

Insulin Pump

Progetto di Ingegneria del Software

Daniele Marretta, Tommaso Maria Lopedote, Jacopo Spallotta



Indice

Introduzione

1. Descrizione Generale

2. Requirements

2.1 Insulin Pump

2.1.1 System Requirements

2.1.1.1 Functional

2.1.1.2 Non-Functional

2.1.2 User Requirements

2.1.2.1 Functional

2.1.2.2 Non-Functional

2.2 Patient

2.2.1 System Requirements

2.2.1.1 Functional

2.2.1.2 Non-Functional

2.2.2 User Requirements

2.2.2.1 Functional

2.2.2.2 Non-Functional

3. Diagrams

3.1 Use-Case UML

3.2 Architettura del Sistema

3.3 State Diagram

3.4 Activity Diagram UML

3.5 Message-Sequence-Chart UML

3.6 Test

4. Componenti

4.1 Modelli

4.1.1 Patient

4.1.1.1 Inizializzazione

4.1.1.2 Calcolo del Glucosio

4.1.2 Insulin Pump

4.1.3 Environment

4.1.4 Monitors

4.2 Display (Interfaccia grafica)

5. Database e Logs

5.1 Patient

5.2 Insulin Pump

5.3 Environment

5.4 Monitors

5.5 Records

5.6 ER Database

6. Server redis

Introduzione

Il sistema “Insulin Pump” è un modello per la gestione, monitoraggio e controllo per pazienti diabetici che fanno uso di una Pompa di Insulina. Il sistema è in grado di regolare i livelli di glucosio all'interno del sangue del paziente rilasciando dosi di Insulina le quali assistono il paziente nel regolare assorbimento degli zuccheri nei tessuti.

1. Descrizione Generale

Il sistema “Insulin Pump” è organizzato in due componenti principali, *Patient* che modella un paziente con diabete di tipo 2 e *Insulin Pump* che modella la pompa dell'insulina. I due modelli comunicano fra di loro tramite due *streams* gestiti da Redis. Il modello *Patient* comunica, inoltre, con un modello *Environment* il quale gestisce i periodi di digiuno e alimentazione del paziente stesso. Entrambi i modelli *Patient* e *Insulin Pump* comunicano con un *Database* sul quale registrano i valori calcolati al momento della loro esecuzione. Infine per verificare che i requisiti funzionali e di sistema previsti dal progetto siano soddisfatti, sono stati implementati tre *monitors* che controllano i logs registrati nel *Database* per osservare eventuali violazioni.

2. Requirements

2.1 Insulin Pump

2.1.1 System Requirements

2.1.1.1 Functional

1. Il livello di glucosio non deve scendere sotto i 50mg/dL (ipoglicemia);
2. Il livello di glucosio non deve salire sopra i 120 mg/dL (iperglicemia);
3. Il livello di glucosio dovrebbe rimanere il più vicino possibile ai 100mg/dL;
4. Il sistema analizza il trend di glicemia in salita o in discesa nel sangue;
5. Con glicemia nei limiti di sicurezza:
 1. In caso di livelli stabili o in discesa di glucosio la pompa non rilascia insulina;
 2. In caso di aumento di glucosio ma tasso di crescita in discesa la pompa non rilascia insulina;
 3. In caso di aumento di glucosio ma tasso di crescita in ascesa la pompa calcola la dose di insulina appropriata;
 4. In caso di trend stabile la pompa non rilascia insulina;
6. Con glicemia alta:
 1. In caso di aumento di glucosio la pompa rilascia una dose appropriata;
 2. In caso di livello glucosio stabile la pompa rilascia una dose minima;
 3. In caso di diminuzione di glucosio ma tasso di decremento in ascesa la pompa non rilascia insulina;
 4. In caso di diminuzione di glucosio con tasso di decremento in discesa la pompa rilascia una dose minima;

2.1.1.2 Non Functional

1. La dose di insulina giornaliera deve essere minimizzata;

2.1.2 User Requirements

2.1.2.1 Functional

1. Garantire un livello di glucosio nel sangue intorno ai 100mg/dL;

2.1.2.2 Non Functional

1. Minimizzare il quantitativo di insulina iniettata giornalmente;
2. Massimizzare il tempo fra un controllo e il successivo, mantenendosi fra 5 e 10 minuti;

2.2 Patient

2.2.1 System Requirements

2.2.1.1 Functional

1. Il sistema consente l'inserimento di parametri peso, età, altezza e sesso per la definizione delle caratteristiche del paziente;
2. Il sistema esegue il logging sul database di parametri chiave per la definizione dello stato del paziente:
 1. Livello del glucosio nel sangue (G - mg/dL);
 2. Contrazione dell'insulina nel Plasma (I - pmol/L);
 3. Quantitativo di glucosio nello stomaco (Q_{sto} - mg);
 4. Produzione endogena di glucosio (EGP - adimensionale);
 5. Utilizzo dipendente di insulina (UID - mg/kg/min);
 6. Escrezione renale (E - mg/kg/min);
 7. Secrezione di Insulina e C-peptide (ISR - pmol/min);

2.2.2 User Requirements

2.2.2.1 Functional

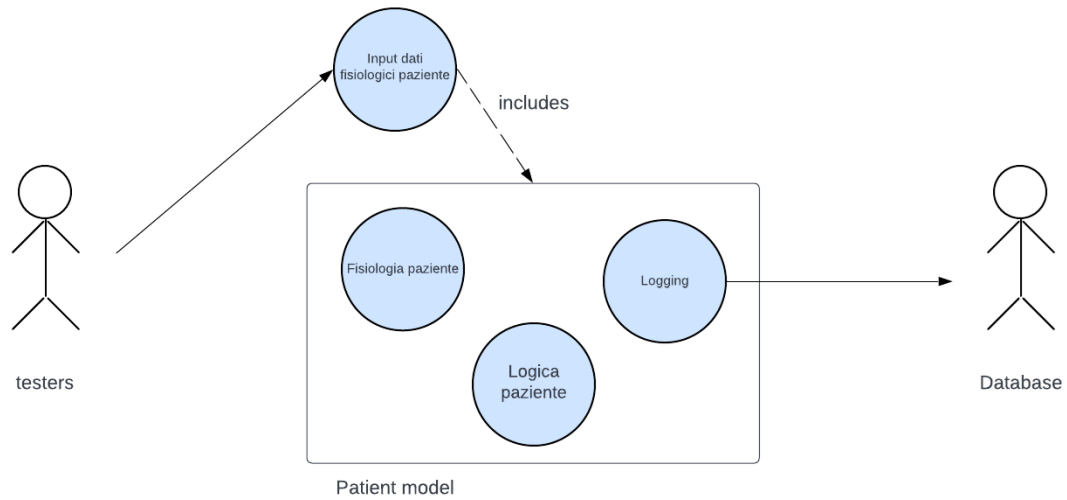
1. Il modello riproduce fedelmente un paziente con diabete di tipo 2;

3. Diagrams

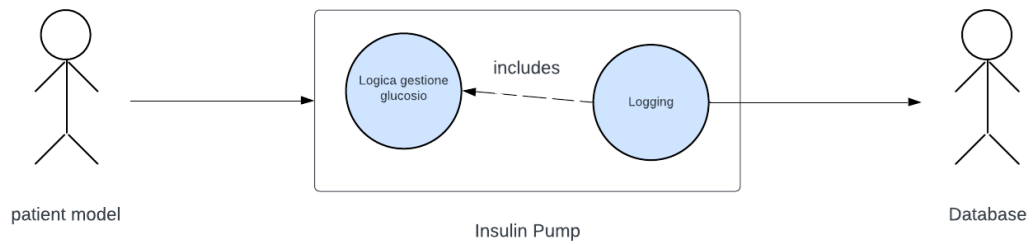
3.1 Use Case UML

Use Case UML di

1. Patient Model per la registrazione nel database dei valori necessari al calcolo del glucosio.

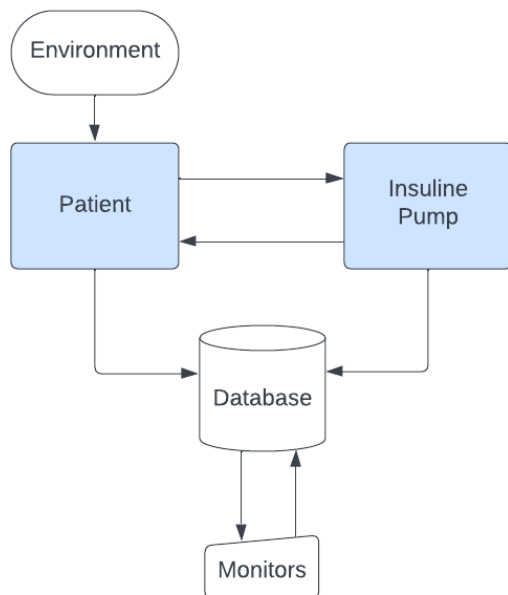


2. Insulin Pump model per la registrazione nel database del valore di insulina somministrata e relativo stato



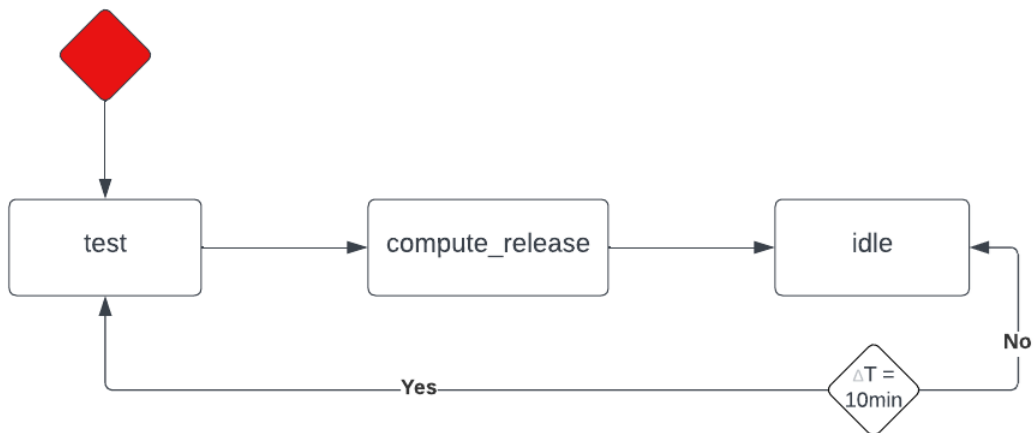
3.2 Architettura del sistema

Architettura del sistema che comprende due modelli (Patient ed Insuline Pump) comunicanti tra loro tramite un server Redis, sul quale agisce un Environment e salva i dati su un Database globale.



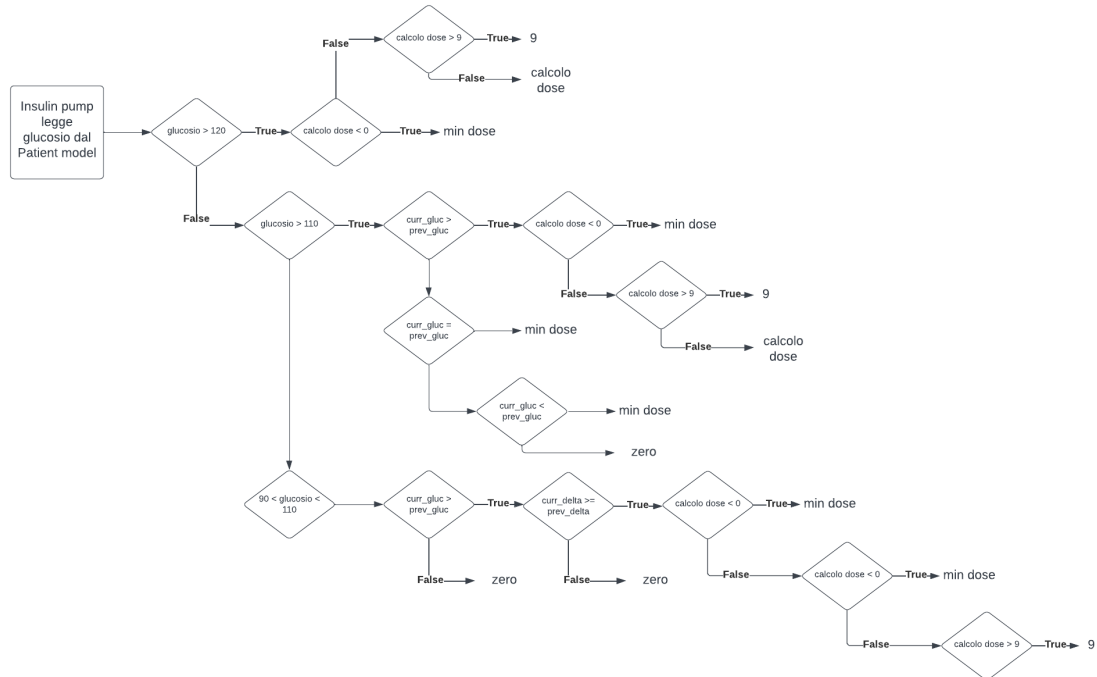
3.3 State Diagram UML

State Diagram UML per gli stati in cui può trovarsi la pompa di insulina.



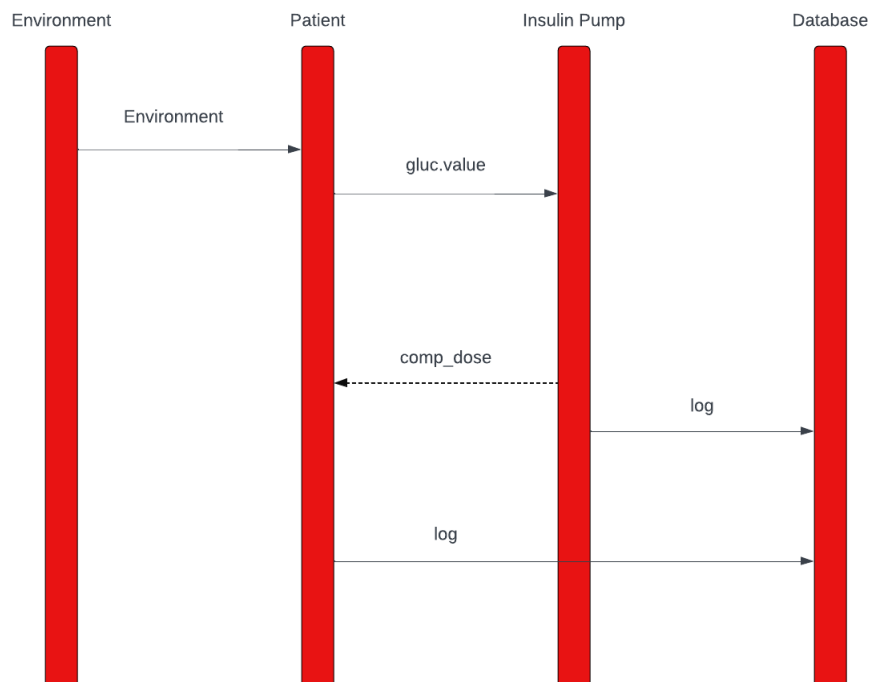
3.4 Activity Diagram UML

Activity Diagram UML per l'inserimento e il calcolo dei valori necessari all'inizializzazione del Patient



3.5 Message Sequence Chart UML

Message Sequence Chart UML per la comunicazione tra Modelli-Server-Database

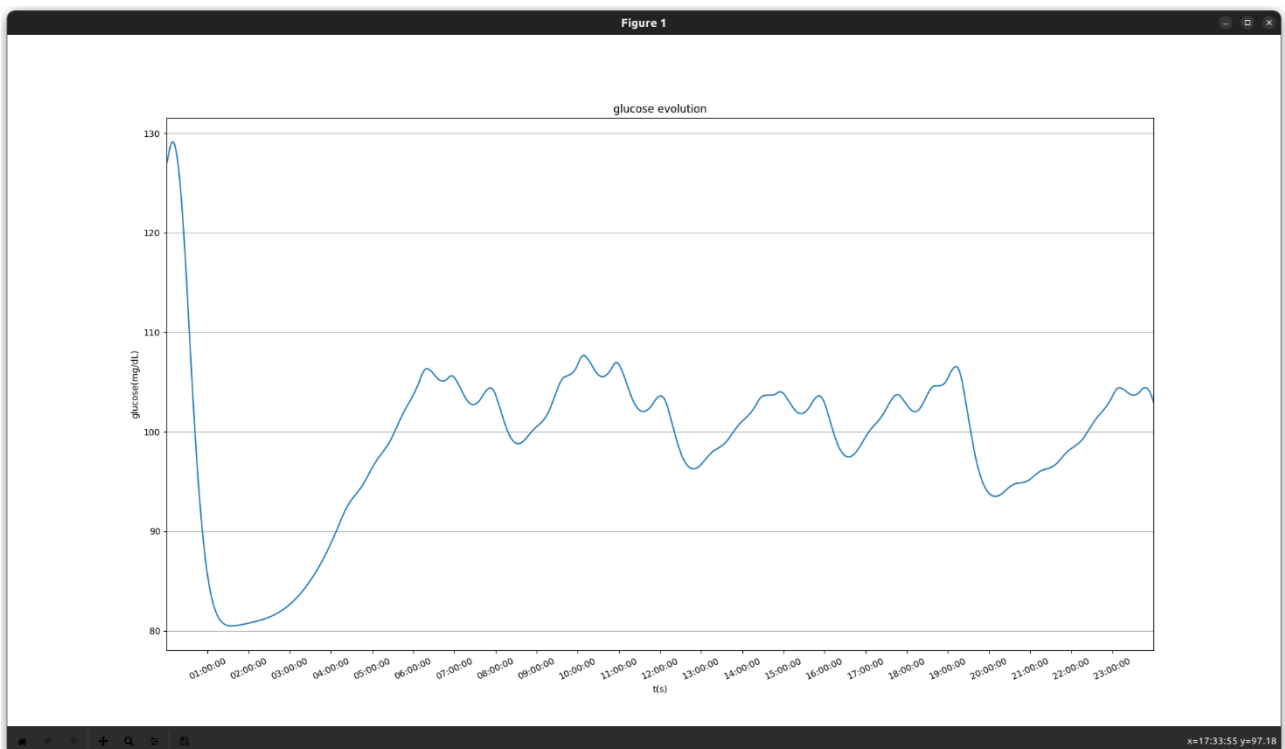
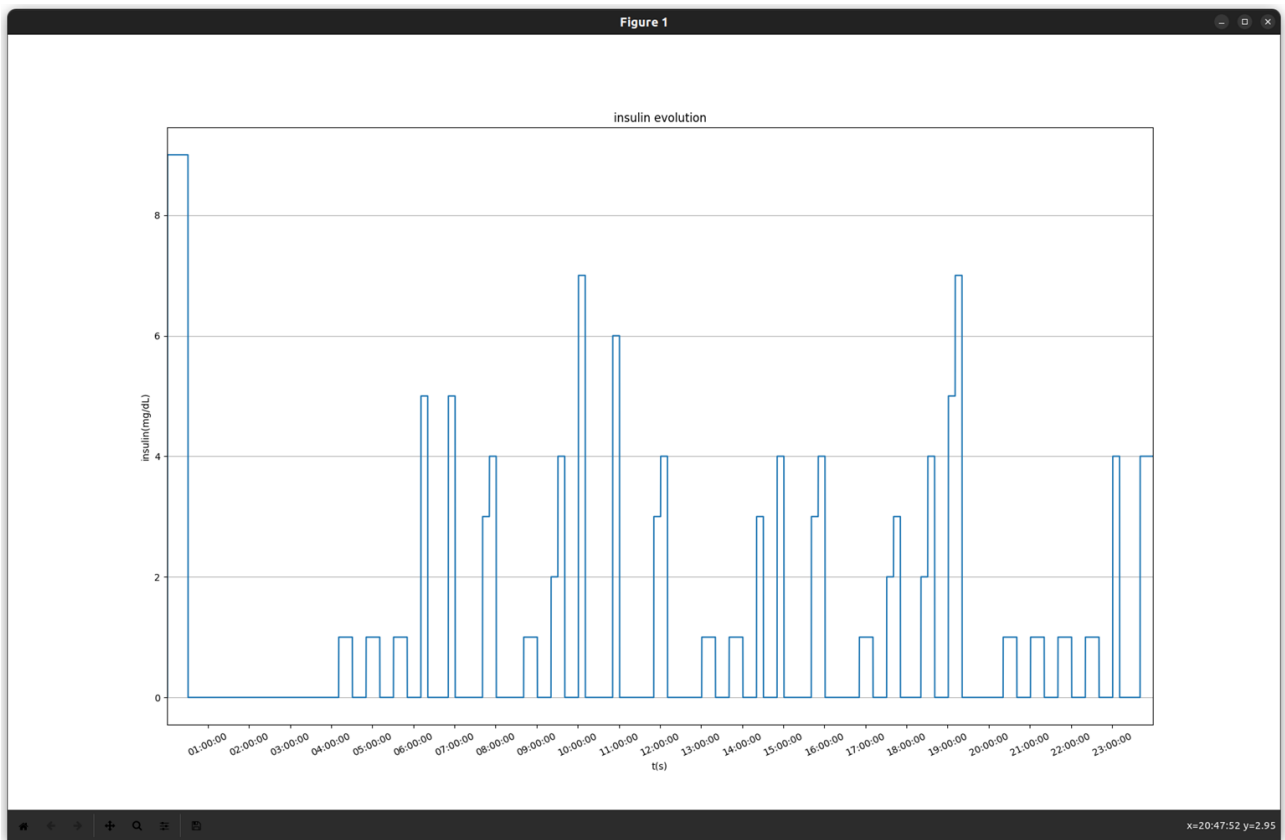


3.6 Risultati sperimentali

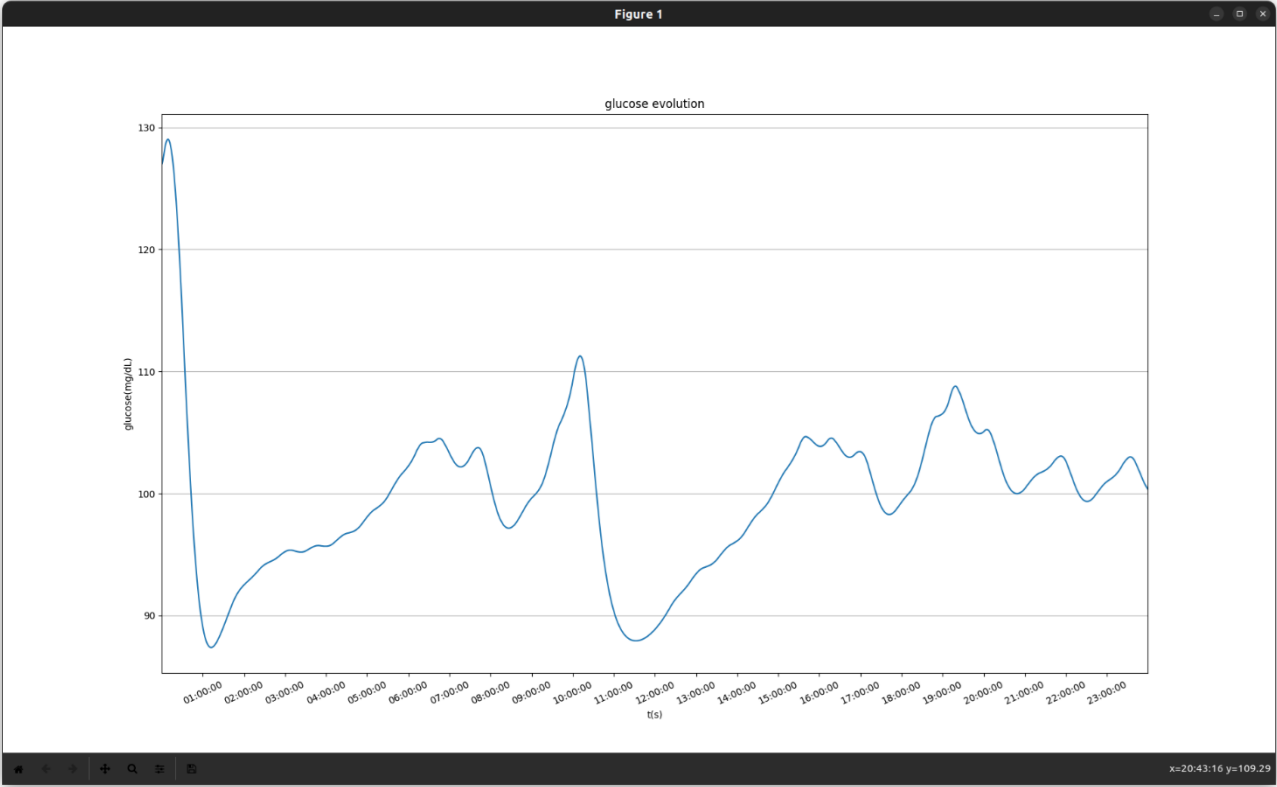
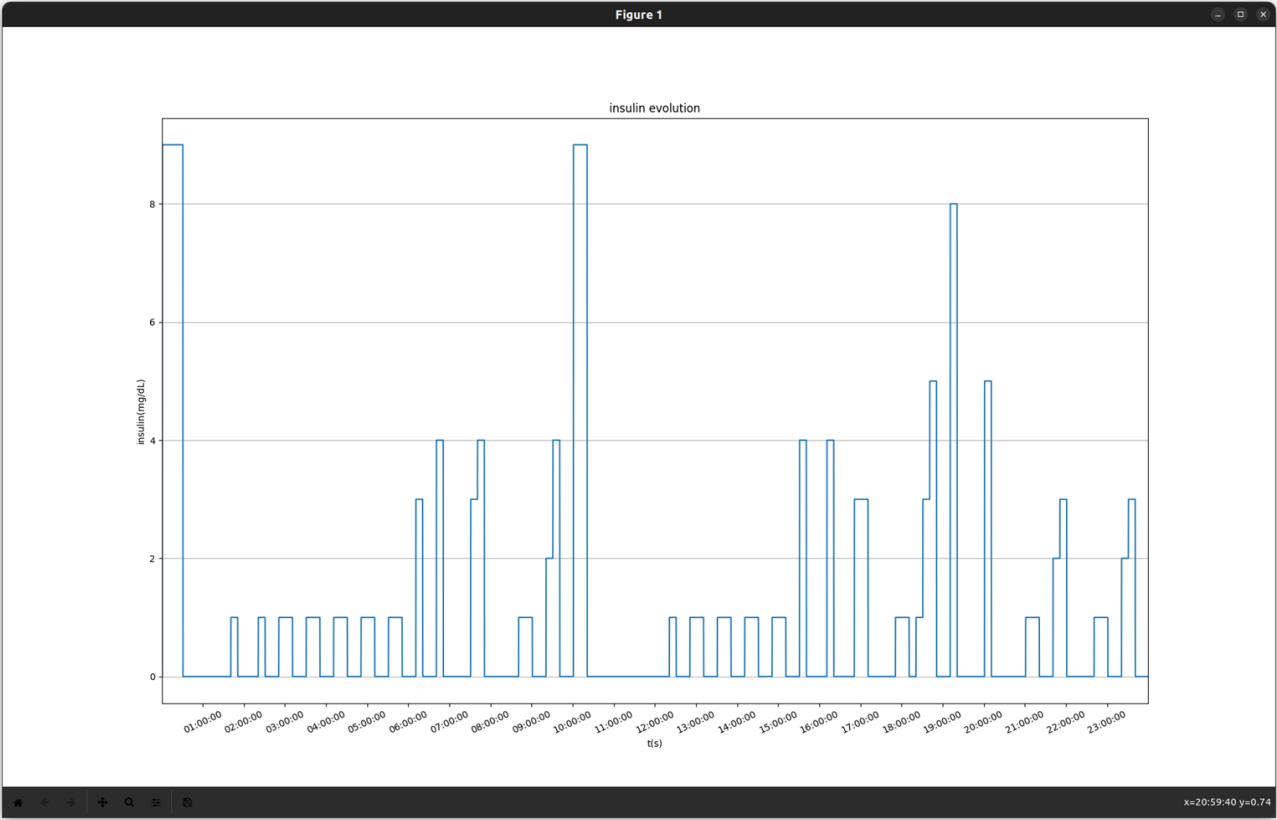
I risultati sperimentali sono stati effettuati su pazienti con caratteristiche differenti per testare in modo completo il modello dell'Insulin Pump, in particolar modo abbiamo i seguenti pazienti

	Sesso	Altezza	Peso	Età
<i>Paziente 1</i>	Uomo	175cm	100kg	40
<i>Paziente 2</i>	Uomo	180cm	60kg	30
<i>Paziente 3</i>	Uomo	180cm	80kg	55
<i>Paziente 4</i>	Donna	160cm	90kg	45
<i>Paziente 5</i>	Donna	160cm	55kg	35
<i>Paziente 6</i>	Donna	150cm	100kg	60

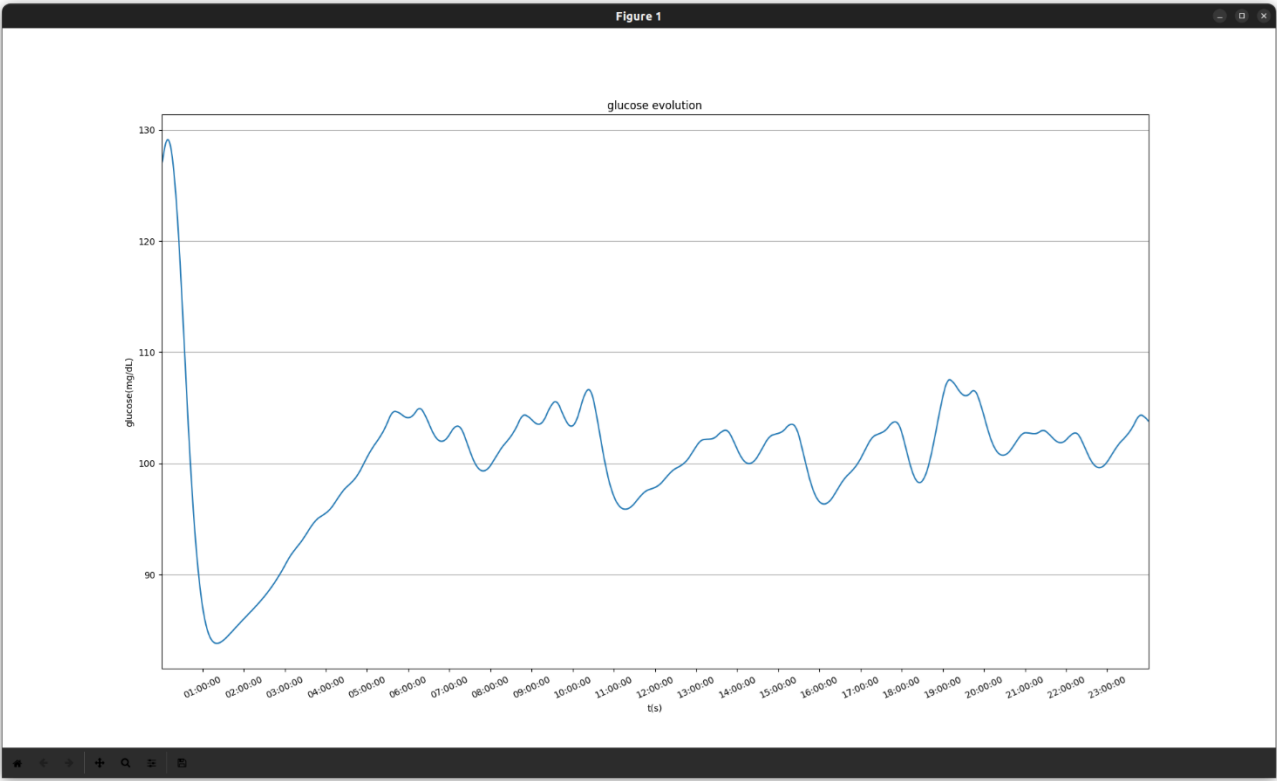
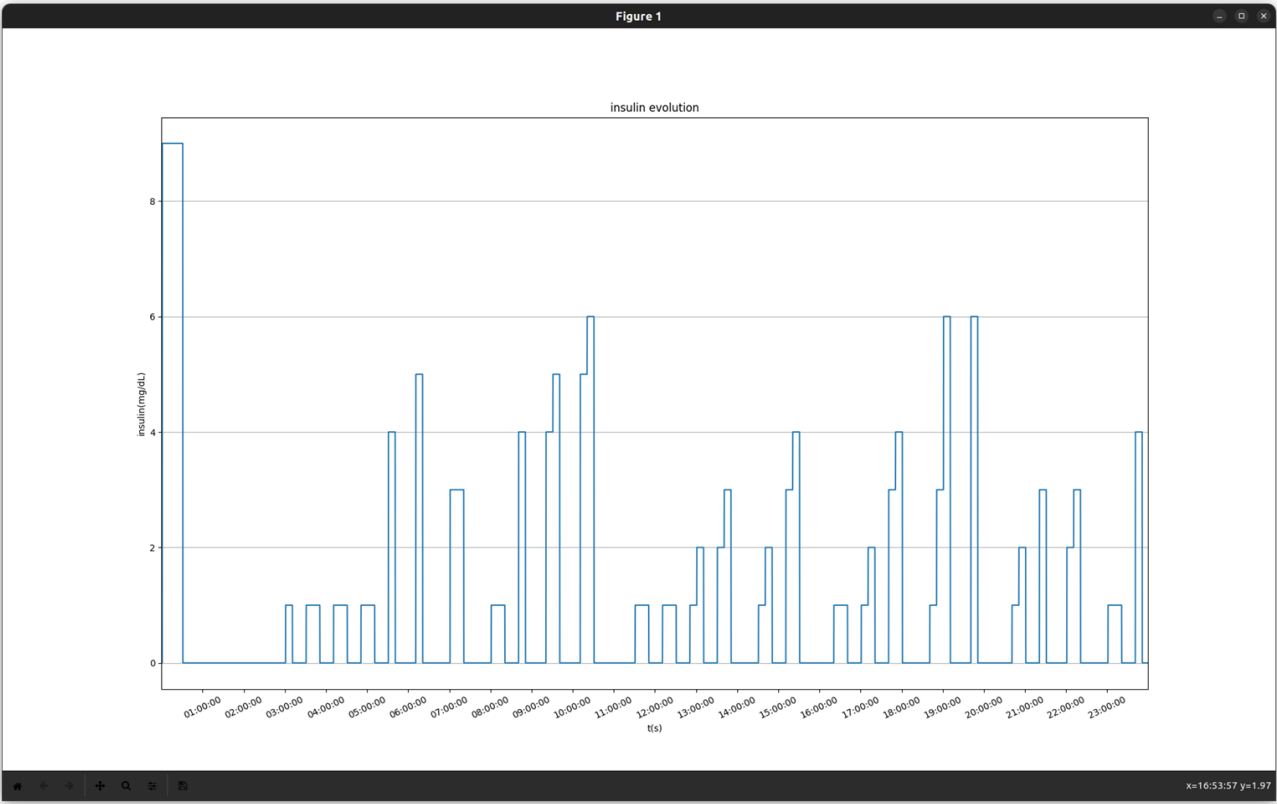
Paziente 1 (Uomo, 175cm, 100kg, 40 anni)



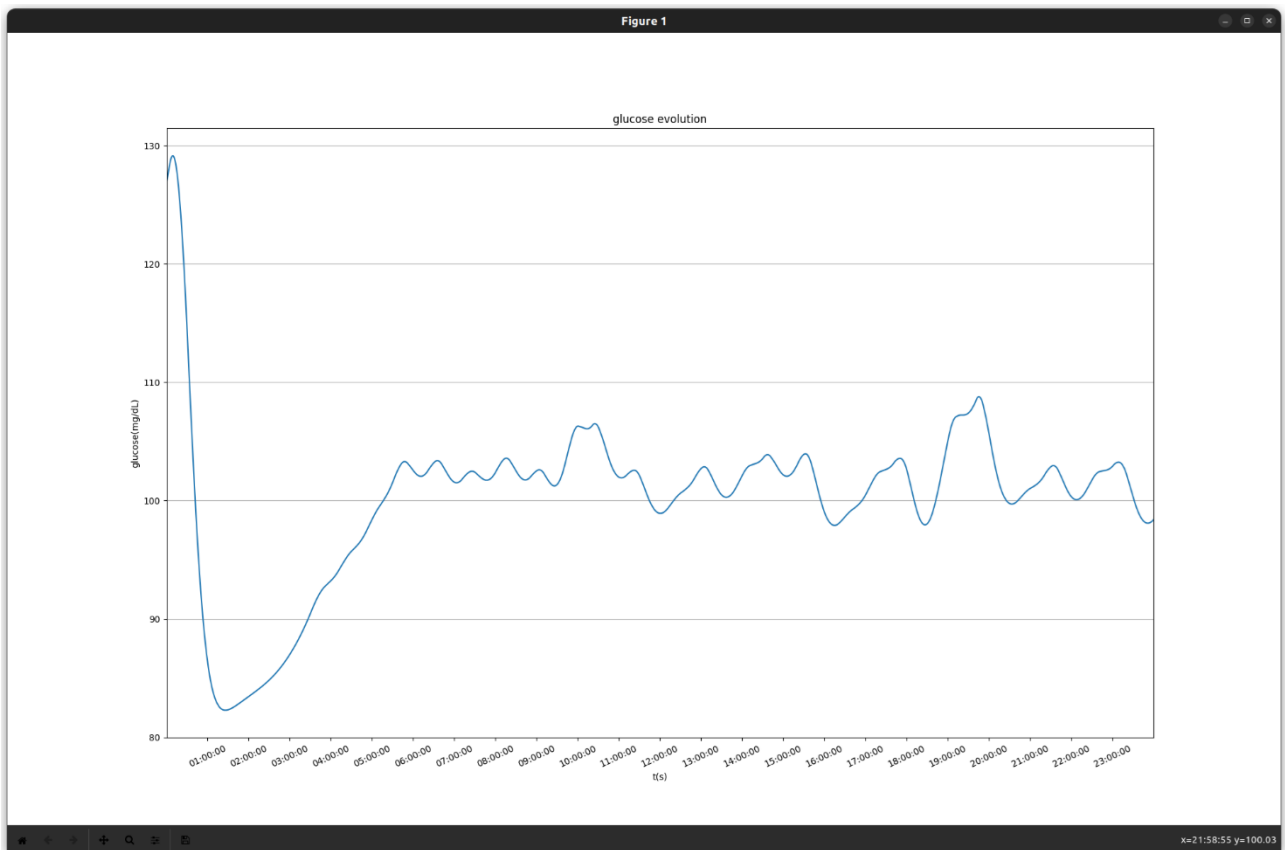
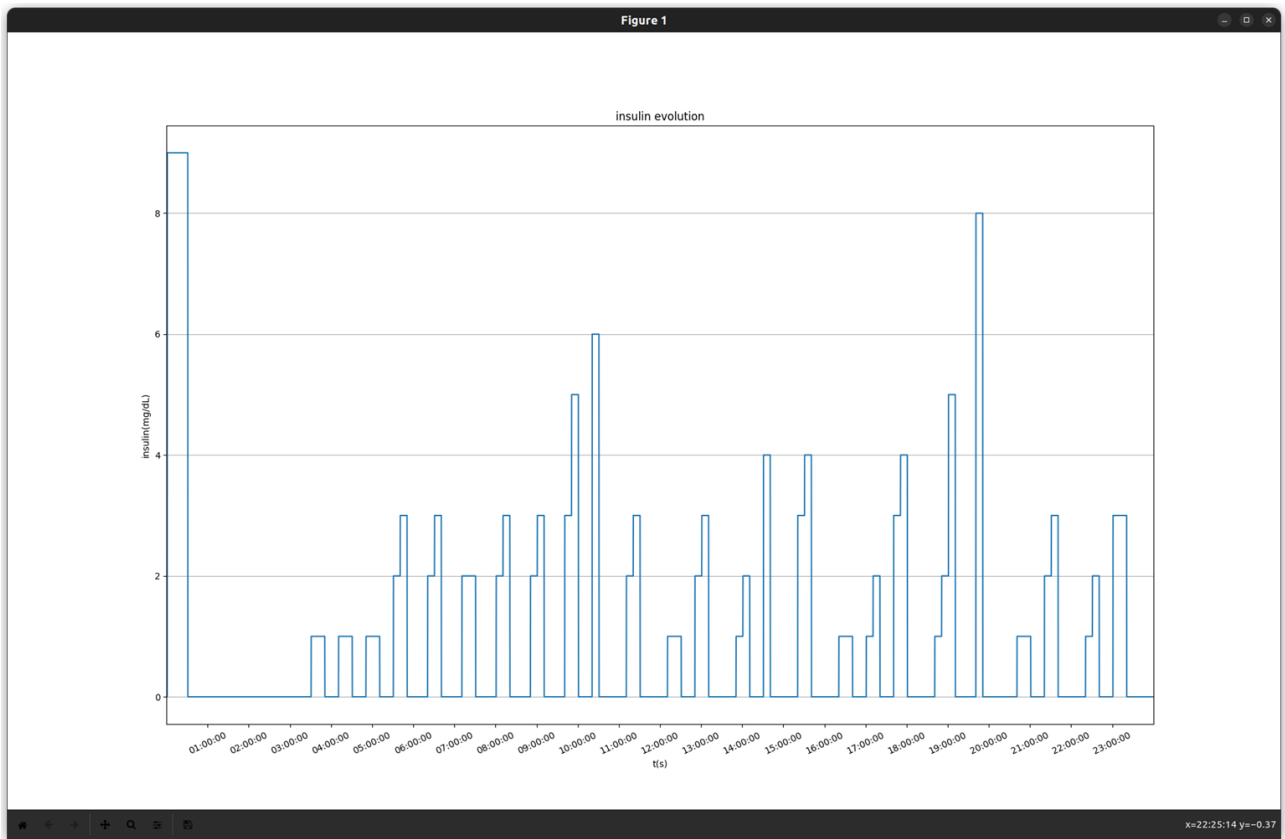
Paziente 2 (Uomo, 180cm, 60kg, 30 anni)



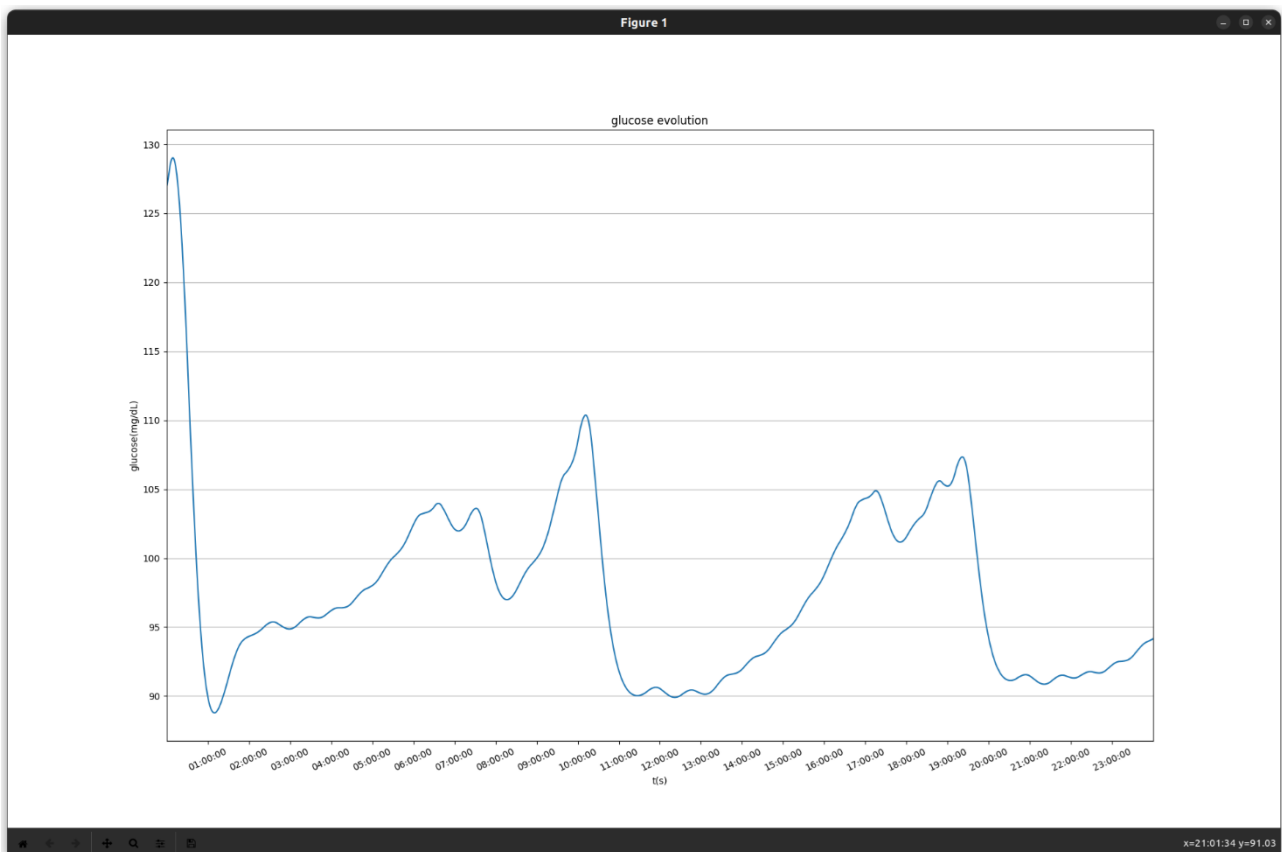
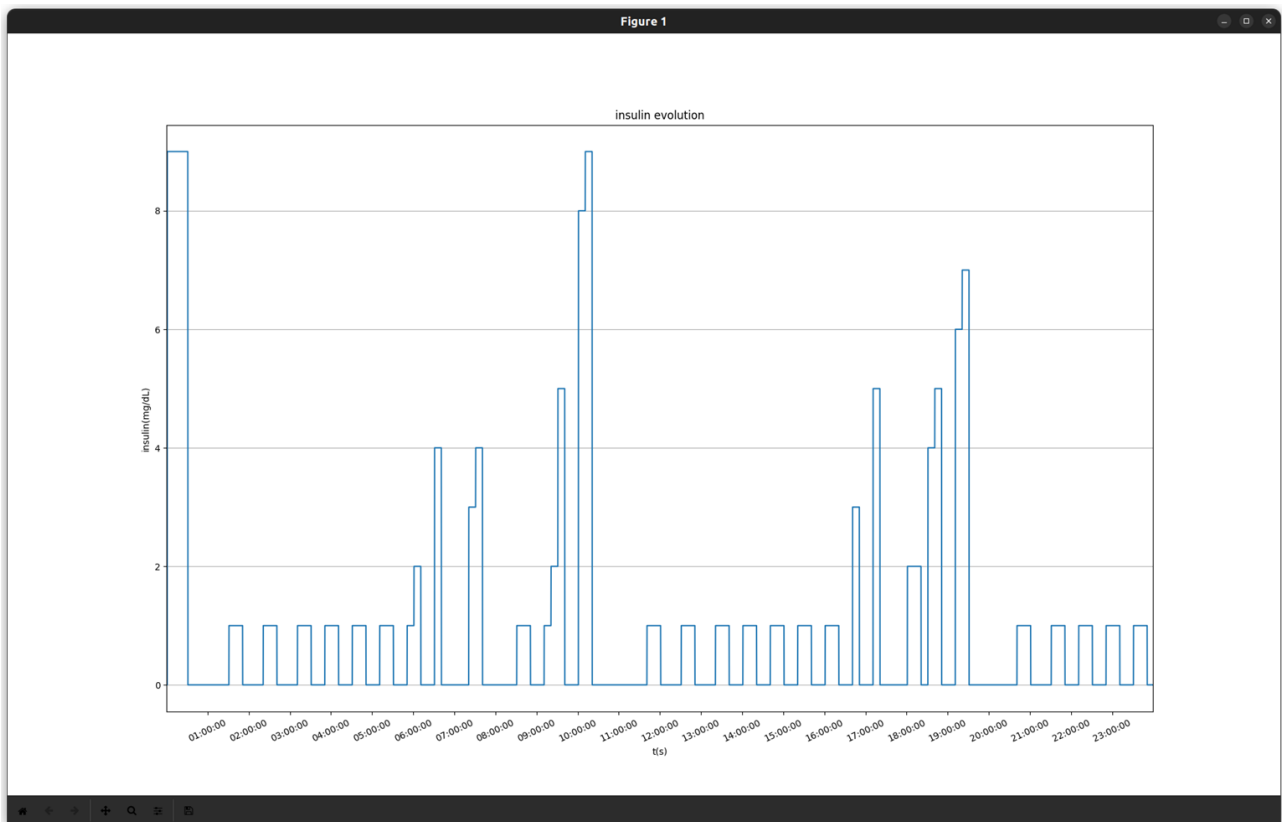
Paziente 3 (Uomo, 180cm, 80kg, 55 anni)



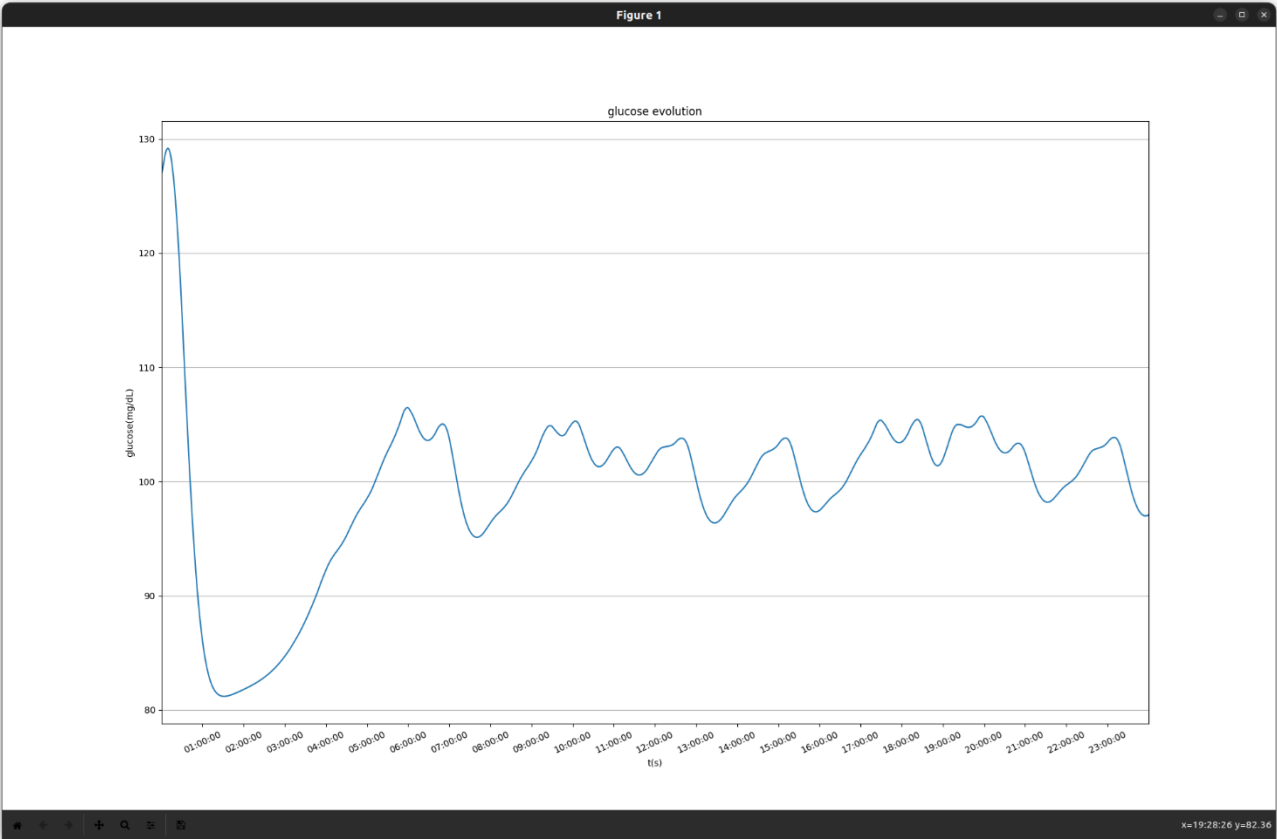
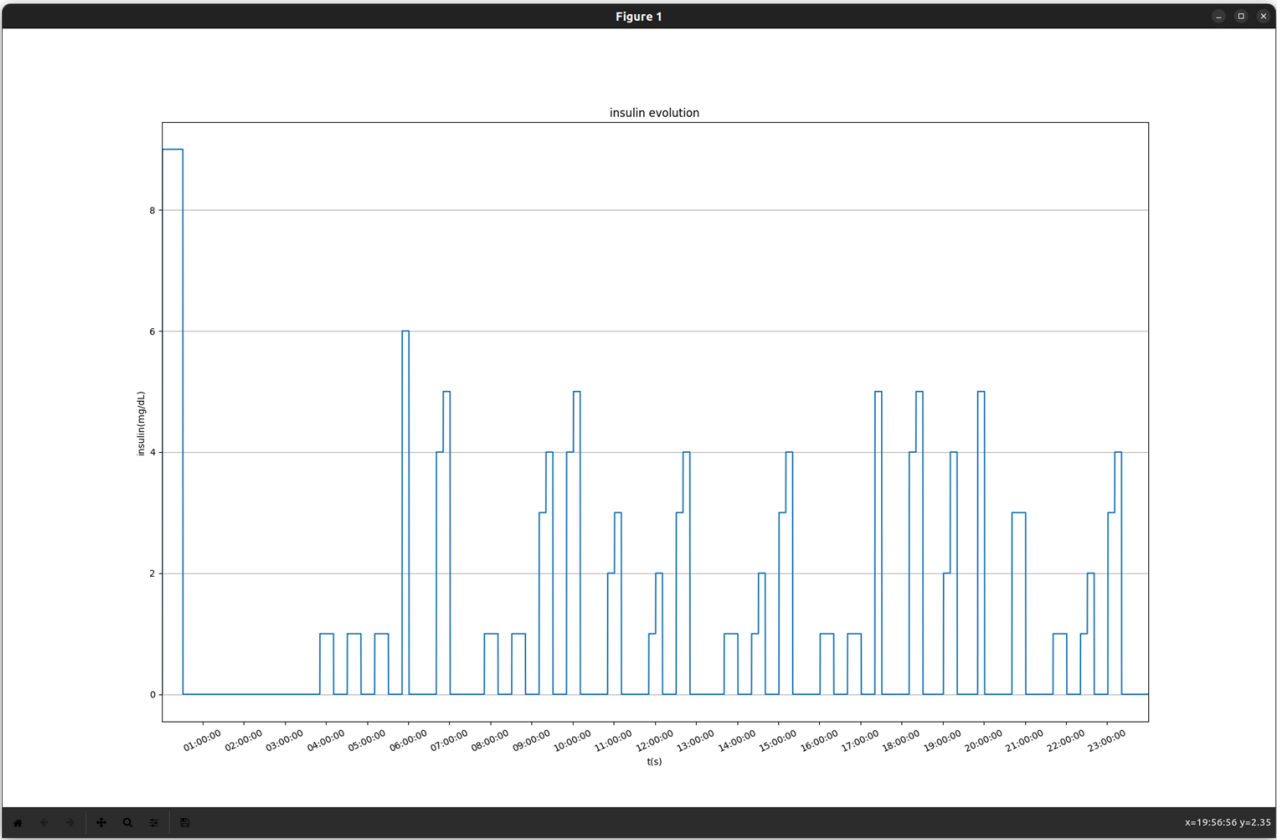
Paziente 4 (Donna, 160cm, 90kg, 45 anni)



Paziente 5 (Donna, 160cm, 55kg, 35 anni)



Paziente 6 (Donna, 160cm, 55kg, 35 anni)



4. Componenti

4.1 Models

4.1.1 Patient

Calcola il valore di insulina necessario in base alla concentrazione di glucosio attuale nel sangue e valori correlati. Il modello “Patient” simula un paziente in grado di interagire con il modello “Insulin Pump” della Pompa di Insulina. La simulazione calcola in modo dinamico i livelli di glucosio nel sangue e l’insulina necessaria a regolare il valore del glucosio stesso.

4.1.1.1 Inizializzazione

La fase di inizializzazione del Paziente richiede alla base 4 parametri necessari

- Età
- Sesso
- Peso
- Altezza

Attraverso i valori dei parametri principali ne vengono calcolati altri 6

- l'indice di massa corporea (BMI): composizione corporea in relazione all'altezza di un individuo
- l'Area Superficiale Corporea (BSA): misura della superficie esterna del corpo di un individuo
- Volume di distribuzione centrale dell'insulina (V_c): volume apparente nel quale l'insulina è distribuita all'interno del corpo
- B_1 : coefficiente o un parametro che influisce sulla velocità o sulla cinetica del glucosio o dell'insulina all'interno del sistema.
- A_1 e FRA: i quali valori vengono stabiliti in relazione al BMI

4.1.1.2 Calcolo del glucosio

Il calcolo del valore del glucosio nel sangue avviene all'interno di un ciclo while e viene aggiornato ad ogni iterazione per ottenere ad ogni intervallo di tempo un valore (possibilmente anche nullo) di insulina necessaria.

I valori necessari per il calcolo del glucosio appartengono alle seguenti componenti

- Glucose Kinetics: *velocità di assorbimento, distribuzione, metabolismo ed eliminazione del Glucosio nel corpo*
- Insulin Kinetics: *velocità di assorbimento, distribuzione, metabolismo ed eliminazione dell'Insulina nel corpo*
- Rate of Glucose Appearance: *velocità con la quale il glucosio appare nel flusso sanguigno*
- Endogenous Glucose Production: *quantità di glucosio prodotta internamente dal corpo*
- Glucose Utilization: *eliminazione del glucosio attraverso i reni*
- C-Peptide Kinetics: *velocità di assorbimento, distribuzione ed eliminazione del C-peptide*

- Insulin and C-Peptide Secretion: *processo di rilascio di insulina e C-peptide da parte delle cellule beta del pancreas in risposta all'aumento di glucosio nel sangue*

Le formule per il calcolo del glucosio sono discretizzate in base al tempo, dunque nel calcolo dei valori di ogni ad ogni componente al tempo $t+1$ sono necessari i relativi valori al tempo t .

4.1.2 Insulin Pump

Riceve dal paziente il valore di insulina di cui ha bisogno e la inietta a determinate condizioni. Il modello *Insulin Pump* simula il comportamento di una Pompa di Insulina. Il sistema è caratterizzato da una serie di controlli e basato su un ciclo temporale. Nell'esecuzione Il programma stampa periodicamente lo stato della pompa sulla console e continua l'esecuzione in modo indefinito.

La struttura *Insulin Pump* viene inizializzata con parametri relativi a:

1. Stato {*Idle, test, compute_release*};
2. Livello di glucosio nel plasma;
3. Livello glucosio 10 minuti prima, memoria del test subito precedente;
4. Livello glucosio 20 minuti prima, memoria di due esecuzioni precedenti;

Il funzionamento del sistema si basa anch'esso su un ciclo while che simula il passare del tempo, attraverso il quale viene calcolato lo stato successivo in cui si trova la pompa utilizzando la funzione next.

4.1.3 Environment

L'Environment ha come compito principale di indicare al modello del Paziente se esso stia mangiando.

La simulazione del sistema è basata su un timer assoluto grazie al quale il Paziente può alternare lo stato di alimentazione con lo stato di digiuno, indicato attraverso la variabile *delta*. La durata di entrambe le fasi, *meal_duration* e *fasting_duration* sono modificabili.

Di default il Paziente si alimenta ogni 8 ore per un'ora.

4.1.4 Monitors

I Monitors presenti all'interno del sistema sono fondamentalmente tre, di cui due si occupano del controllo del livello del glucosio nel sangue. In particolar modo il primo monitor si assicura che il valore del glucosio non scenda mai sotto i 50 mg/dL, mentre il secondo che il livello di glucosio sia più vicino possibile ai 100 mg/dL.

Infine il terzo monitor mostra la dose giornaliera di insulina impiegata durante la giornata.

Monitor per il controllo dei livelli minimi di glucosio mg/dL

Questo monitor è progettato per garantire che i livelli di glucosio nel sangue non scendano al di sotto del valore di riferimento di 50 mg/dl. È fondamentale prevenire l'ipoglicemia per evitare complicazioni pericolose per la salute, come perdita di coscienza o coma. Il monitor monitora costantemente i livelli di glucosio e invia segnali alla pompa per erogare l'insulina in modo appropriato per mantenere il livello di glucosio sopra il valore soglia.

Monitor per il controllo dei livelli di glucosio prossimi a 100 mg/dL

Questo monitor ha il compito di garantire che i livelli di glucosio siano mantenuti il più possibile vicino al valore target di 100 mg/dl. Questo valore è considerato ideale per mantenere un controllo glicemico ottimale nel sangue. Il monitor monitora costantemente i livelli di glucosio e ogni ora verifica se il livello sia compreso fra 90 e 110 mg/dL.

Monitor per il controllo dell'Insulina giornaliera e dose minima

Lo scopo di questo monitor è garantire una distribuzione ottimale dell'insulina nel corso del tempo, minimizzando al massimo le dosi somministrate. In questo modo si evita la somministrazione di dosi eccessive in un breve periodo, favorendo una distribuzione più uniforme nel corso della giornata.

4.3 Display (Interfaccia Grafica)

Funge da interfaccia con l'utente mostrando i valori attuali di glucosio nel sangue ed insulina somministrata. La console ad ogni intervallo di tempo aggiorna

- Il valore attuale del glucosio nel sangue del paziente;
- Il valore di Insulina fornita dalla Pompa di Insulina;
- Il tempo
- Lo stato del paziente (se sta mangiando o meno)

Di seguito un'anteprima dell'interfaccia grafica



5. Database e Logs

Il Database utilizzato nel sistema, implementato attraverso l'utilizzo di PostgreSQL, è strutturato per organizzare i log generati dai modelli Patient e Insulin Pump, che vengono salvati nelle tabelle del database attraverso record con attributi associati. Di seguito analizziamo in maniera più dettagliata i record generati da Patient, Insulin Pump ed Environment

Per ogni record inserito si hanno 5 attributi che indicano

- *tempo* assoluto in nanosecondi in cui sono stati inseriti i dati
- *vid* id univoco associato alle variabili inserite
- *varvalue* valore della grandezza associata al *vid* corrispondente
- *loginfo* informazione dettagliata sulla tipologia del valore (es. Glucose)
- *t* unità di tempo interno al sistema

5.1 Patient

I record del **Patient** inseriti all'interno della **LogTable** sono relativi al

1. Glucose value
2. Insulin value
3. Endogenous Glucose Production
4. Glucosio nello stomaco
5. Insulin utilization
6. Renal Excretion
7. Insulin secretion

Ogni record possiede 5 attributi come indicato in 5.

5.2 Insulin Pump

I record dell'**Insuline Pump** inseriti all'interno della **LogTable** sono

1. Stato della pompa di insulina
2. Dose calcolata (comp_dose)

Ogni record possiede 5 attributi come indicato in 5.

5.3 Environment

L'unico record dell'**Environment** inserito all'interno della **LogTable** è il

1. Cibo assunto (delta)

Ogni record possiede 5 attributi come indicato in 5.

5.4 Monitors

I record dei **Monitor** inseriti all'interno della **LogTable** riguardano la **safetyTable**, la **livenessTable** e la **minimalDose**. Rispettivamente sono composti da

- 1. Timestamp e valore *safety*
- 2. Timestamp e valore *safety*
- 3. Timestamp e valore *min_dose*

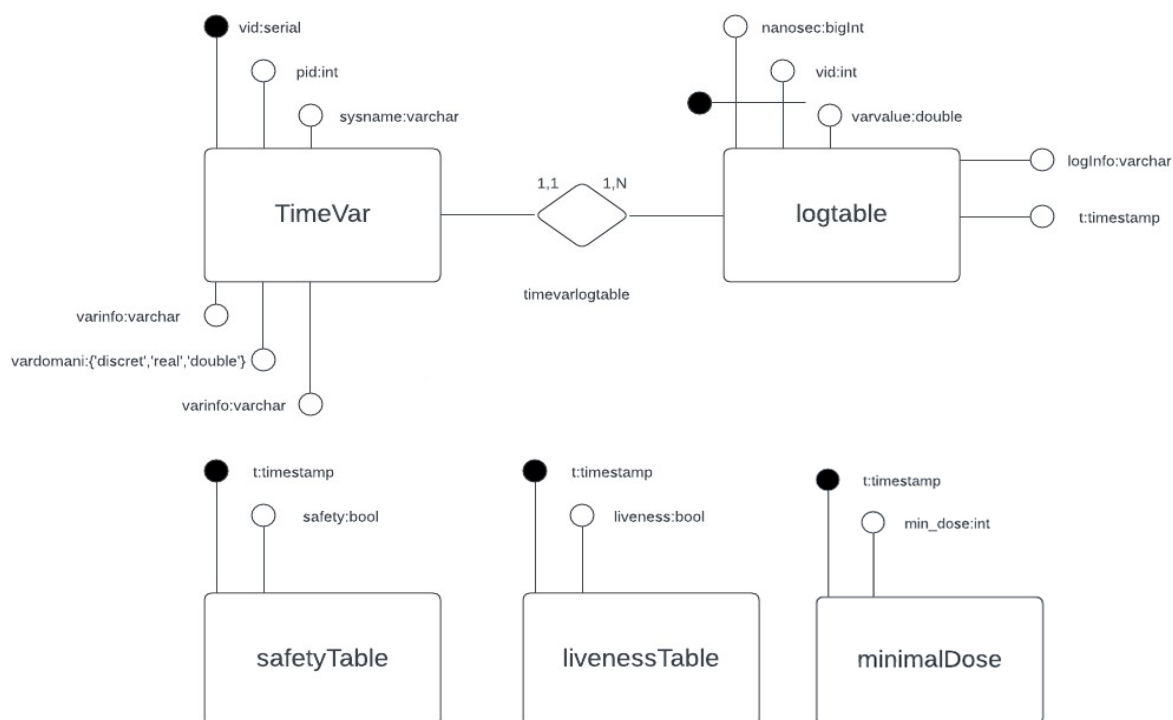
5.5 Records

t [PK] interval (3)	safety boolean	t [PK] interval (3)	liveness boolean	nanosec [PK] bigint	vid [PK] integer	varvalue double precision	loginfo character varying (200)	t interval (3)
00:00:06	false	01:00:00	false	0	1	127	Glucose value	00:00:00
00:00:12	false	02:00:00	false	0	2	54	Insulin value	00:00:00
00:00:18	false	03:00:00	false	0	3	0	Glucose in stomach	00:00:00
00:00:24	false	04:00:00	false	0	4	1.59	Endogenous glucose prod	00:00:00
00:00:30	false	05:00:00	false	0	5	-4.41	Insulin utilization	00:00:00
00:00:36	false	06:00:00	false	0	6	0	Renal excretion	00:00:00
00:00:42	false	07:00:00	false	0	7	49.902501	Insulin secretion	00:00:00
00:00:47.999	false	08:00:00	false	0	8	0	Food intake	00:00:00
00:00:53.999	false	09:00:00	false	0	9	1	test	00:00:00
00:01:00	false	10:00:00	false	0	10	0	Computed dose	00:00:00
00:01:05.999	false	11:00:00	false	1710784347903137205	1	127	Glucose value	00:00:06
00:01:11.999	false	12:00:00	false	1710784347903137205	2	54	Insulin value	00:00:06
00:01:18	false	13:00:00	false	1710784347903137205	3	0	Glucose in stomach	00:00:06
00:01:24	false	14:00:00	false	1710784347903137205	4	1.59	Endogenous glucose prod	00:00:06
				1710784347903137205	5	-4.41	Insulin utilization	00:00:06
				1710784347903137205	6	0	Renal excretion	00:00:06
				1710784347903137205	7	49.902501	Insulin secretion	00:00:06
				1710784348346621535	8	1	Food intake	00:00:06
				1710784348348577724	9	2	compute_release	00:00:06
				1710784348348577724	10	0	Computed dose	00:00:06

t [PK] interval (3)	min_dose integer
24:00:00	140

vid [PK] integer	pid integer	sysname character varying (200)	varname character varying (30)	vardomain domaintype	varinfo character varying (200)
1	501940	Patient	G	double	Plasma glucose concentration (mg/dL)
2	501940	Patient	I	double	Plasma insulin concentration (pmol/L)
3	501940	Patient	Qsto	double	Amount of glucose in the stomach (mg)
4	501940	Patient	EGP	double	endogenous glucose production
5	501940	Patient	Uid	double	Insulin-dependent utilization (mg/kg/min)
6	501940	Patient	E	double	Renal excretion (mg/kg/min)
7	501940	Patient	ISR	double	Insulin and c-peptide secretion (pmol/min)
8	501951	Environment	delta	discrete	Food intake
9	501950	Insulin pump	pump_state	discrete	State of insulin pump
10	501950	Insulin pump	comp_dose	double	dose (pmol/L)

5.5 ER Database



6. Server Redis

Il server Redis utilizzato nel sistema svolge anche la funzione di database di tipo chiave-valore e supporta diverse strutture dati come stringhe, liste, insiemi e altri. Nel sistema dell'Insulin Pump il server è utilizzato per scrivere e leggere valori tramite stream di dati.

L'interazione tra i modelli ed il server Redis può essere riassunta in 4 fasi differenti:

- **Connessione** I modelli si connettono al server Redis in esecuzione sulla porta predefinita sull'host "localhost" utilizzando la funzione `redisConnect()`.
- **Operazioni sui dati** Vengono eseguite varie operazioni sui dati nel server Redis, come eliminare i flussi esistenti (`DEL`), creare nuovi flussi (`XADD`) e leggere i messaggi da un flusso (`XREADGROUP`).
- **Scrittura e lettura dei dati** I dati relativi al paziente e alle varie grandezze fisiologiche vengono scritti e letti dal server Redis utilizzando i comandi di scrittura (`XADD`) e lettura (`XREADGROUP`). Ad esempio, i valori relativi al glucosio vengono scritti dal paziente in `PATIENT_TO_PUMP` e letti dalla pompa dallo stesso flusso. L'insulina invece viene scritta dalla pompa in `PUMP_TO_PATIENT` e letta dal paziente dallo stesso flusso. L'environment scrive nel flusso `ENV_STREAM` e il paziente legge dallo stesso flusso.
- **Elaborazione dei dati** Dopo aver letto i dati dal server Redis, vengono eseguite operazioni di elaborazione su di essi per calcolare nuovi valori fisiologici. Questi nuovi valori vengono quindi scritti nuovamente nel server Redis per essere eventualmente letti da altri componenti del sistema.