

Progetto: "GreenPass"  
Reti Di Calcolatori

Simone Perrotta  
Mat:0124002350

Flavio Ruggiero  
Mat:0124002342

Anno accademico 2022-2023



---

# Indice

<b>1</b>	<b>Descrizione del progetto</b>	<b>3</b>
<b>2</b>	<b>Descrizione e schemi dell'architettura</b>	<b>3</b>
<b>3</b>	<b>Descrizione e schemi del protocollo applicazione</b>	<b>4</b>
<b>4</b>	<b>Dettagli implementativi</b>	<b>4</b>
4.1	ClientUtente . . . . .	4
4.2	ClientS . . . . .	4
4.3	ClientT . . . . .	4
4.4	Centro Vaccinale . . . . .	4
4.5	ServerV . . . . .	5
4.6	ServerG . . . . .	5
4.7	Centro Vaccinale . . . . .	5
<b>5</b>	<b>Manuale Utente</b>	<b>6</b>
5.1	Istruzioni per la compilazione . . . . .	6
5.2	Istruzioni per l'esecuzione . . . . .	6
<b>6</b>	<b>Simulazione dell'applicazione</b>	<b>8</b>
<b>7</b>	<b>Glossario</b>	<b>9</b>

# 1 Descrizione del progetto

La Certificazione Verde (Green Pass) è un documento che attesta che una persona ha ricevuto una o più dosi di vaccino contro il COVID-19, o che ha recuperato dal virus, o che ha fatto un test negativo nelle ultime 48 ore.

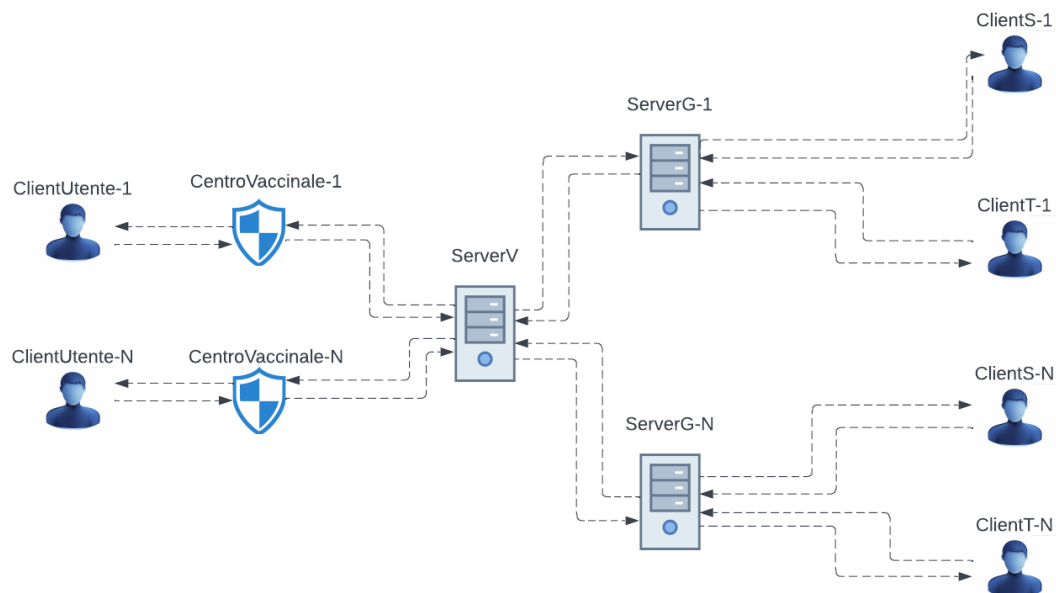
Essa è stata introdotta per facilitare la libera circolazione all'interno dell'Unione Europea, garantendo ai cittadini la possibilità di spostarsi senza restrizioni all'interno dei Paesi membri durante la pandemia. La certificazione è stata rilasciata dal Ministero della Salute italiano a partire dal 1 luglio 2021.

Il Green Pass è un documento digitale che può essere scaricato attraverso il sito web ufficiale del Ministero della Salute oppure attraverso l'applicazione Immuni. Il documento contiene un codice QR univoco che può essere scansionato per verificare la validità della certificazione.

La Certificazione Verde è valida in tutti i Paesi dell'Unione Europea ed è accettata come prova di vaccinazione o di negatività al COVID-19 per l'accesso a luoghi pubblici e privati come ristoranti, bar, teatri, cinema, stadi, e per i viaggi all'estero.

È importante notare che la Certificazione Verde non è obbligatoria, ma è altamente raccomandata per facilitare la circolazione durante la pandemia. Inoltre, il documento non è un passaporto sanitario, non contiene informazioni personali e non può essere utilizzato per tracciare o monitorare le attività delle persone. La traccia del progetto riguarda l'implementazione di un servizio di gestione del Green Pass utilizzando i socket per la comunicazione tra processi. Il progetto è stato implementato in linguaggio C su piattaforma Linux.

# 2 Descrizione e schemi dell'architettura



L'architettura più adatta al nostro progetto è un'architettura di tipo client/server multilivello poichè c'è la necessita che diversi client e server siano in esecuzione sulla stessa macchina o su diverse macchine. Le entità che compongono l'architettura sono le seguenti:

- ClientUtente: comunica solo con un centro vaccinale per richiedere di vaccinarsi;
- ClientS: comunica solo con un ServerG per verificare che un Green Pass sia valido;
- ClientT: comunica solo con un ServerG per modificare lo stato di un Green Pass;
- CentroVaccinale: comunica sia con più ClientUtente che con il ServerV per inviare il codice ricevuto dal ClientUtente e inviarlo al ServerV insieme alla data di validità del Green Pass;
- ServerG: comunica sia con più ClientS sia con più ClientT che con il ServerV facendo da tramite per applicare le direttive del server e inviare gli esiti ai client;
- ServerV: comunica sia con più Centri Vaccinali che con il ServerG per modificare lo stato di un Green Pass secondo le direttive del ClientT;

Nel file WrapperFunction.c e WrapperFunction.h ci saranno tutte le funzioni "wrappate" per la creazione, eliminazione e gestione dei socket per la comunicazione tra client e server.

---

### 3 Descrizione e schemi del protocollo applicazione

I socket in C sono un meccanismo di comunicazione che consente a processi in esecuzione su diversi computer di comunicare tra di loro attraverso una rete. In C, i socket sono implementati attraverso la libreria di sistema "socket.h". La libreria fornisce funzioni per creare, configurare e utilizzare i socket.

La specifica utilizzata dai socket del nostro progetto è TCP, inoltre il nostro progetto è basato sulla comunicazione Client-Server sulle funzioni **full\_write** e **full\_read** che servono per effettuare delle operazioni di lettura e scrittura. Tali funzioni garantiscono che tutti i dati vengano scritti o letti sul socket ovvero che tutti i dati vengano inviati al server.

Essendo necessaria una connessione socket di tipo TCP, abbiamo scelto come indirizzo il classico localhost (127.0.0.1).

Per usare lo stesso indirizzo, abbiamo scelto di dividere le porte per le varie entità in questo modo:

- 10000: connessione col Centro Vaccinale;
- 20000: connessione col ServerV;
- 30000: connessione col ServerG;

### 4 Dettagli implementativi

Nel nostro progetto sono stati individuati diversi attori che si comportano da Client:

- ClientUtente;
- ClientS;
- ClientT;
- Centro Vaccinale;

Nel nostro progetto sono stati individuati diversi attori che si comportano da Server:

- ServerV;
- ServerG;
- CentroVaccinale;

#### 4.1 ClientUtente

**ClientUtente** è un client che si connette al centro vaccinale per effettuare la richiesta di vaccinazione. Questo avviene inviando al centro vaccinale il codice della tessera sanitaria (16 caratteri) dell'utente in questione. Dopodiché il centro vaccinale comunicherà al client utente la corretta somministrazione del vaccino e il rilascio del Green Pass.

#### 4.2 ClientS

**ClientS** è un client che verifica la validità del Green Pass. Per effettuare il controllo bisogna passare un codice di tessera sanitaria, che sarà inviato al ServerG, che comunicando col ServerV restituirà al ClientS la validità o invalidità del Green Pass associato a quel codice. Se valido verrà mostrata anche la data di scadenza della Certificazione Verde.

#### 4.3 ClientT

**ClientT** è un client che modifica lo stato di validità di un Green Pass. Questo avviene inviando al ServerG un codice di tessera sanitaria e un valore booleano che indicherà la validazione o l'invalidazione del Green Pass associato al codice.

#### 4.4 Centro Vaccinale

Il **centro vaccinale** si comporta da client nel momento in cui invia al ServerV la richiesta di vaccinazione per il codice di tessera sanitaria passato. Il ServerV risponderà al centro vaccinale con un messaggio di successo o fallimento relativo alla possibilità di vaccinare la persona collegata al codice di tessera sanitaria. Il centro vaccinale comunicherà di conseguenza la risposta al ClientUtente.

---

## 4.5 ServerV

Il ServerV comunica col centro vaccinale per ricavare il Green Pass e aggiungere al suo file di riferimento il nuovo Green Pass associato al codice fiscale passato.

Il ServerV comunica anche con il ServerG per eseguire richieste effettuate da ClientT e ClientS.

## 4.6 ServerG

Il ServerG comunica col ServerV e riceve richieste da parte di ClientT e ClientS, nello specifico:

- il ClientT richiede al ServerG di modificare lo stato di validità di un Green Pass associato al codice fiscale passato come parametro;
- il ClientS richiede al ServerG di visualizzare lo stato di un Green Pass associato al codice fiscale passato come parametro;

## 4.7 Centro Vaccinale

Il centro vaccinale comunica con il ServerV per effettuare la somministrazione del vaccino se possibile. E' possibile quando:

- Il ClientUtente in questione non è mai stato vaccinato;
- Il ClientUtente può effettuare l'ulteriore dose di vaccino poichè sono passati più di 6 mesi;

Il centro vaccinale inoltre comunica con il ClientUtente per inviargli il messaggio di successo o fallimento relativo alla somministrazione del vaccino. Per la corretta implementazione della comunicazione tra client e server sono state utilizzate le seguenti strutture:

```
typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    char dataScadenzaGreenPass[LUNGHEZZA_DATA];
    unsigned short int requestResult;
} centroVaccinaleRispondeAClientUtente;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    char dataScadenzaGreenPass[LUNGHEZZA_DATA];
} centroVaccinaleRichiedeAServerV;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    char dataScadenzaGreenPass[LUNGHEZZA_DATA];
    unsigned short int requestResult;
} serverVRispondeACentroVaccinale;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    unsigned short int requestResult;
} serverVRispondeAServerGVersoClientS;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    unsigned short int updateValue;
} serverGRichiedeAServerV_PerContoDiClientT;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    unsigned short int updateResult;
} serverVRispondeAServerGVersoClientT;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    unsigned short int requestResult;
} serverGRispondeAClientS;
```

```
typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    unsigned short int updateValue;
} clientTRichiedeAServerG;

typedef struct {
    char codiceTesseraSanitaria[LUNGHEZZA_CODICE_TESSERA_SANITARIA];
    unsigned short int updateResult;
} serverGRispondeAClientT;
```

## 5 Manuale Utente

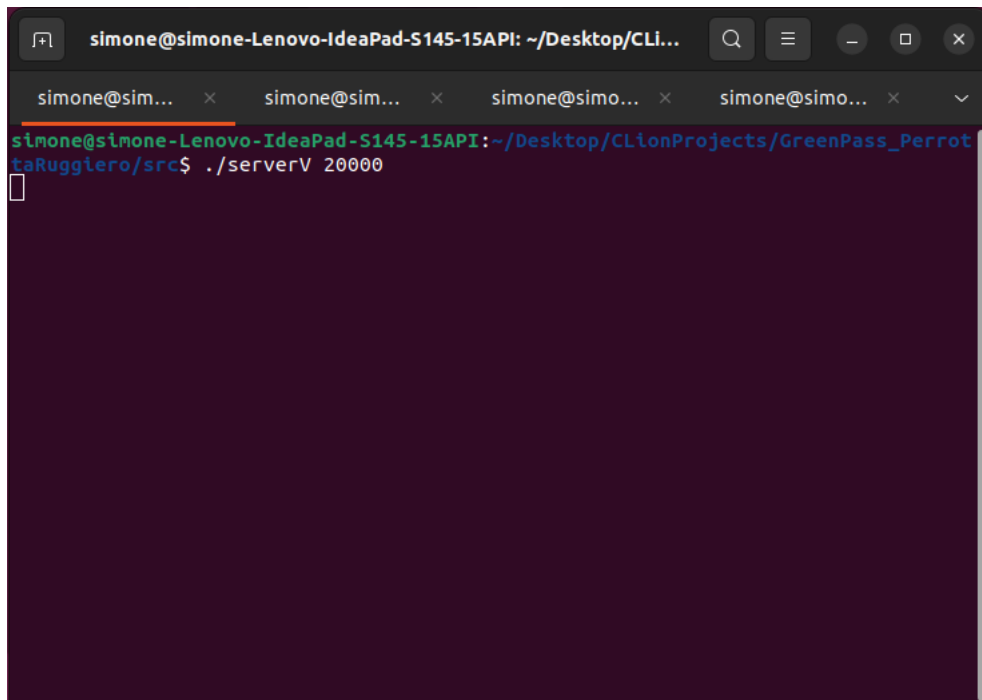
### 5.1 Istruzioni per la compilazione

Per la corretta compilazione è stato creato un makefile che contiene le istruzioni di compilazione per i vari file. Quindi basterà eseguire il comando make nella directory src.

### 5.2 Istruzioni per l'esecuzione

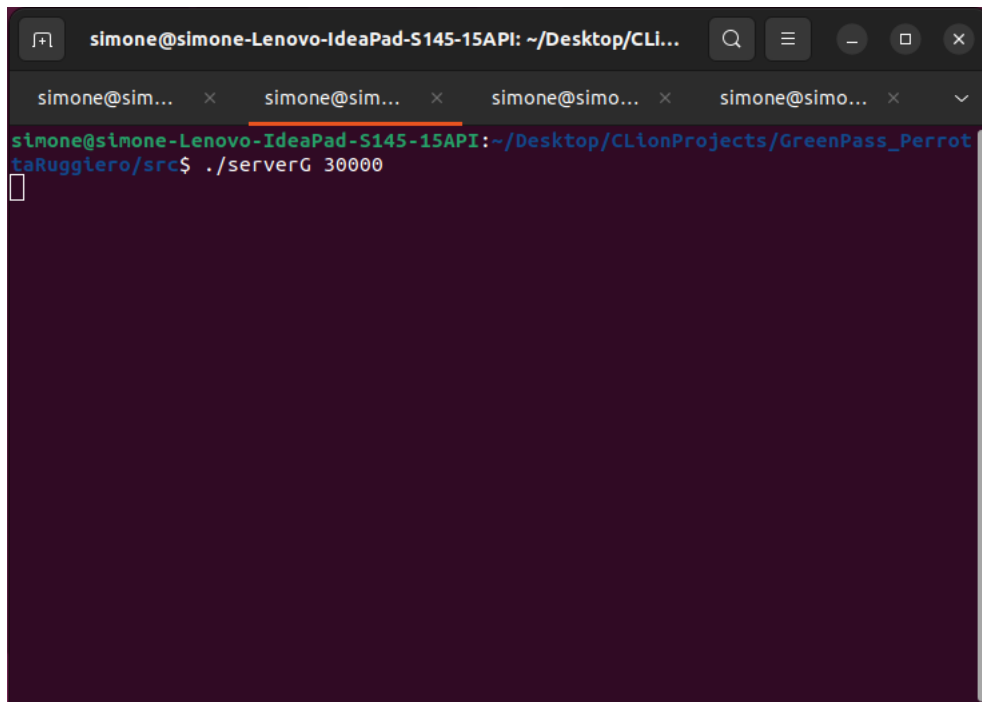
Per la corretta esecuzione del programma è necessario che vengano messi in ascolto i vari server:

- ServerV



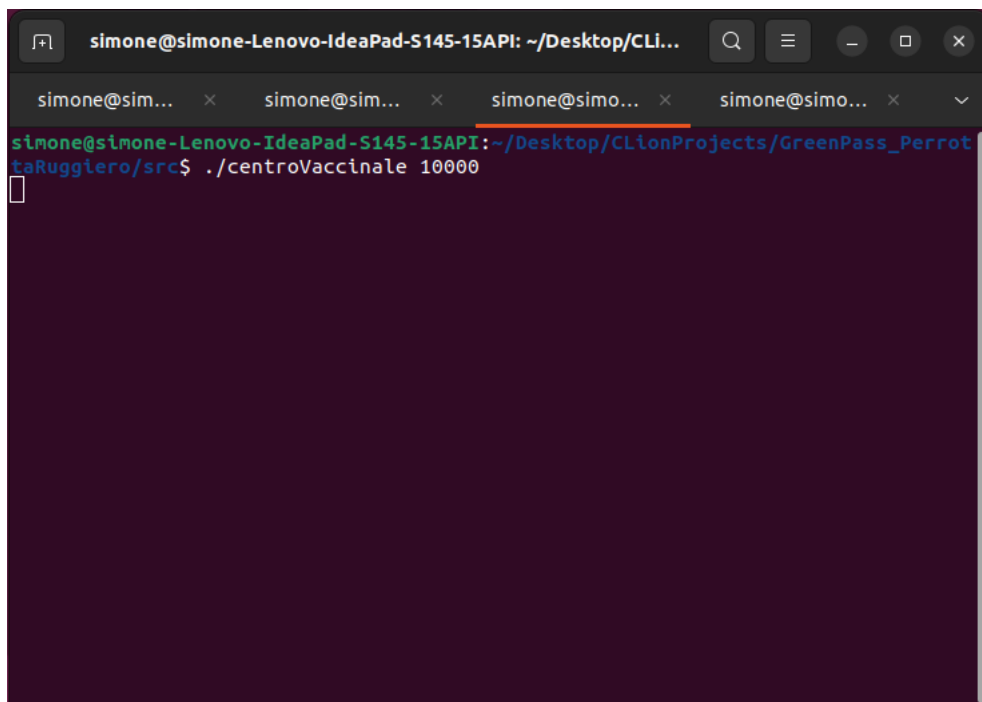
The screenshot shows a terminal window with a dark background. The title bar at the top reads "simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLI...". Below the title bar, there are four tabs, each labeled "simone@sim...". The main terminal area shows the prompt "simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLionProjects/GreenPass\_PerrottaRuggiero/src\$" followed by the command "./serverV 20000". A cursor is visible on the line following the command.

- ServerG



```
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLI...
simone@sim... x simone@sim... x simone@simo... x simone@simo... x
simone@simone-Lenovo-IdeaPad-S145-15API:~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$ ./serverG 30000
```

- CentroVaccinale

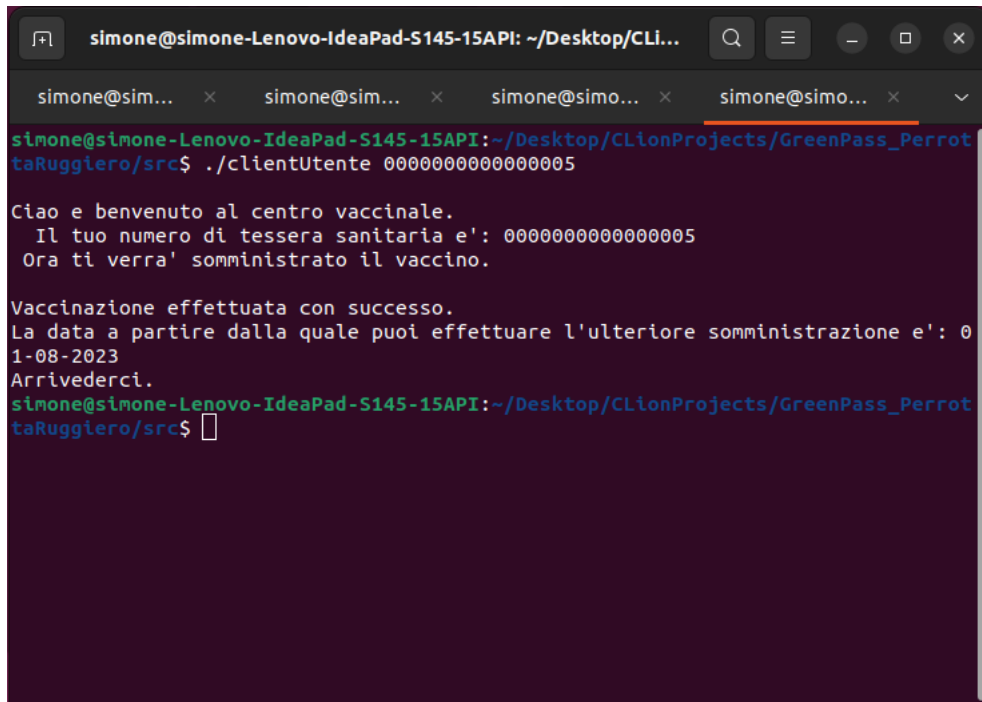


```
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLI...
simone@sim... x simone@sim... x simone@simo... x simone@simo... x
simone@simone-Lenovo-IdeaPad-S145-15API:~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$ ./centroVaccinale 10000
```

## 6 Simulazione dell'applicazione

Una volta eseguite le operazioni preliminari possiamo finalmente utilizzare il programma per la gestione del servizio della Certificazione Verde.

Nella seguente immagine possiamo notare un utente che fa la richiesta di vaccinazione e il server gli invia la conferma della vaccinazione e la data di scadenza del Green Pass:

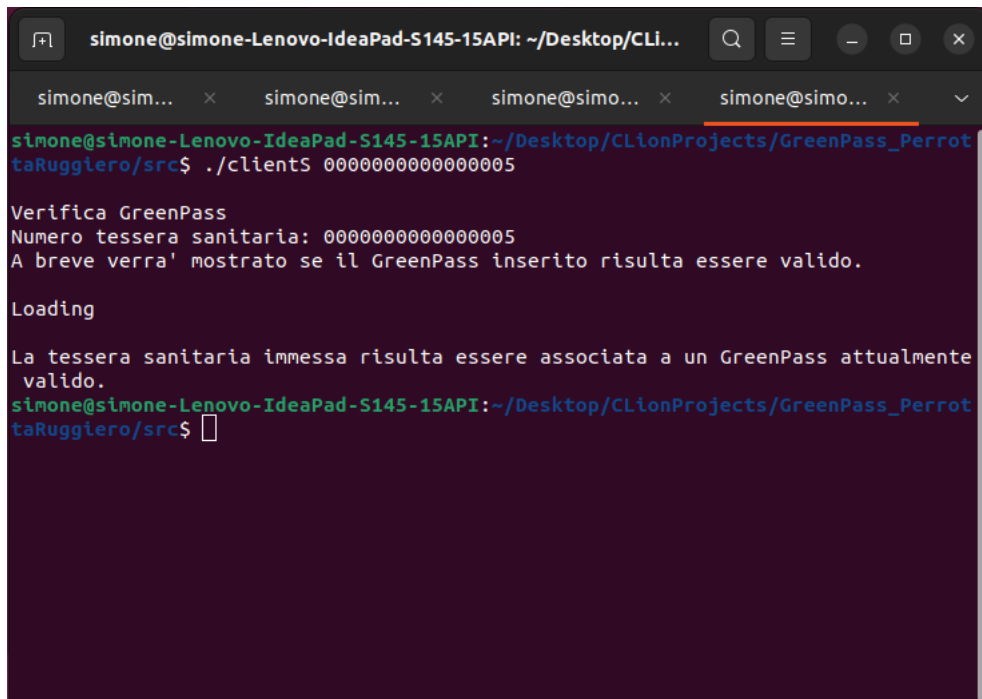


```
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLi...
simone@sim... x  simone@sim... x  simone@simo... x  simone@simo... x
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$ ./clientUtente 0000000000000005

Ciao e benvenuto al centro vaccinale.
  Il tuo numero di tessera sanitaria e': 0000000000000005
  Ora ti verra' somministrato il vaccino.

Vaccinazione effettuata con successo.
La data a partire dalla quale puoi effettuare l'ulteriore somministrazione e': 0
1-08-2023
Arrivederci.
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$
```

Nella seguente immagine possiamo notare un utente che verifica la validità del suo Green Pass tramite richiesta al server.



```
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLi...
simone@sim... x  simone@sim... x  simone@simo... x  simone@simo... x
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$ ./clientS 0000000000000005

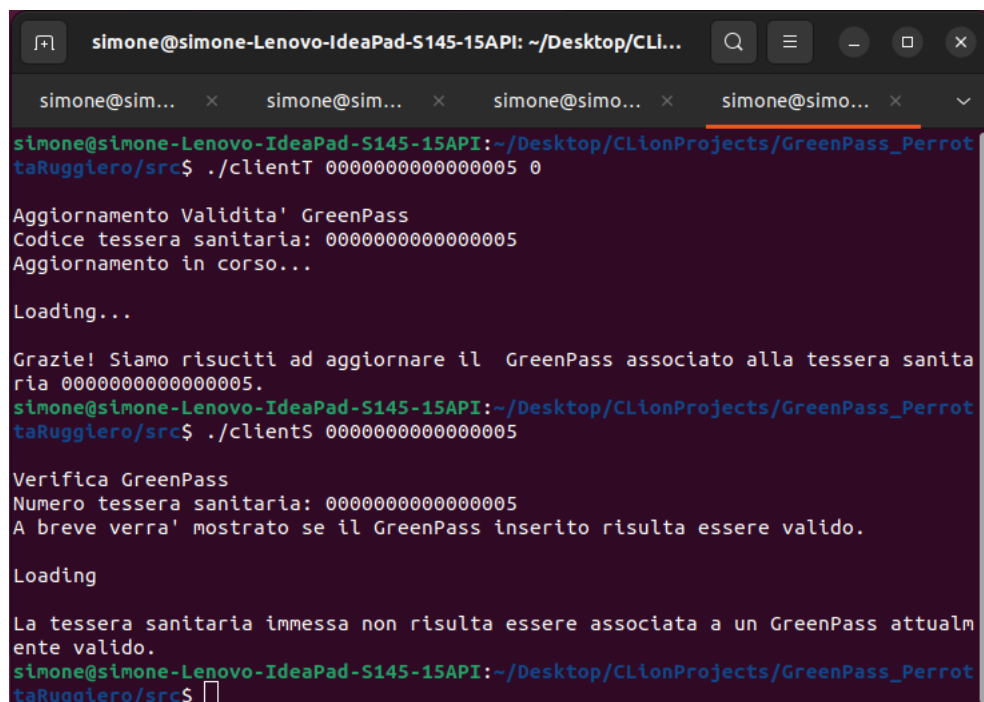
Verifica GreenPass
Numero tessera sanitaria: 0000000000000005
A breve verra' mostrato se il GreenPass inserito risulta essere valido.

Loading

La tessera sanitaria immessa risulta essere associata a un GreenPass attualmente
valido.
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$
```



Nella seguente immagine possiamo notare un utente che richiede al server di modificare lo stato del suo Green Pass e il server accetta la richiesta e aggiorna lo stato.



```
simone@simone-Lenovo-IdeaPad-S145-15API: ~/Desktop/CLi...
simone@sim... x  simone@sim... x  simone@simo... x  simone@simo... x
simone@simone-Lenovo-IdeaPad-S145-15API:~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$ ./clientT 0000000000000005 0

Aggiornamento Validita' GreenPass
Codice tessera sanitaria: 0000000000000005
Aggiornamento in corso...

Loading...

Grazie! Siamo riusciti ad aggiornare il GreenPass associato alla tessera sanita
ria 0000000000000005.
simone@simone-Lenovo-IdeaPad-S145-15API:~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$ ./clientS 0000000000000005

Verifica GreenPass
Numero tessera sanitaria: 0000000000000005
A breve verra' mostrato se il GreenPass inserito risulta essere valido.

Loading

La tessera sanitaria immessa non risulta essere associata a un GreenPass attualm
ente valido.
simone@simone-Lenovo-IdeaPad-S145-15API:~/Desktop/CLionProjects/GreenPass_Perrot
taRuggiero/src$
```

## 7 Glossario

Termine	Definizione	Sinonimi	Omonimi
Codice Fiscale Sanitaria	codice univoco associato a ogni persona fisica	codice tessera sanitaria	—