladné testování ve správně zvolených scénářích
 - in-silico (preklinické)
 - in-vivo (klinické)

Odolnost proti poruchám

- „fault-tolerance“
- Verifikace

Zabezpečení

Životní cyklus

- záruka, aktualizace, pravidelné kontroly odborníkem, ...

Certifikace

FDA (USA), EMA (Evropa)

Obrovský proces

- Dlouhý
 - Klidně několik let
- Drahý
 - Klidně několik desítek miliónů \$
- Namáhavý

Certifikovaná zařízení mají zaručenou kvalitu

Verifikace algoritmů/zařízení

Systematické testování všech možných stavů systému s následnou validací odpovědi

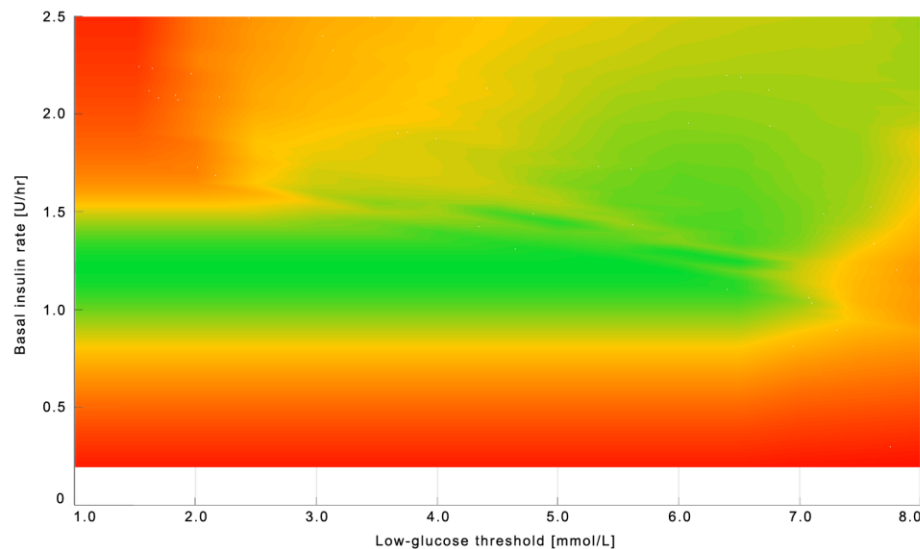
Jednoduchý příklad:

- Dvouparametrický regulátor
- Kartézský součin vzorkovaných intervalů
- Vyhodnocení metriky na sadě scénářů
- Pokus o identifikaci „faulty“ kombinací

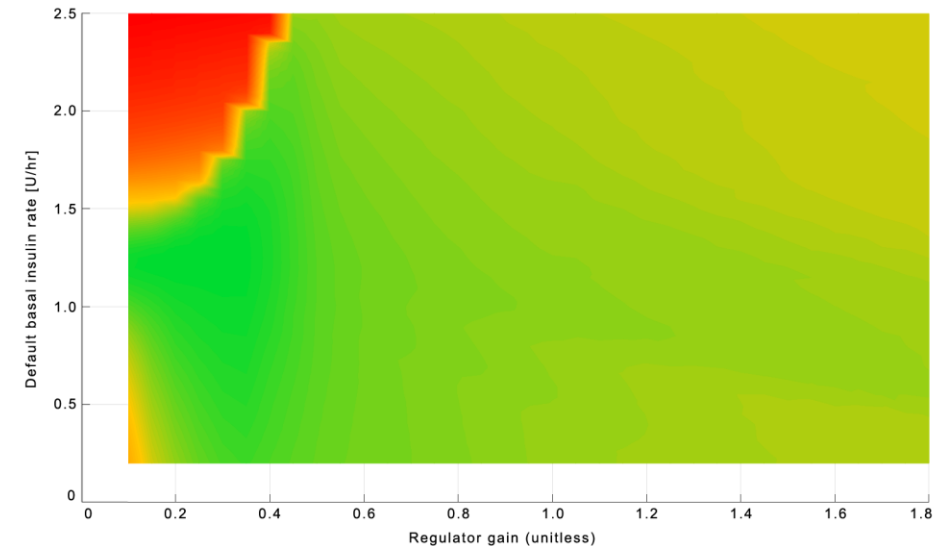
Konkrétní příklad

- 3 regulátory, všechny mají na vstupu 2 parametry
- Jen u 2 lze vymezit fyziologicky přijatelné regiony
- Legenda:
 - Červená – pravděpodobně smrtelné
 - Žlutá – na hraně, potenciálně nebezpečné
 - Zelená – nejlépe jak regulátor umí

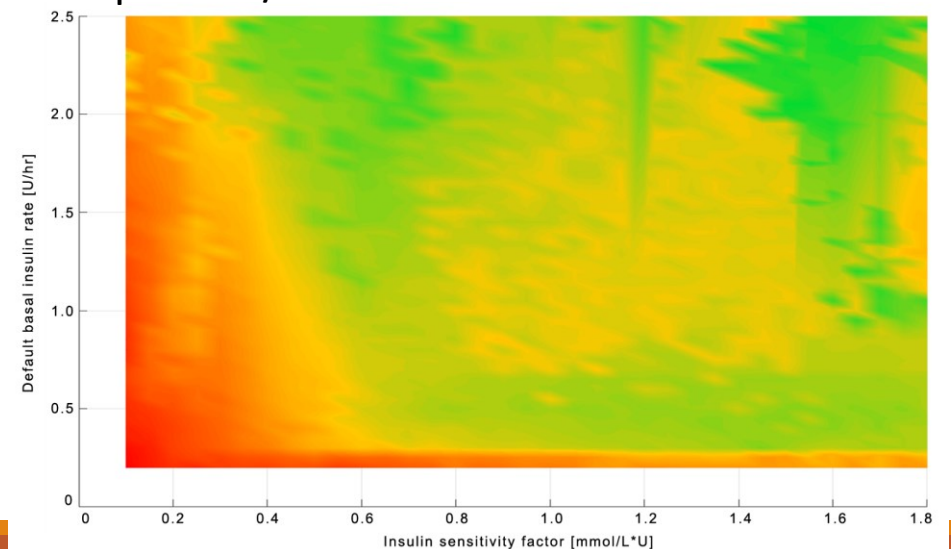
LGS – standard léčby, má certifikaci



BetaPID – adaptivní PID regulátor



OpenAPS/oref1 – DIY řešení



Nositelná zařízení

Mobil, hodinky, fitness náramky, pásy, ale i CGM senzory a jiné

Spousty senzorů

- Spousty dat
- Spousty možností



Akcelerometr

Magnetometr

Senzor světla

GPS

Senzor srdečního tepu

Senzor elektrodermální aktivity

Senzor krevního tlaku

Senzor míry okysličení krve

...

Nositelná zařízení - data

Personalizovaná medicína?

- Personalizace léčebných modelů

Telemedicína?

- Lékař má vždy k dispozici aktuální data
- Rodič má k dispozici data svého dítěte

Vývoj nových modelů?

- Datová množina pro prvotní cross-validaci

Nositelná zařízení - data

Problémy?

SmartCGMS

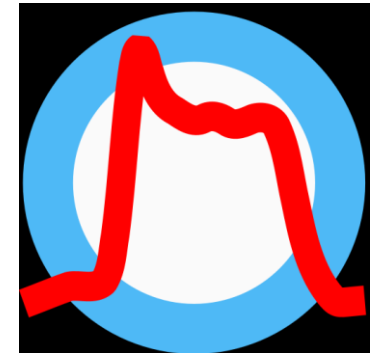
System vyvíjený zde na katedře

Framework pro analýzu signálů

Navržený tak, aby dosahoval produkčních kvalit

- Odolnost proti chybám
- Verifikovatelnost
- Jednoduchost
- Stabilita
- Přenositelnost
- Efektivita, úspornost

Podporuje jak simulace, tak „ostrý“ provoz



SmartCGMS

Implementace rozdělena do modulů různých typů

- Filtr
- Model
- Signál
- Solver
- Metrika
- ...

Každý modul lze verifikovat samostatně

- Výrazné ulehčení v procesu případné certifikace

Nový modul = verifikace modulu

- Není nutné znovu verifikovat celý systém

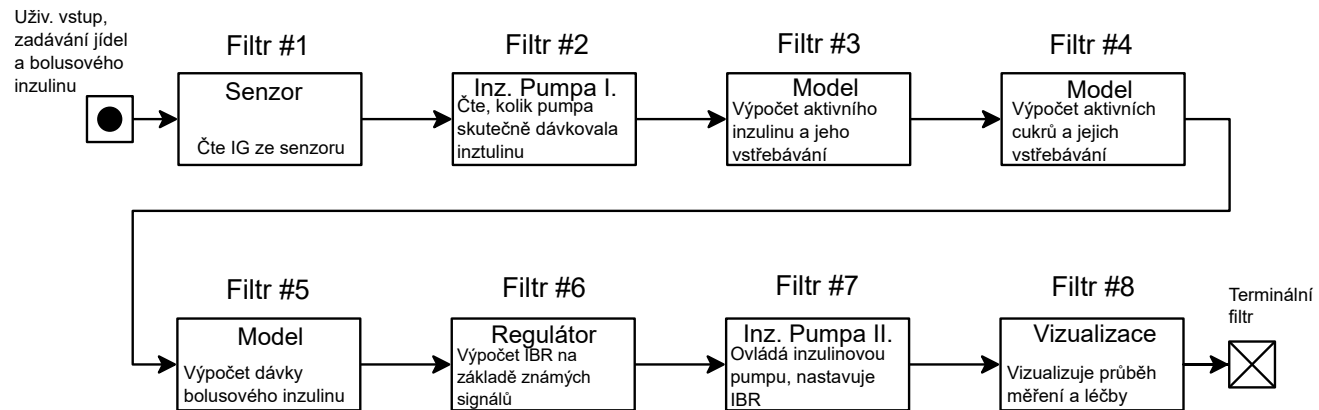
Snadný přechod od simulace do ostrého provozu

- Výměna modulu

SmartCGMS

Filtry lineárně propojené za sebou

Princip předávání zpráv („zleva doprava“)



```
struct TDevice_Event {
    NDevice_Event_Code event_code;
    GUID device_id;
    GUID signal_id;
    double device_time;
    int64_t logical_time;
    uint64_t segment_id;
    union {
        double level;
        IModel_Parameter_Vector* parameters;
        wstr_container* information;
    };
};
```

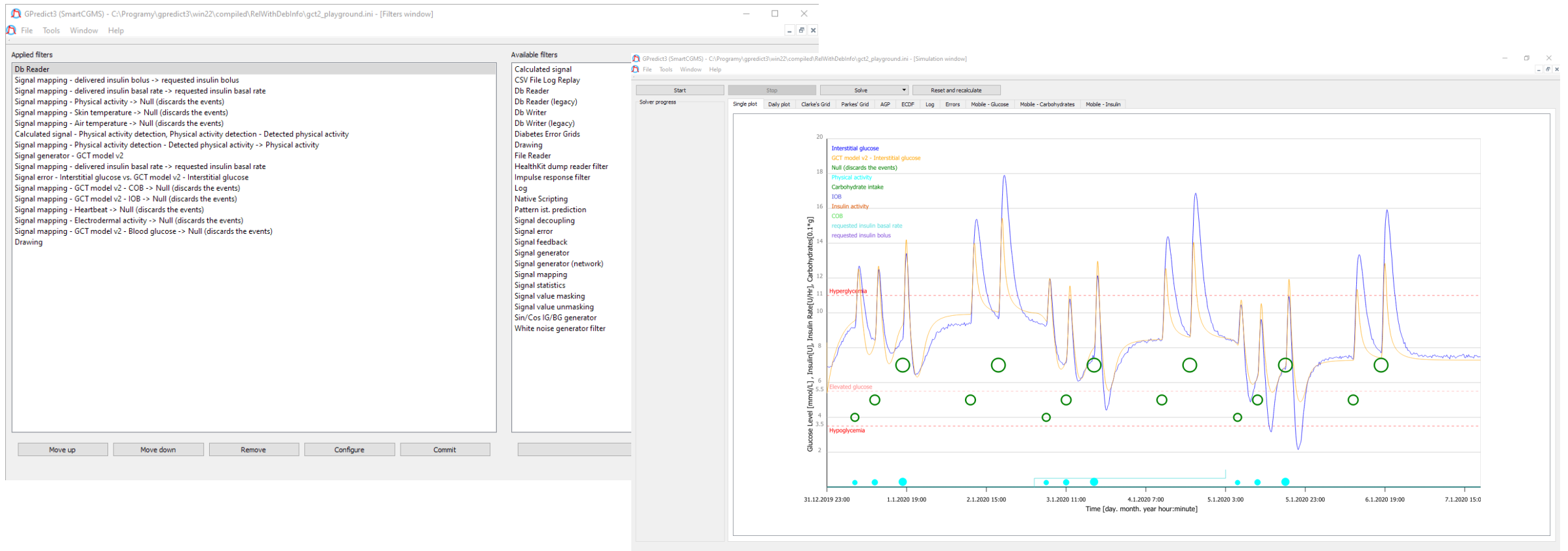
SmartCGMS

Sám o sobě back-end – framework, sada komponent a SDK

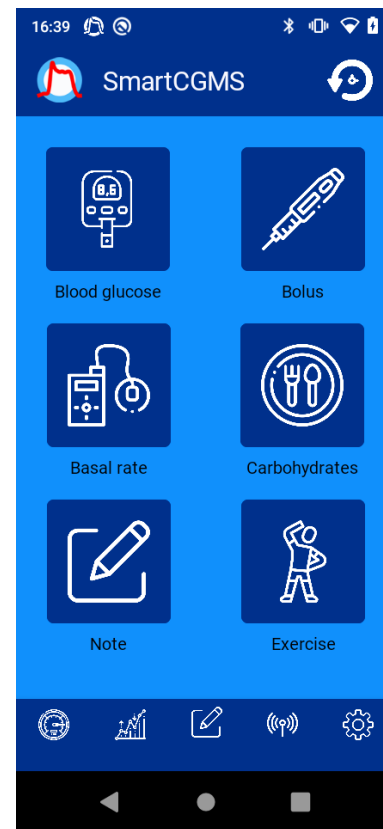
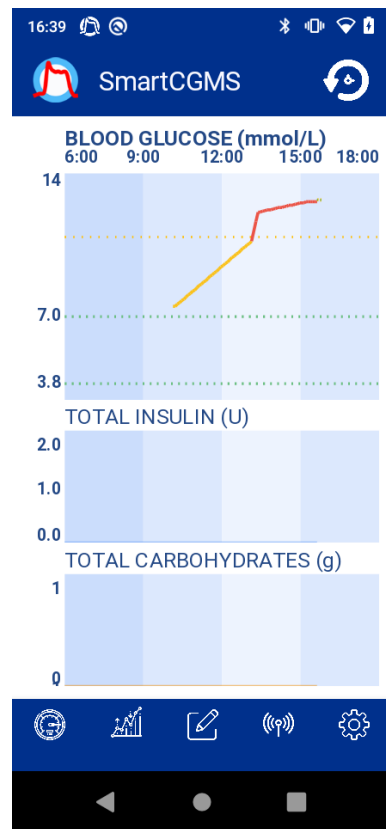
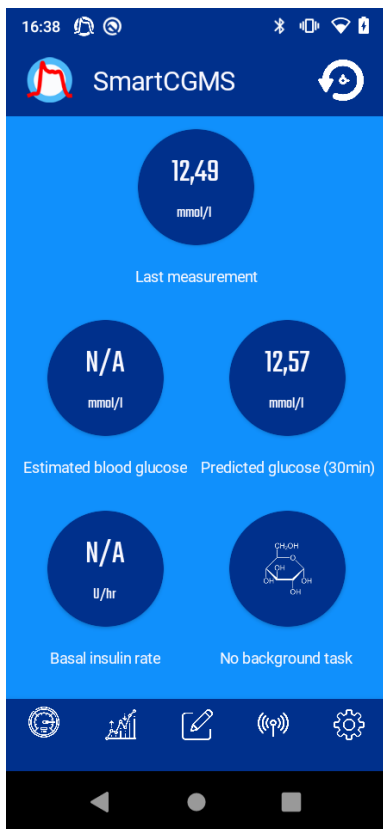
Front-endy

- gpredict3 – pro výzkum
- SmartCGMS Mobile – pro monitoring na straně pacienta
- Íkarus má Diabetes – hra
- Trenažér – pro nově diagnostikované - seznámení s nemocí

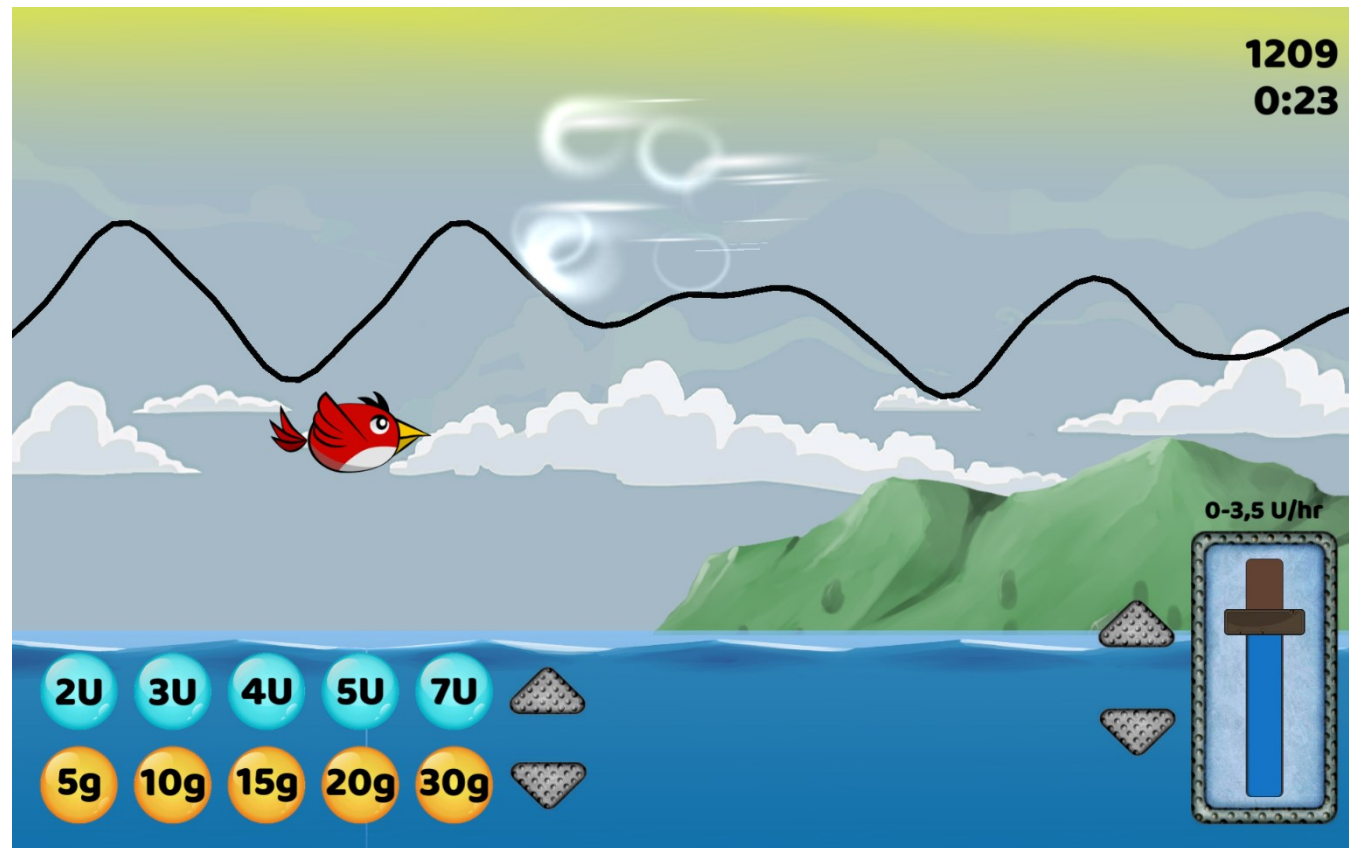
SmartCGMS – gpredict3



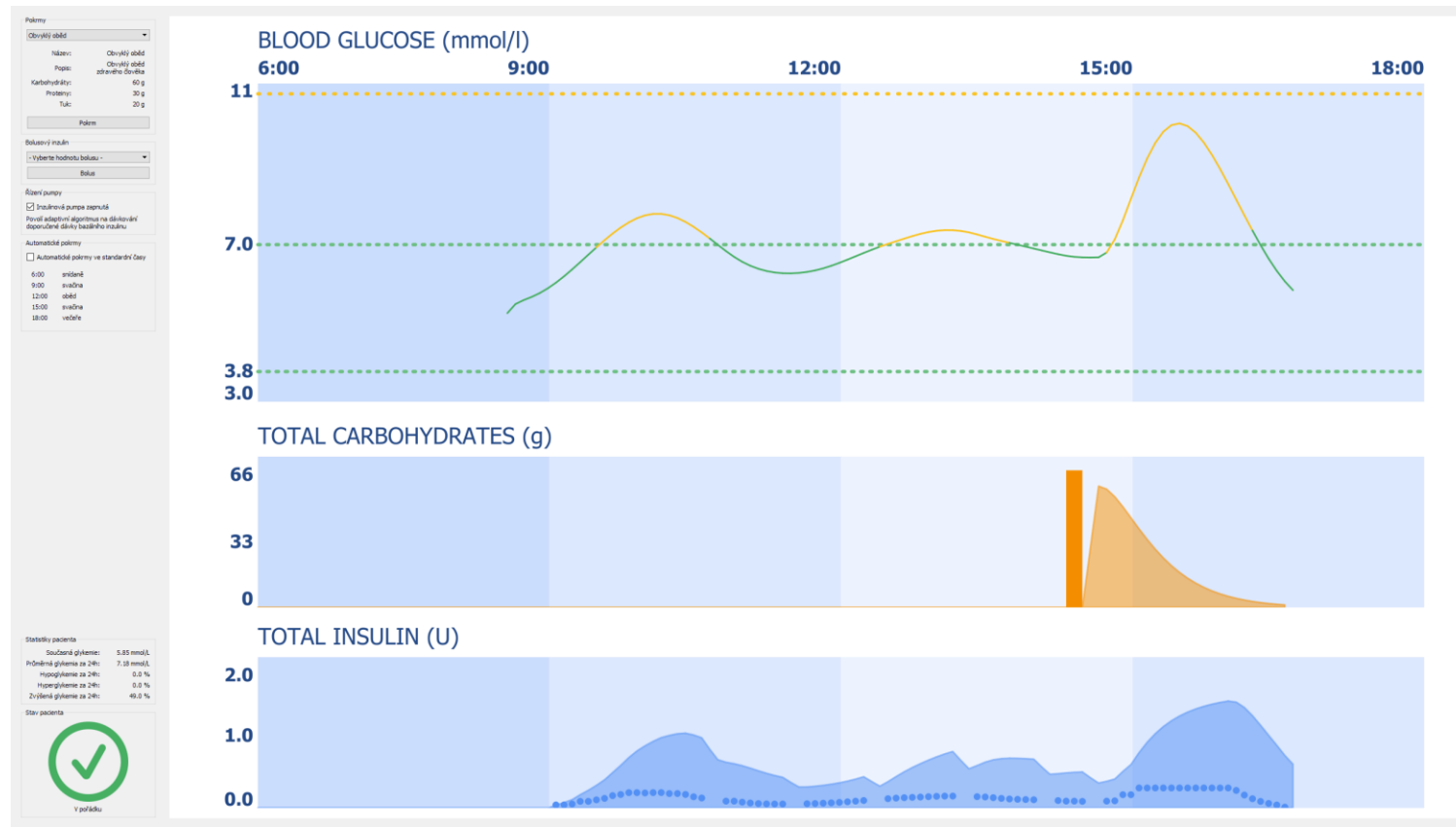
SmartCGMS – Mobile



SmartCGMS – Íkaros má Diabetes



SmartCGMS – Trenažér



Děkuji za pozornost

Dotazy, diskuse...

