

Manuale Tecnico

Università degli Studi dell'Insubria
Dipartimento di Scienze Teoriche e Applicate

LABORATORIO INTERDISCIPLINARE B
PROGETTAZIONE DI CENTRI VACCINALI



Maven



Università degli Studi dell'Insubria
Dipartimento di Scienze Teoriche e Applicate

LABORATORIO INTERDISCIPLINARE B
PROGETTAZIONE DI CENTRI VACCINALI

MANUALE TECNICO

REDATTO DA:

Rosario Catalfamo, matricola 744951
Andrea Pigliafreddo, matricola 746578
Alessandro Barone, matricola 744453

SOMMARIO

ANALISI DEI REQUISITI.....	7
ANALISI DEI REQUISITI.....	7
ANALISI DEI REQUISITI DELLA BASE DI DATI.....	7
REQUISITI MINIMI PER L'INSTALLAZIONE ED UTILIZZO DELL'APPLICAZIONE.....	7
ANALISI DEI REQUISITI DELLA BASE DI DATI.....	8
LATO SERVER.....	8
LATO CLIENT.....	9
INSTALLAZIONE DEL DATABASE.....	10
INSTALLAZIONE DEL CLIENT.....	10
Avvio del Client.....	11
Fase 1: Compilazione.....	11
Fase 2: Avvio della connessione.....	11
Fase 3: Avvio del programma.....	11
ANALISI DEL SISTEMA.....	12
USE CASE.....	12
AUTENTICAZIONE.....	12
AZIONI CITTADINO.....	13
AZIONI OPERATORE SANITARIO.....	13
AZIONI USER.....	14
SEQUENCE DIAGRAM.....	15
CREAZIONE CENTRO.....	15
ESEGUI VACCINAZIONE.....	16
LOGIN.....	17
PRENOTA VACCINAZIONE.....	18
REGISTRA CENTRO.....	19
RICERCA CENTRO.....	19
ACTIVITY DIAGRAM.....	20
CREAZIONE CENTRO.....	20
EVENTI AVVERSI.....	21
PRENOTAZIONE VACCINO.....	22
REGISTRAZIONE CITTADINO.....	23
REGISTRAZIONE DOTTORE.....	24
RICERCA CENTRO.....	25

ESEGUI VACCINAZIONE.....	26
STATE DIAGRAM	27
REGISTRAZIONE CITTADINO	27
REGISTRAZIONE MEDICO	27
CLASS DIAGRAM.....	28
GLOBAL CLASS DIAGRAM	29
GLOBAL CLASS DIAGRAM: ClientCV (Focus)	30
GLOBAL CLASS DIAGRAM: Classi Oggetto (Focus)	31
GLOBAL CLASS DIAGRAM: ServerCV (Focus)	32
DESCRIZIONE INTERFACCIA GRAFICA.....	33
HOME PAGE	33
LOGIN CITTADINO	34
LOGIN OPERATORE SANITARIO.....	34
REGISTRAZIONE CITTADINO	35
REGISTRAZIONE OPERATORE SANITARIO.....	36
REGISTRA CENTRO.....	37
REGISTRAZIONE VACCINATO	38
SCHERMATA OS.....	39
SCHERMATA CITTADINO.....	40
OPERAZIONI CITTADINO	41
STATISTICHE	42
EVENTI AVVERSI	42
STRUTTURE DATI IMPLEMENTATE.....	43
PROGETTAZIONE	44
SCELTE STRATEGICHE	44
DAO (Data Access Object).....	44
Singleton	44
Factory Layout.....	44
Observer	44
MVC (Model-View-Controller)	44
Login/connessione al database tramite comando Apache ANT	44
Salvataggio sul database solo di password criptate.....	44
PROGETTAZIONE LAVORATIVA	45
PROGETTAZIONE DELLA BASE DI DATI	47

RACCOLTA ED ANALISI DEI REQUISITI	47
SPECIFICHE SUI DATI IN LINGUAGGIO NATURALE	47
CARICO DI LAVORO.....	48
SCHEMA CONCETTUALE	51
DOCUMENTAZIONE	52
PROGETTAZIONE LOGICA	53
TRADUZIONE	53
NORMALIZZAZIONE	54
PROGETTAZIONE PRATICA.....	54
CREAZIONE DEL DATABASE.....	54
CREAZIONE TABELLE.....	54
OPERAZIONI DI INSERT	56
QUERY	57
IMPLEMENTAZIONE	59
FUNZIONALITA' AGGIUNTIVE	59
PROBLEMI RISCONTRATI	59

ANALISI DEI REQUISITI

ANALISI DEI REQUISITI

L'analisi dei requisiti è la prima fase di qualsiasi processo di sviluppo, in questa fase si definiscono tutti i requisiti funzionali e non funzionali relativi al sistema da costruire. È importante dedicare un tempo ragionevole per comprendere appieno le funzionalità richieste. La completezza della definizione dei requisiti può comportare notevoli vantaggi nelle fasi successive.

ANALISI DEI REQUISITI DELLA BASE DI DATI

La fase della raccolta dei requisiti consiste nell'individuazione delle informazioni da memorizzare all'interno della base di dati. I requisiti descritti nel paragrafo successivo sono stati determinati a partire dallo studio dei documenti di specifica disponibili, individuando le informazioni fondamentali che la base di dati dovrà contenere.

REQUISITI MINIMI PER L'INSTALLAZIONE ED UTILIZZO DELL'APPLICAZIONE

Per poter procedere all'utilizzo del DBMS è necessario che i dispositivi rispettino delle caratteristiche minime:

Per l'utilizzo dell'applicazione è necessario dedicare un dispositivo su cui venga installato il DBMS, sullo stesso potrà comunque girare anche più di un'istanza dell'applicazione. I Client si connetteranno ad esso per il recupero dei dati.

ANALISI DEI REQUISITI DELLA BASE DI DATI

LATO SERVER

Per l'utilizzo del software è necessario procedere all'installazione in rete di un dispositivo che offrirà un servizio di Database Relazionale. La scelta DB per soluzione committente è PostgreSQL

I requisiti minimi HW per il dispositivo sono:

1 GHz processor
2 GB of RAM
512 MB of HDD

I requisiti SW per il dispositivo sono:

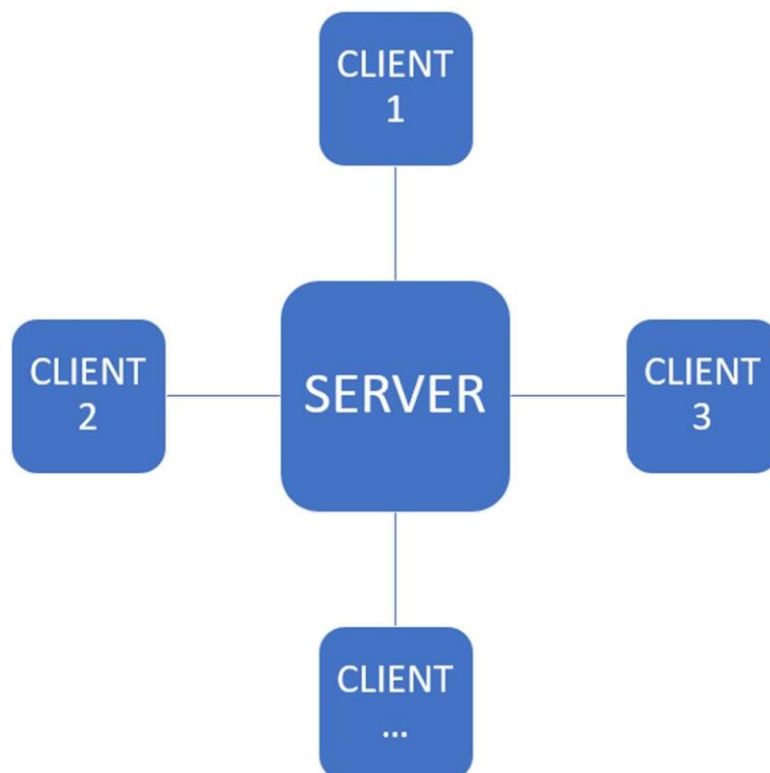
Windows: Windows 7 o superiore (obbligatorio che l'architettura sia a 64 bit)

Linux: Debian, Ubuntu, RedHat, Fedora, CentOS

MacOs: OSX 10.7 o superiore

Non è strettamente necessario che il S.O. sia di tipo server.

Per fare in modo che gli altri dispositivi possano connettersi e utilizzare il servizio offerto, è necessario che venga sbloccata lato firewall la porta del DB, che per standard è la 5432.



LATO CLIENT

Come requisito di sistema per il funzionamento del programma è necessario che sia installata la JRE (Java Runtime Environment) versione minima 1.8.xx (in alternativa è possibile utilizzare anche la JDK (Java Development Kit) versione minima 11.0.xx).

Ottenibili dal sito Oracle: <https://www.java.com/it/download/>

Per il funzionamento è inoltre necessario l'utilizzo delle librerie Maven.

Ottenibili dal sito: <https://maven.apache.org/download.cgi>

Per l'utilizzo della piattaforma è necessario che il/i client siano compatibili con Java.

I requisiti sono (fonte oracle.com):

- *Windows Vista Sp2 o superiore oppure Windows Server 2008 R2 SP1 o superiore*
- *RAM: 128 MB (a questi vanno aggiunti i costi in memoria dell'applicazione circa 100MB)*
- *Spazio sul disco: 150 MB*

INSTALLAZIONE DEL DATABASE

Una volta installato il DBMS occorre procedere alla creazione della struttura per la collezione dei dati.

La scelta del Nome del DB, dell'username e della password viene lasciata libera, è necessario ricordarsi di annotarla poiché sarà utile per la connessione dei Client.

La struttura di base si compone di cinque tabelle:

Centrivaccinali: dove vengono registrate le strutture per la somministrazione dei Vaccini;

Cittadini_Registrati: dove risiede l'anagrafica dei cittadini registrati;

Eventi_Avversi: dove vengono registrati tutti i casi di eventi avversi in maniera anonima;

Vaccinati_NomeCentro: dove vengono registrate le vaccinazioni dei Cittadini ("NomeCentro" cambia dinamicamente con il centro di vaccinazione).

Per procedere alla creazione delle tabelle del database è necessario eseguire il programma, in

esecuzione verranno richiesti i dati per accedere al Database. Successivamente verranno create

dinamicamente delle tabelle per ogni centro vaccinale.

INSTALLAZIONE DEL CLIENT

- Procedere all'installazione di Oracle Java
- Estrarre il pacchetto delle Librerie Maven e posizionarlo in una directory accessibile (suggerimento: posizionarlo nel root del disco fisso es: c:\Maven\)
- Configurare le Variabili Ambientali:

JAVA_HOME	Directory di JAVA (Es: C:\Program Files\Java\jre1.8.0_291)
PATH	...; %JAVA_HOME%\bin ...; %ANT_HOME%\bin

N.B. dopo la configurazione si consiglia il riavvio della postazione

AVVIO DEL CLIENT

L'esecuzione dell'applicazione avviene tramite Terminale partendo dalla directory dove risiede il programma.

FASE 1: COMPILAZIONE

Assicurati di aver installato Maven sul tuo sistema.

Se non lo hai installato, puoi scaricarlo dal sito Web di Apache Maven (<https://maven.apache.org/download.cgi>) o installarlo utilizzando un gestore di pacchetti come Homebrew (per Mac) o Chocolatey (per Finestre).

Passa alla directory principale del tuo progetto Maven. Questa è la directory che contiene il file pom.xml.

Esegui il comando seguente per creare il tuo progetto:

```
mvn clean install
```

Questo compilerà il tuo codice, eseguirà eventuali test e creerà un file JAR.

FASE 2: AVVIO DEL PROGRAMMA

Per eseguire il tuo progetto, usare il seguente comando:

```
mvn exec:java -Dexec.mainClass="ClientCV.CentriVaccinali.CentriVaccinali.java"
```

Sostituisci com.example.Main con il nome completo della tua classe principale.

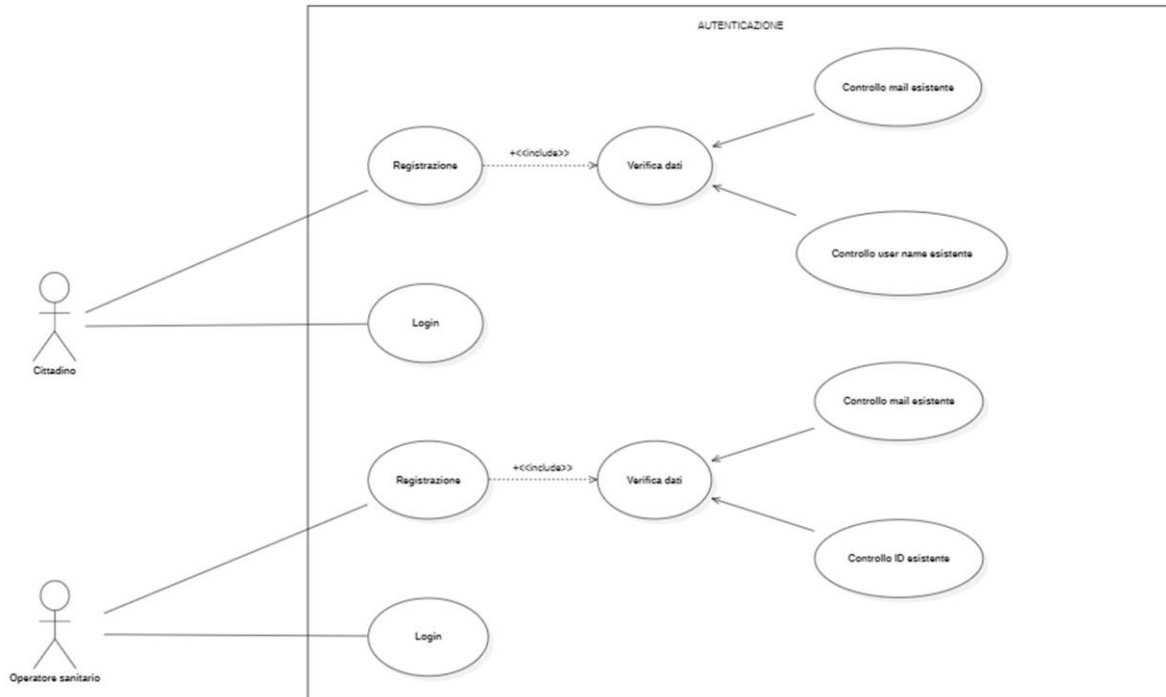
Se hai problemi, puoi consultare la documentazione ufficiale su

<https://maven.apache.org/guides/getting-started/>.

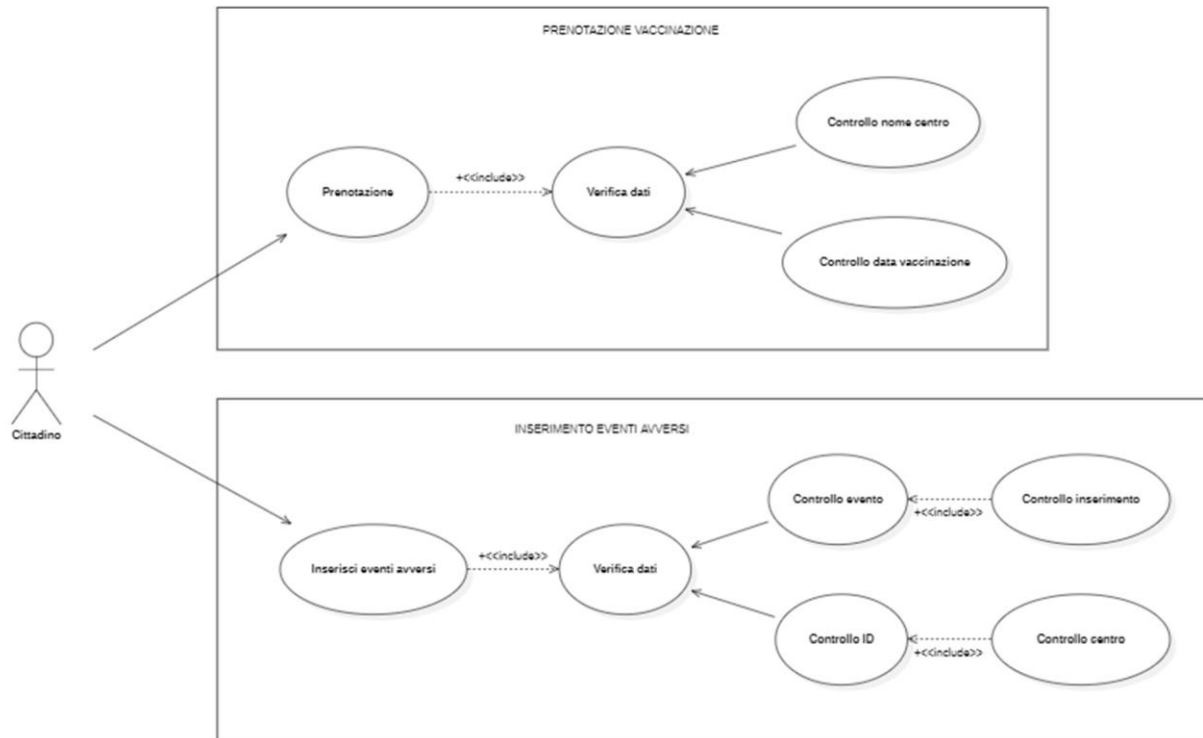
ANALISI DEL SISTEMA

USE CASE

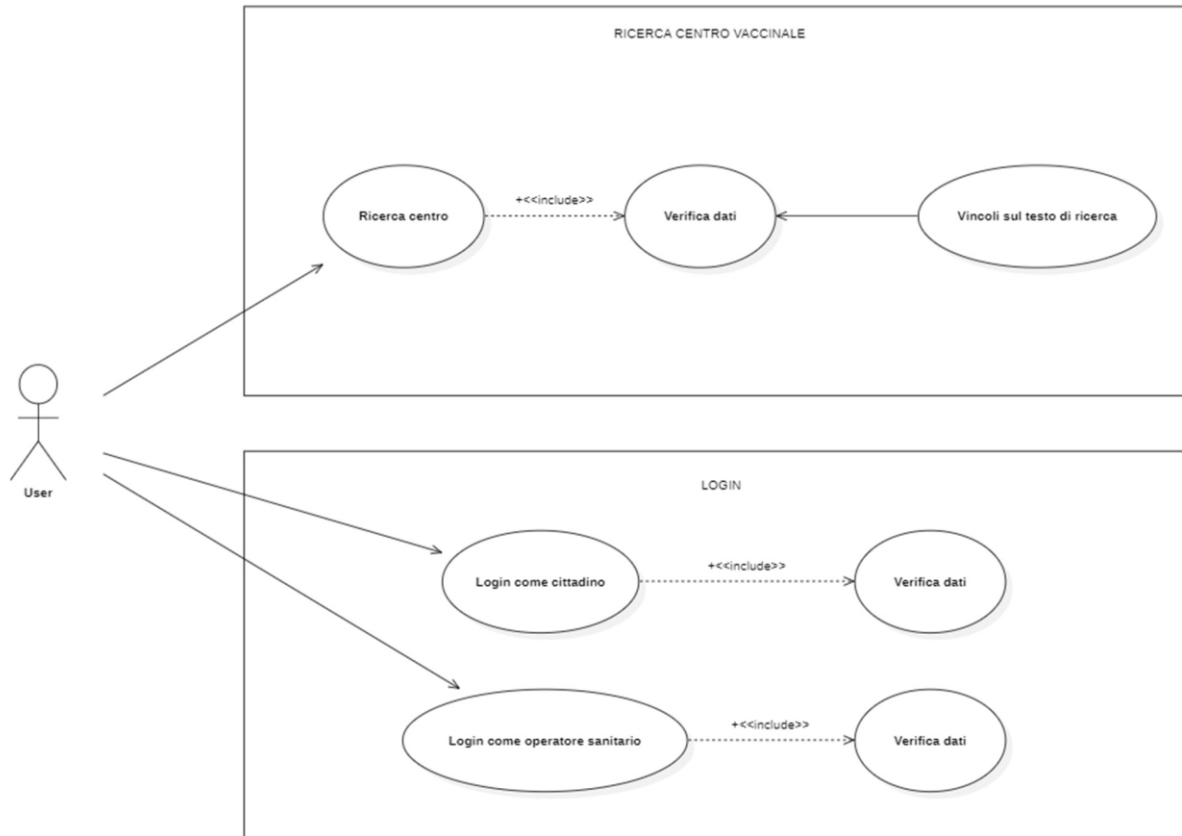
AUTENTICAZIONE



AZIONI CITTADINO

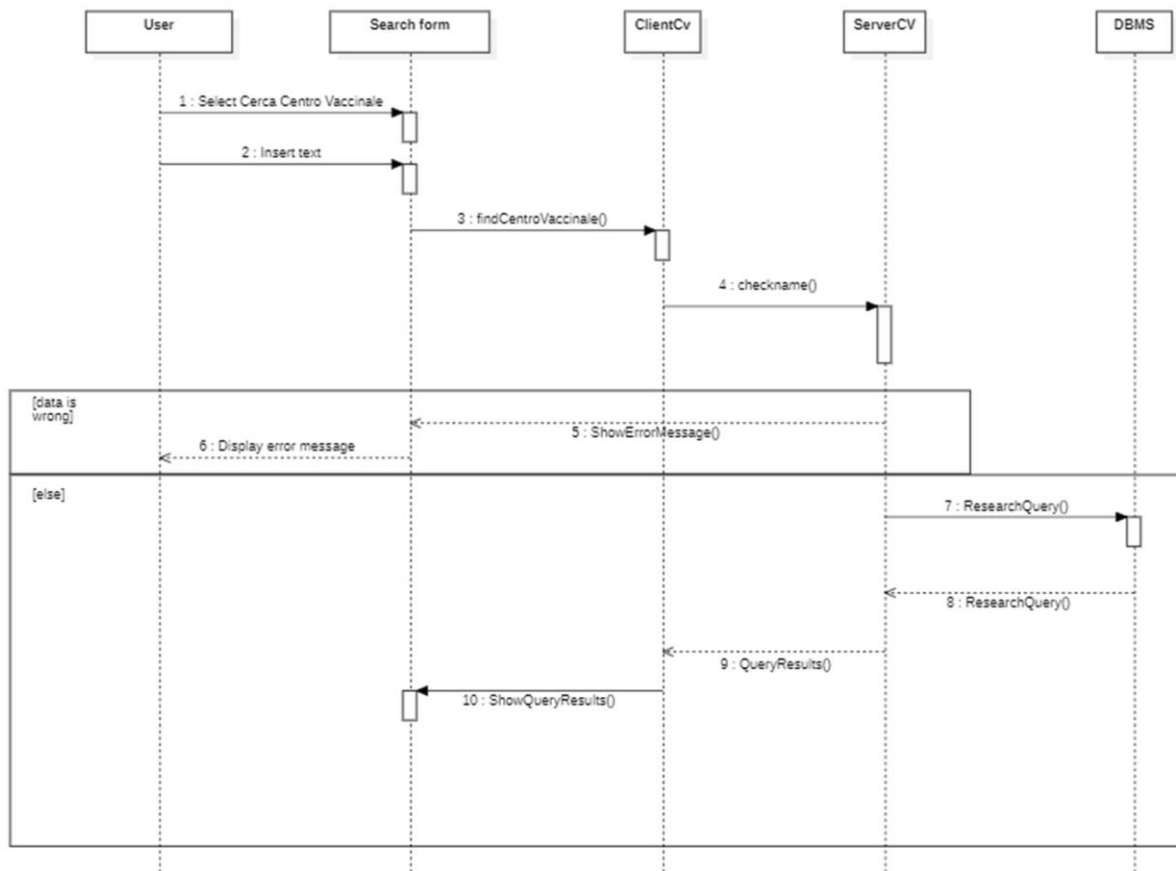


AZIONI USER

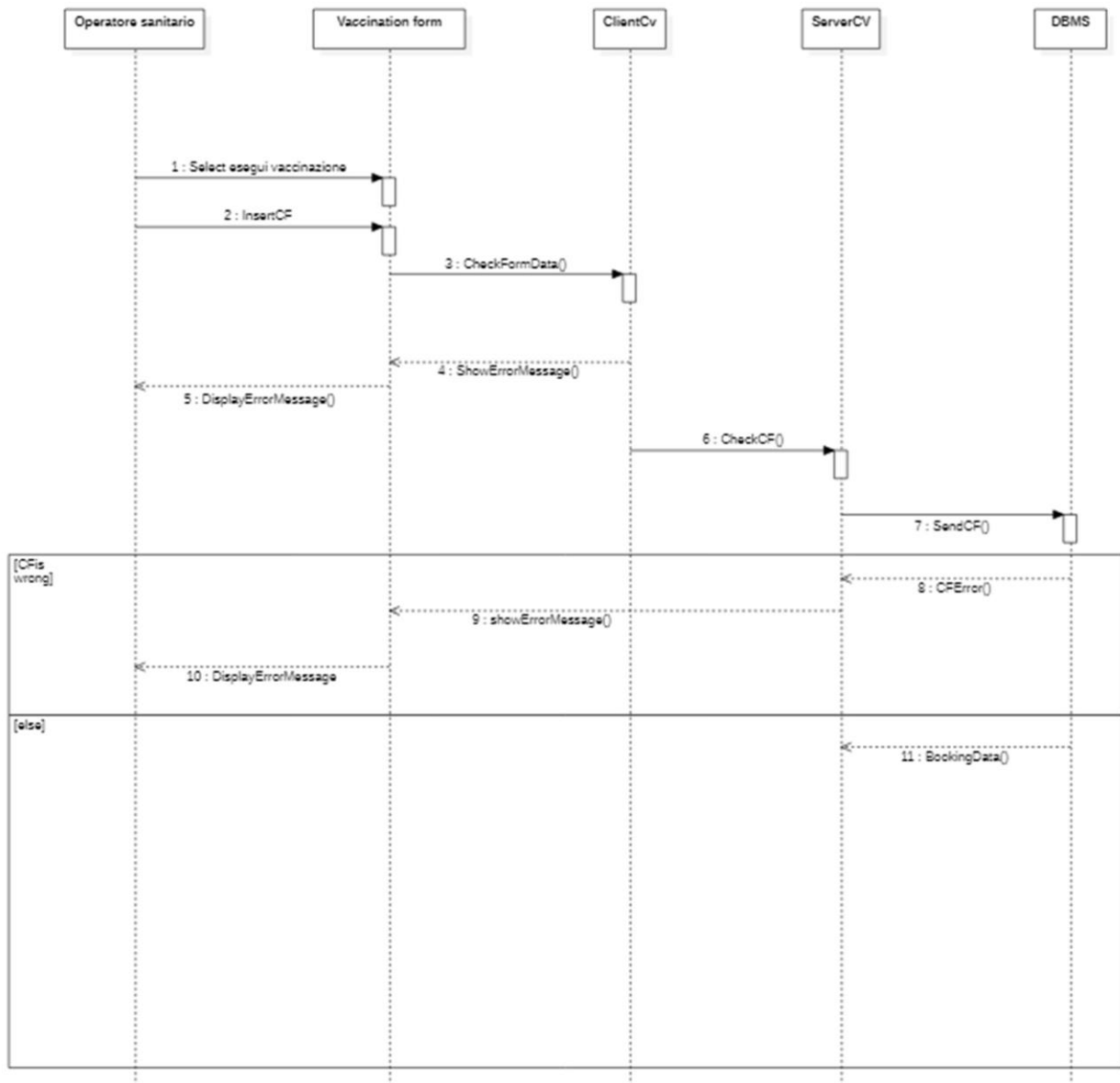


SEQUENCE DIAGRAM

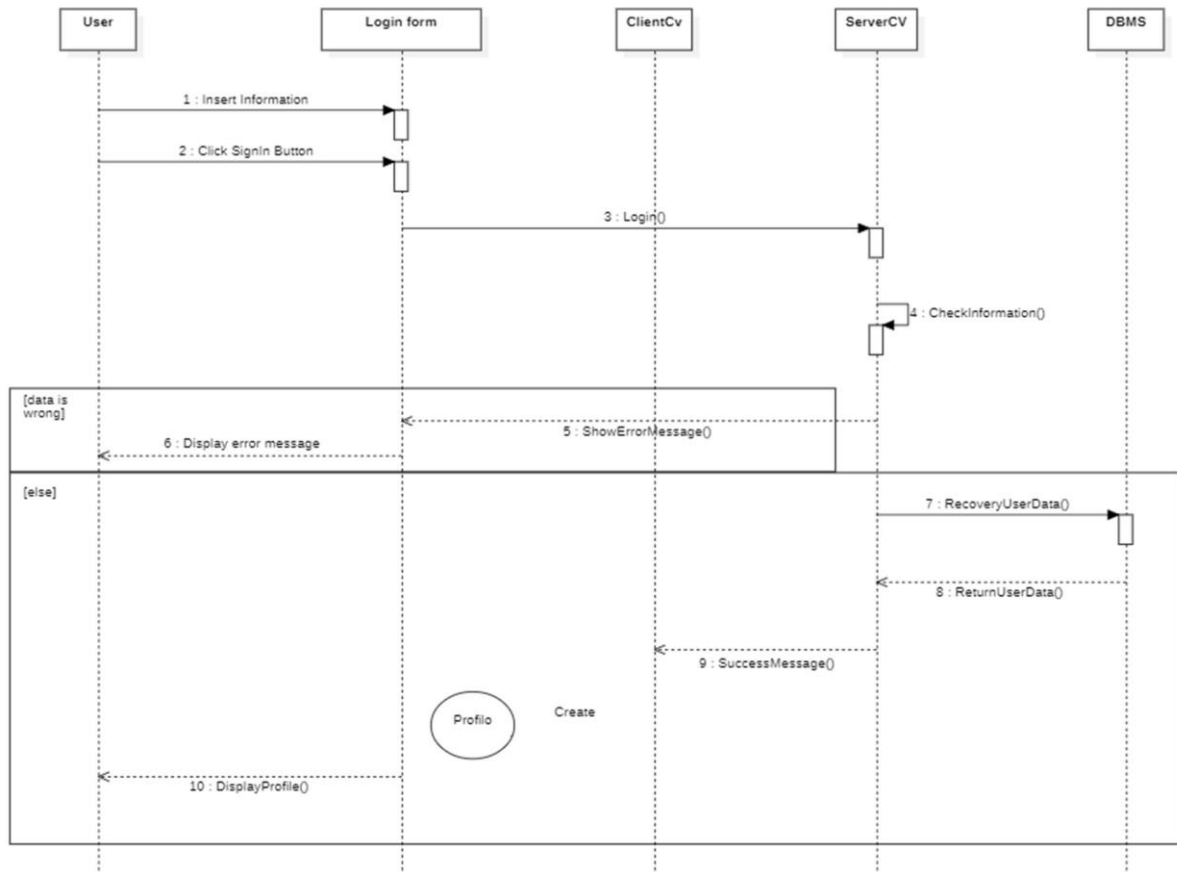
CREAZIONE CENTRO



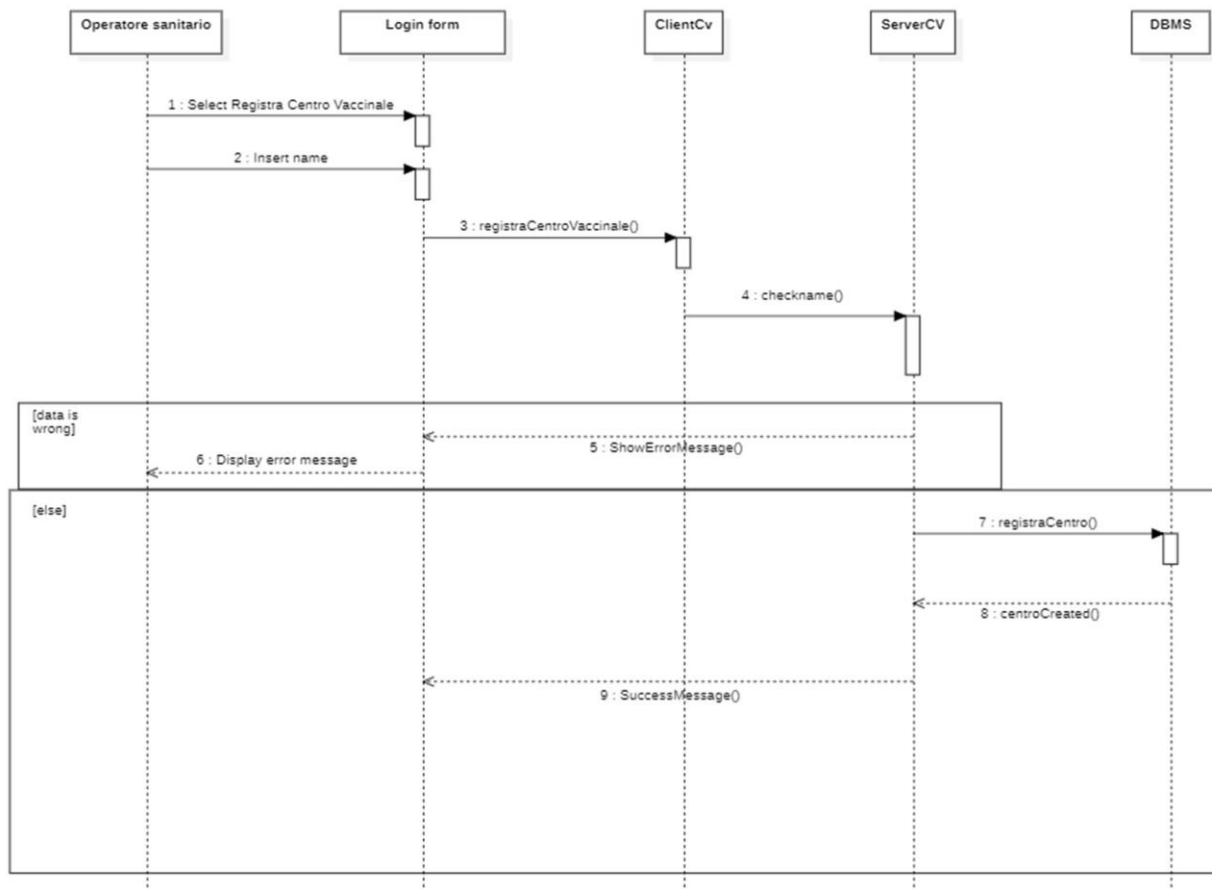
ESEGUI VACCINAZIONE



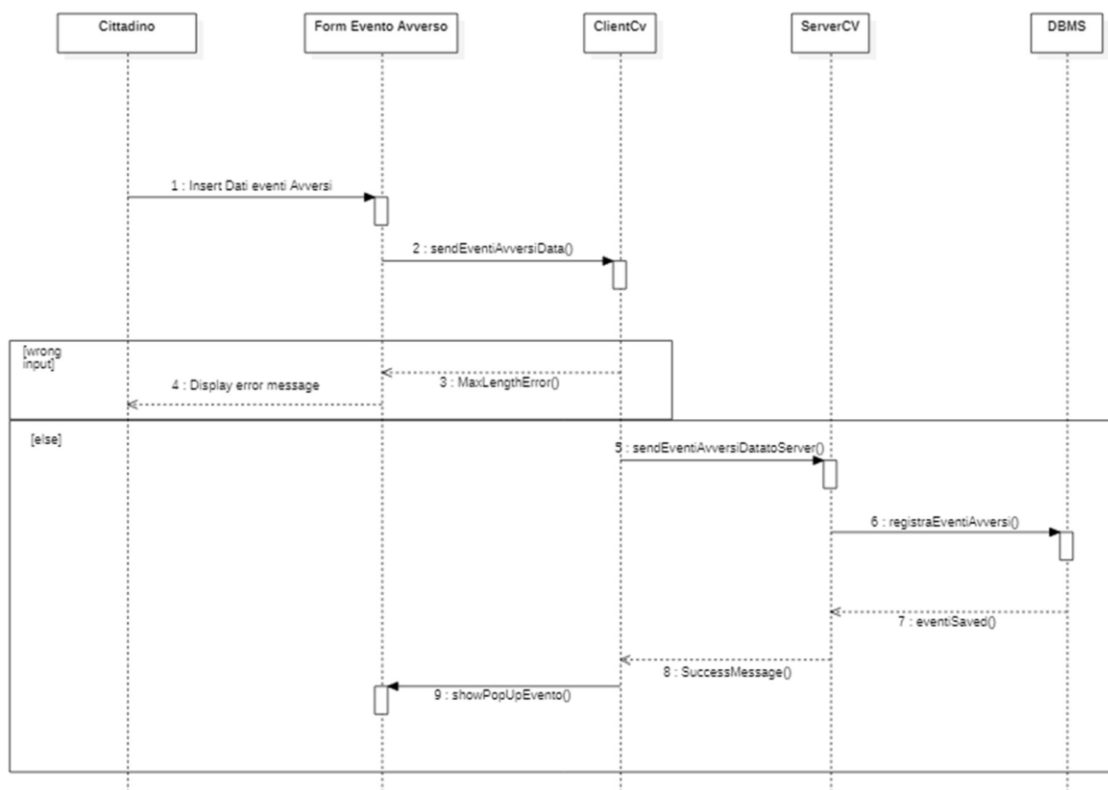
LOGIN



REGISTRA CENTRO

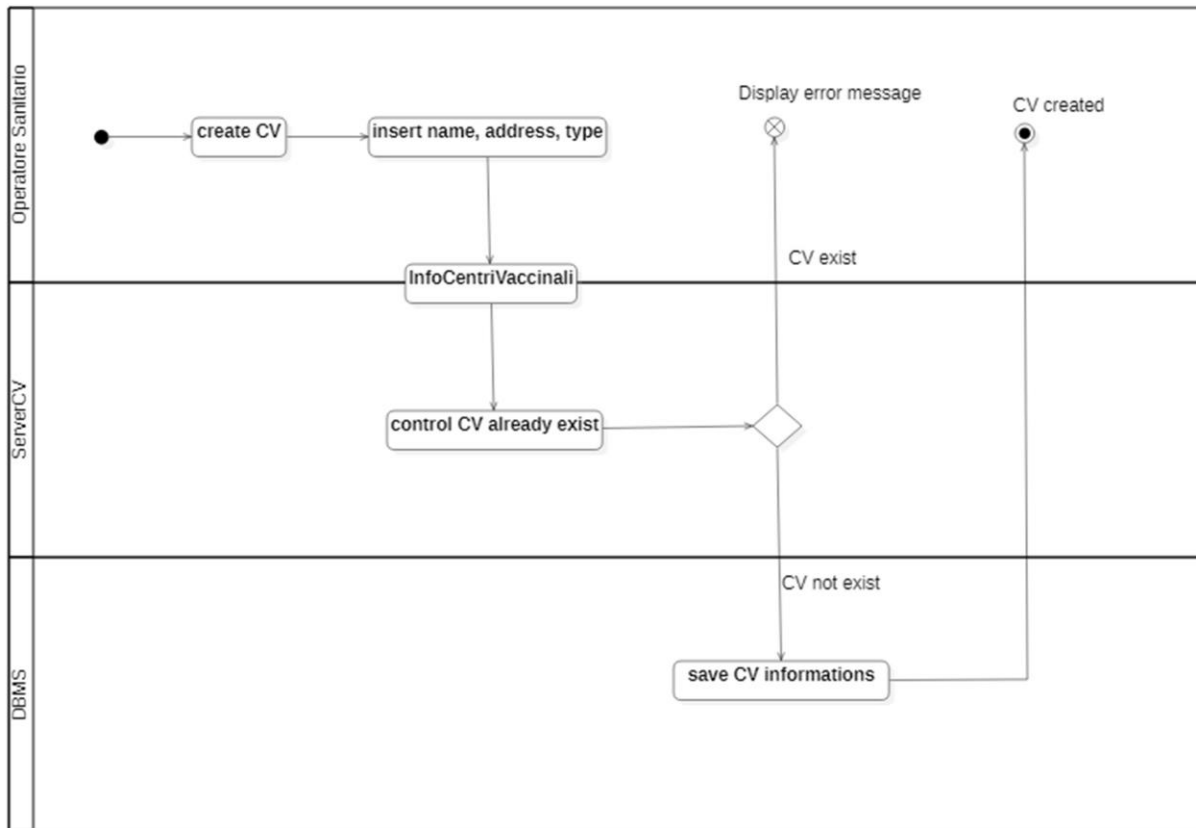


RICERCA CENTRO

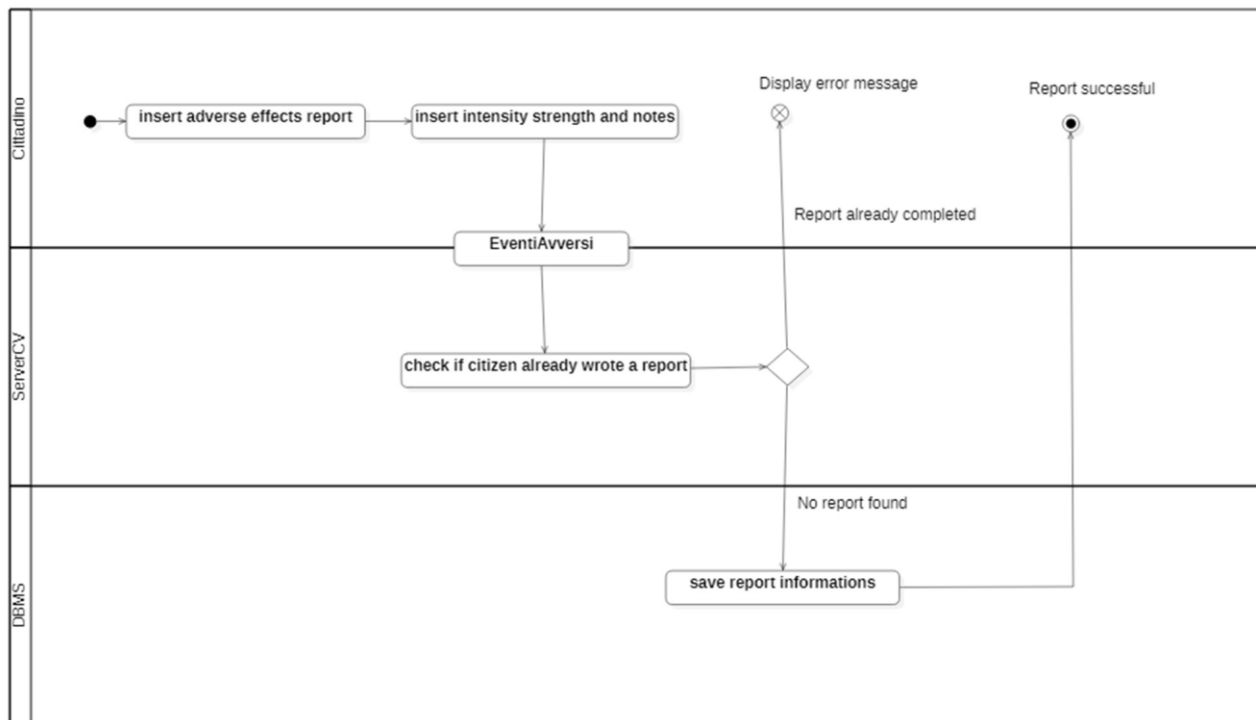


ACTIVITY DIAGRAM

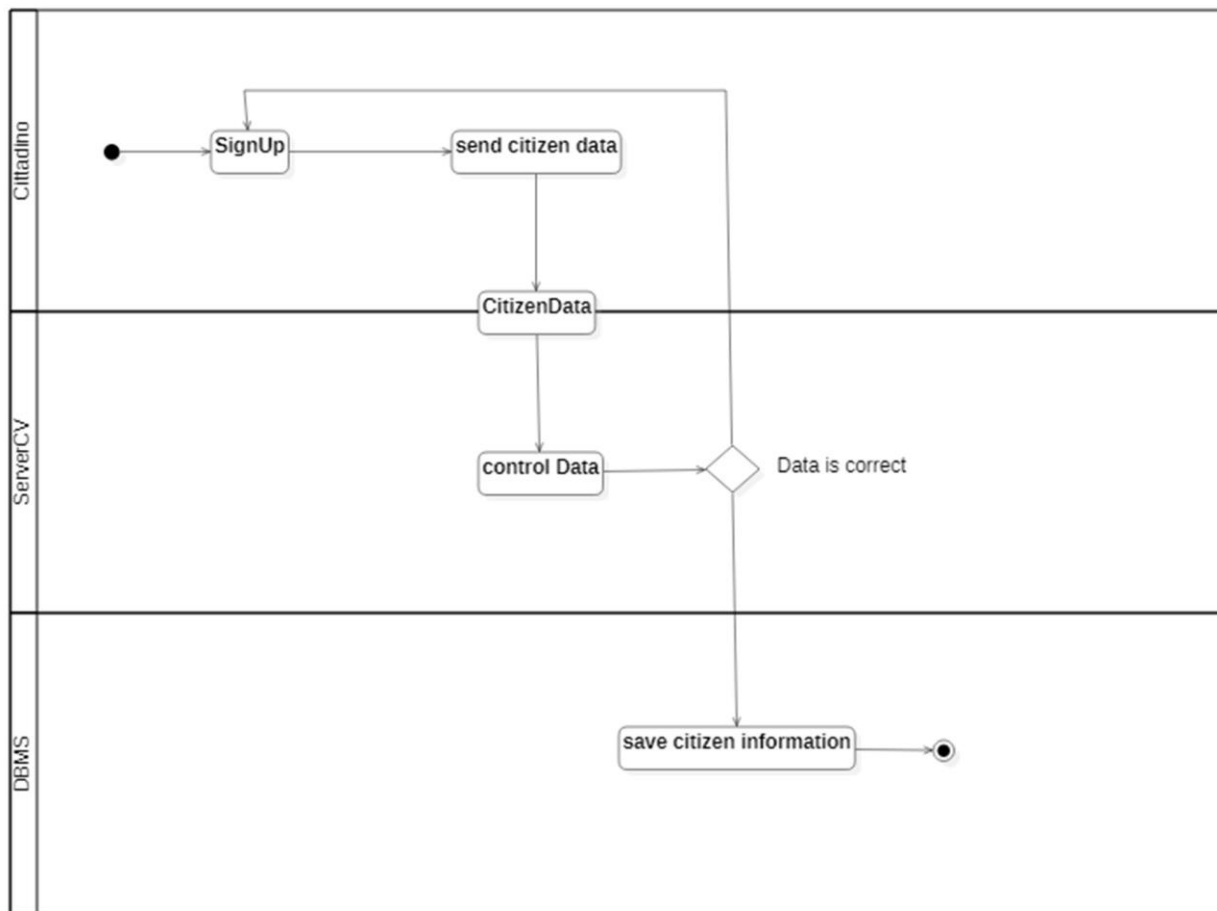
CREAZIONE CENTRO

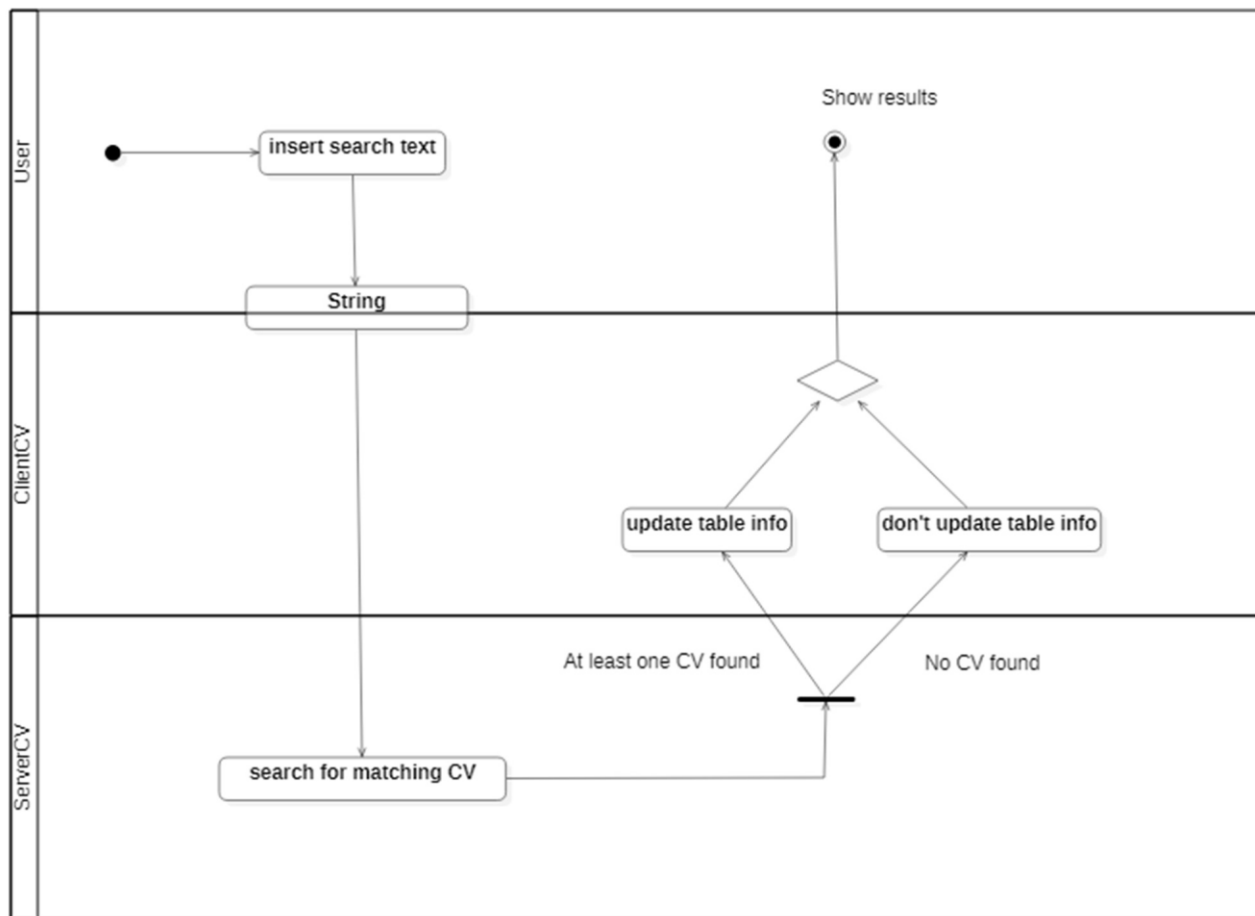


EVENTI AVVERSI

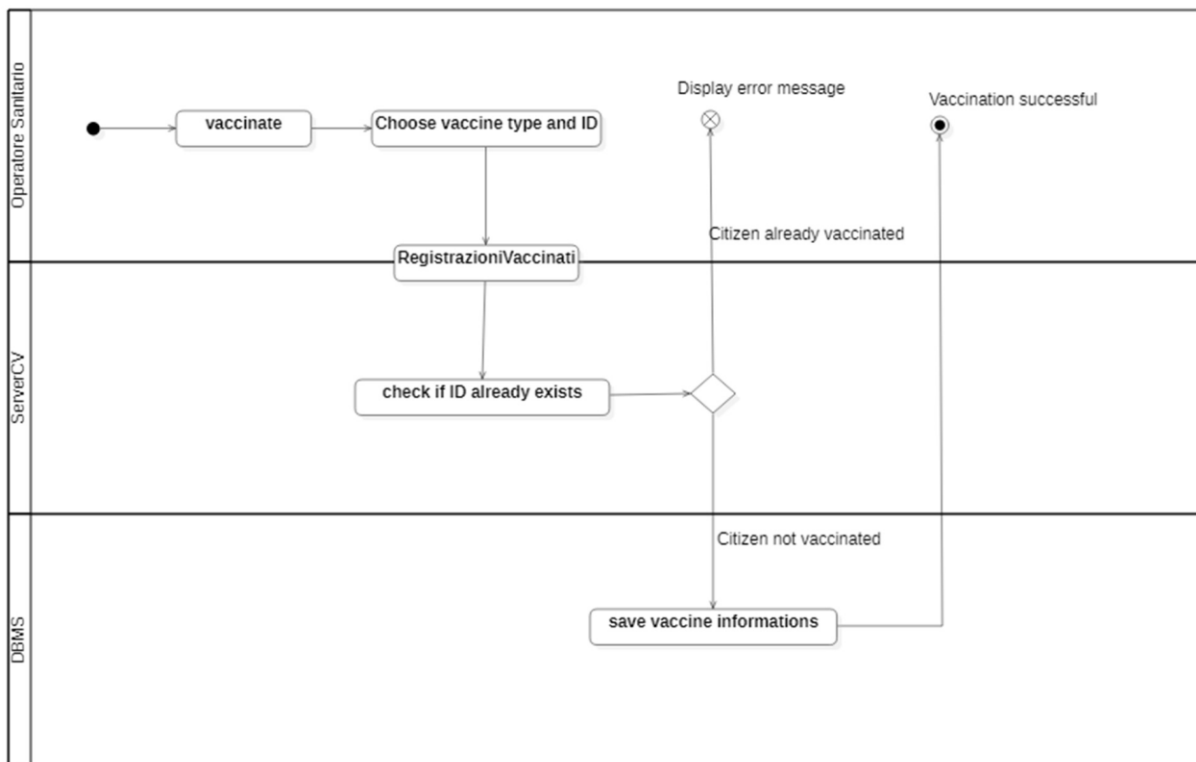


REGISTRAZIONE CITTADINO



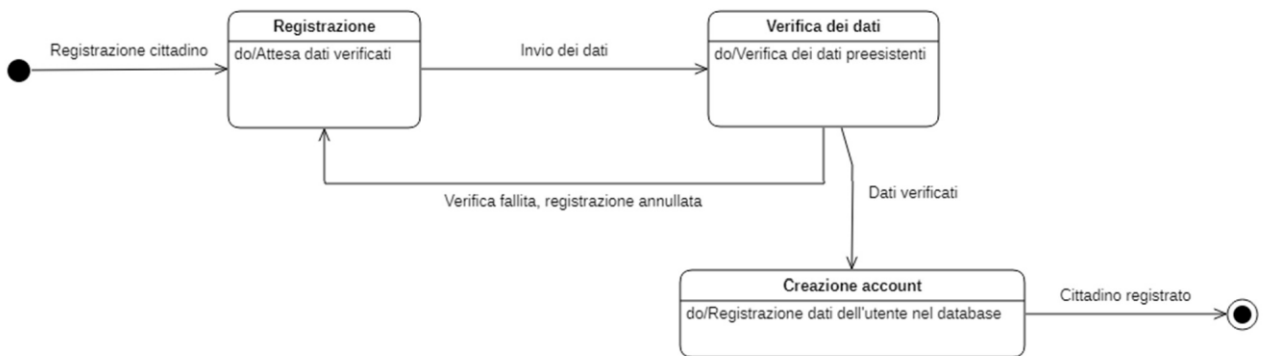


ESEGUI VACCINAZIONE

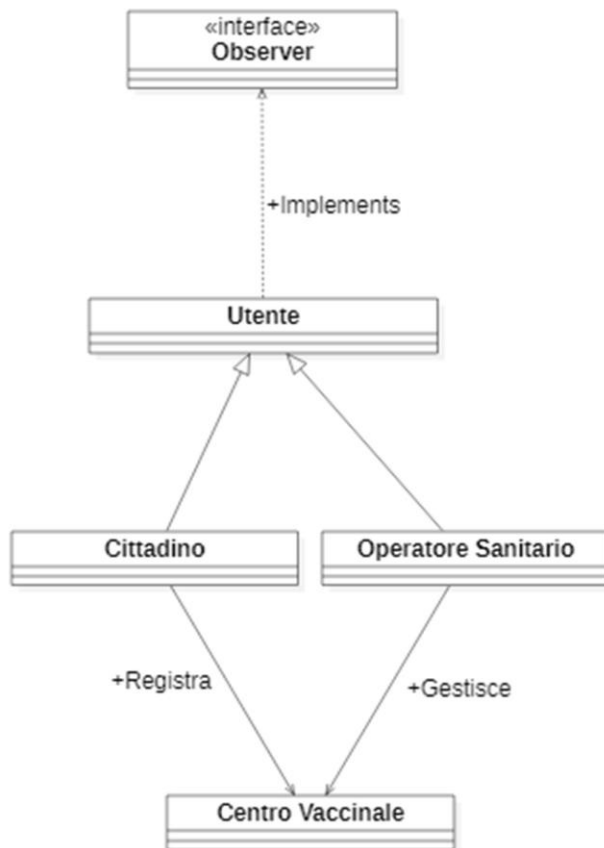


STATE DIAGRAM

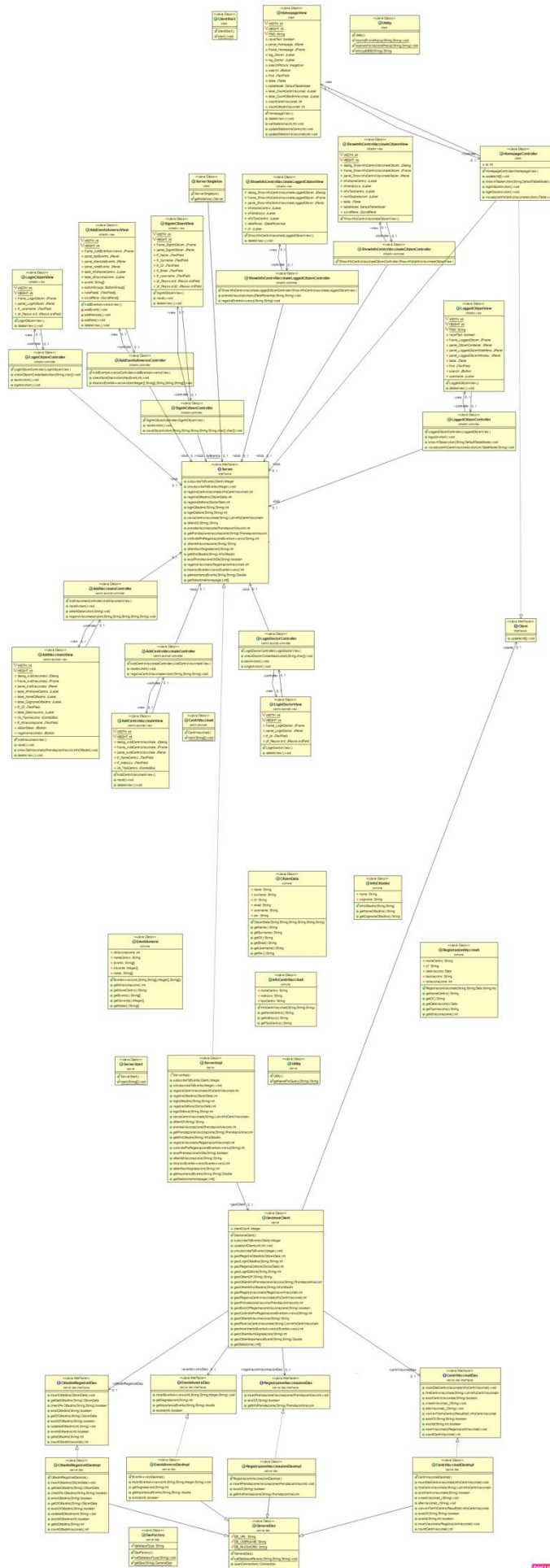
REGISTRAZIONE CITTADINO



CLASS DIAGRAM



GLOBAL CLASS DIAGRAM

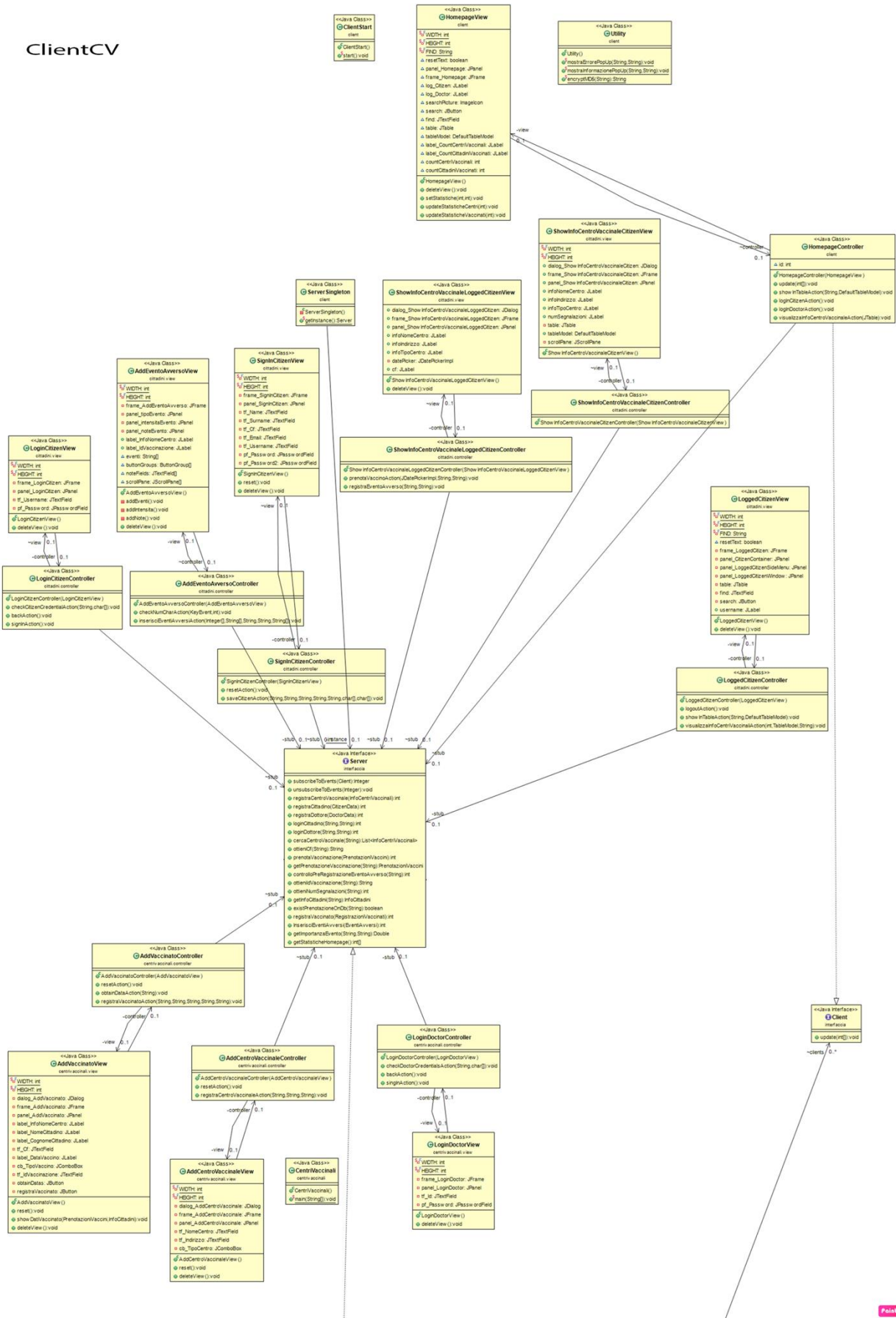


ClientCV

Classi Oggetto

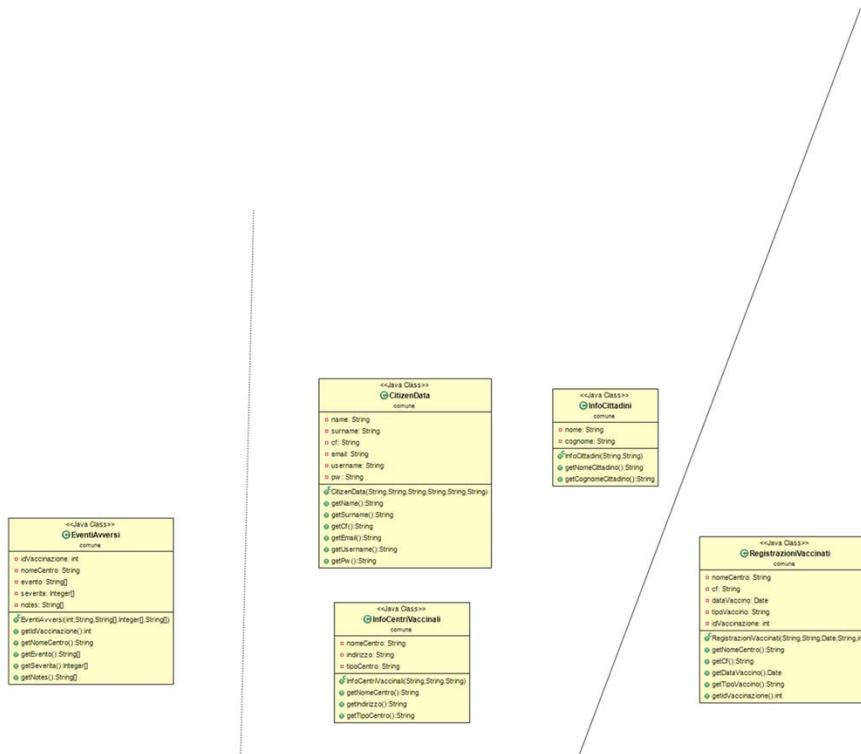
Server

GLOBAL CLASS DIAGRAM: CLIENTCV (FOCUS)



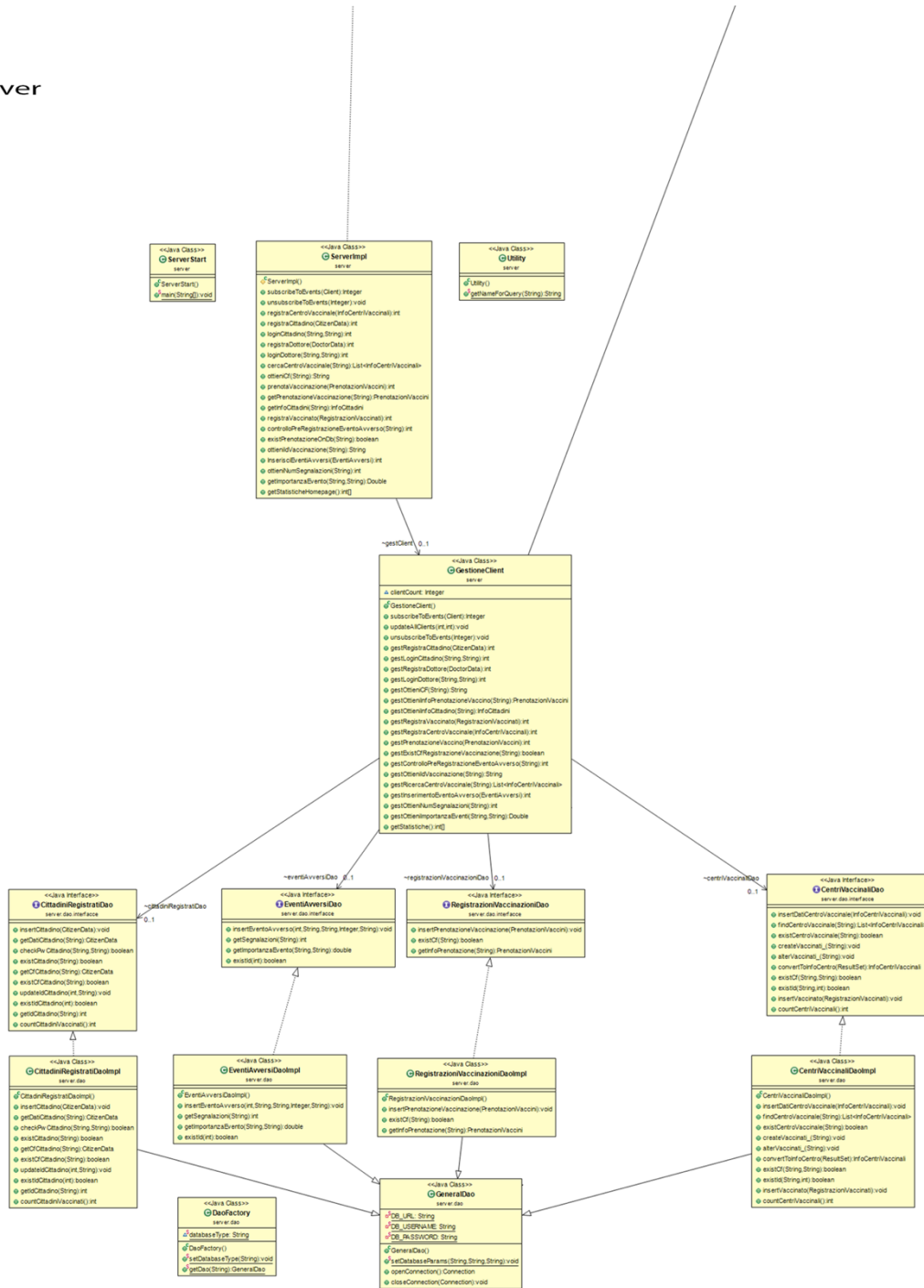
GLOBAL CLASS DIAGRAM: CLASSI OGGETTO (FOCUS)

Classi Oggetto



GLOBAL CLASS DIAGRAM: SERVERCV (FOCUS)

Server



DESCRIZIONE INTERFACCIA GRAFICA

Durante la fase di analisi dei requisiti abbiamo pensato a diverse soluzioni grafiche:

HOME PAGE

L' HOME PAGE è la finestra principale e viene mostrata all'utente dopo l'avvio del programma, in cui, tramite il menu di navigazione è possibile muoversi all'interno del programma.



CENTRI VACCINALI



LOGIN CITTADINO

Questa finestra compare dopo aver cliccato nel campo *login_cittadino*, i cittadini potranno così inserire USERNAME, PASSWORD e codice fiscale, in caso di utenti già registrati. In caso di utente non registrato sarà possibile effettuare la registrazione premendo nel campo “Sing-in”.



LOGIN CITTADINO

Username

Password

Codice Fiscale

BACK

SIGN-IN

LOGIN

LOGIN OPERATORE SANITARIO

Questa finestra compare dopo aver cliccato nel campo *login centri vaccinali*, dove gli operatori potranno così inserire USERNAME e PASSWORD con le rispettive informazioni, in caso di utenti già registrati. In caso di utente non registrato sarà possibile effettuare la registrazione premendo nel campo “sing-in”.



LOGIN Centro Vaccinale

Username

Password

BACK

SIGN-IN

LOGIN

REGISTRAZIONE CITTADINO

Questa finestra compare dopo aver cliccato nel campo *registrati* durante il login, nel quale i cittadini potranno inserire NOME, COGNOME, CF, EMAIL, USERNAME, PASSWORD, ID-VACCINAZIONE E ID-CENTRO con le rispettive informazioni, così da effettuare la registrazione premendo sull'apposito bottone.



REGISTRAZIONE CITTADINO

Nome

Cognome

CF

Email

Username

Password

ID-Vaccinazione

ID-Centro




BACK

SIGN-IN

RESET

EFFETTUA SEGNALAZIONE EVENTO AVVERSO

Questa schermata comparirà una volta effettuato il login come centro vaccinale, contiene una lista di problematiche più frequenti dovute a una vaccinazione con la possibilità di indicare severità di quest'ultima, prevede anche la possibilità di inserimento di un testo di massimo 256 caratteri per ogni problematica utile per la segnalazione di problematiche diverse da quelle già presenti



ID Centro:
legnano2022

ID Vaccinazione:
vaccinato22

Eventi Avversi:

Tipo Evento:	Intensità:	Note:
Mal di testa	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili
Mal di pancia	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili
Febbre	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili
Dolori articolari - muscolari	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili
Linfoadenopatia	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili
Tachicardia	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili
Crisi ipertensiva	<div><input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5</div>	<div></div> 256 caratteri disponibili

Indietro

Salva evento

REGISTRA CENTRO

Questa finestra compare dopo aver cliccato nel campo *registrati* durante il login, nel quale gli operatori potranno inserire ID CENTRO, NOME CENTRO, TIPOLOGIA CENTRO, QUALIFICATORE, INDIRIZZO, NUMERO CIVICO, COMUNE, PROVINCIA, CAP, USERNAME E PASSWORD, così da effettuare la registrazione premendo sull'apposito bottone “sing-in”.



REGISTRA NUOVO Centro Vaccinale

ID centro	<input type="text"/>
Nome centro	<input type="text"/>
Tipologia Centro	<input type="text"/>
via/viale/piazza	<input type="text"/>
Indirizzo	<input type="text"/>
Numero Civico	<input type="text"/>
Comune	<input type="text"/>
Provincia (sigla)	<input type="text"/>
CAP	<input type="text"/>
Username	<input type="text"/>
Password	<input type="password"/>

BACK

SIGN-IN

RESET

REGISTRAZIONE VACCINATO

Questa finestra compare dopo aver effettuato il login come centro vaccinale. Permette all'operatore, dopo aver inserito i dati del vaccinato, di registrare la vaccinazione.



Registra Vaccinato

Nome centro vaccinale

ID centro

Codice fiscale

Data (dd/MM/yyyy)

Nome vaccino

ID vaccinazione

Nome

Cognome

BACK

REGISTRA

SCHERMATA ACCESSO LIBERO

Questa finestra comparirà al Cittadino dopo aver cliccato sul tasto *accesso libero*, sarà presente un tasto per perproseguire alla ricerca dei centri.




Accesso Libero

Back

Consulta Informazioni Centro Vaccinale

OPERAZIONI CITTADINO

Questa finestra compare dopo aver cliccato su ricerca centro vaccinale. Qui i cittadini potranno effettuare la ricerca in base a tre differenti campi: ricerca per nome, ricerca per comune e ricerca per tipologia.

 Consulta Informazioni Centri Vaccinali

—

□

×

Digita per Cercare il Centro Vaccinale di tuo interesse

Cerca per Nome Centro Vaccinale

Cerca per Comune

Cerca per Tipologia

Back

Cerca

Reset

EVENTI AVVERSI

Questa finestra compare dopo aver effettuato la ricerca dei centri per uno dei tre campi di ricerca, questa schermata conterrà una tabella contenente i dati dei centri risultati dalla ricerca.

Risultati ricerca												
Nome	Tipologia	Provincia	Comune	Indirizzo	Civico	Mal di testa	Mal di pancia	Febbre	Dolori articolari	Linfoadenopatia	Tachicardia	Crisi ipertensiva

Indietro

STRUTTURE DATI IMPLEMENTATE

- ArrayList: Utilizzata per salvare e visualizzare i dati dei Centri Vaccinali. Il costo per l'inserimento è $O(n)$, mentre il costo per l'acquisizione dei dati è costante ($O(1)$).
- LinkedList: Utilizzata per salvare e visualizzare gli ID dei Client che sono stati chiusi e quindi rimossi dall'Observer. Il costo per l'inserimento è costante ($O(1)$), mentre l'acquisizione di un dato è $O(n)$.
- Array: Utilizzata per salvare e visualizzare molteplici informazioni in svariati campi. Il costo per l'acquisizione dei dati è $O(n)$, in quanto gli array non sono ordinati.
- ConcurrentHashMap: Utilizzata per salvare e visualizzare i Client. Sia il costo per l'inserimento che quello per l'acquisizione dei dati è costante ($O(1)$).

N.B. Tutte le complessità riportate fanno riferimento al tempo.

PROGETTAZIONE

SCELTE STRATEGICHE

DAO (DATA ACCESS OBJECT):

Il pattern Data Access Object (DAO) è un pattern architetturale utilizzato per separare la logica applicativa dalle operazioni di accesso al livello di gestione dati del database. Il Data Access Object nasconde completamente i dettagli dell'interazione con la sorgente dati. L'interfaccia esposta dal DAO al server non cambia quando l'implementazione dell'origine dati sottostante cambia, e questo consente al pattern di adattarsi a diversi schemi di archiviazione senza dover modificare nulla sugli altri layer. In breve, il DAO ha il ruolo di *adapter* tra l'interfaccia utente e l'origine dati.

SINGLETON:

Utilizzato dal Server, permette la creazione di una singola istanza di tipo Server. Per assicurarci che ci sia una sola istanza del server e che faccia da unico punto di accesso al database, ci si è affidati al Singleton.

FACTORY LAYOUT:

Utilizzato dal Database, permette la creazione del *DAO* senza esporre la sua logica di creazione al Client.

OBSERVER:

Utilizzato sia dal lato Client che dal lato Server, aggiorna le statistiche dei Centri Vaccinali esistenti e dei Cittadini Vaccinati, mostrando nell'Home Page il numero di Centri Vaccinali e di Cittadini Vaccinati.

MVC (MODEL-VIEW-CONTROLLER):

Utilizzato dal lato Client divide le GUI in due parti, quella di VIEW contenente solo le impostazioni grafiche, e quella di CONTROLLER contenente solo la logica per quella specifica VIEW. Al lato client si è utilizzato il pattern Model-View-Controller per separare la logica applicativa dall'interfaccia utente il cui risultato è stato di una divisione in due package del programma: View (per la GUI) e Controller (per la logica di controllo).

LOGIN/CONNESSIONE AL DATABASE TRAMITE COMANDO APACHE ANT:

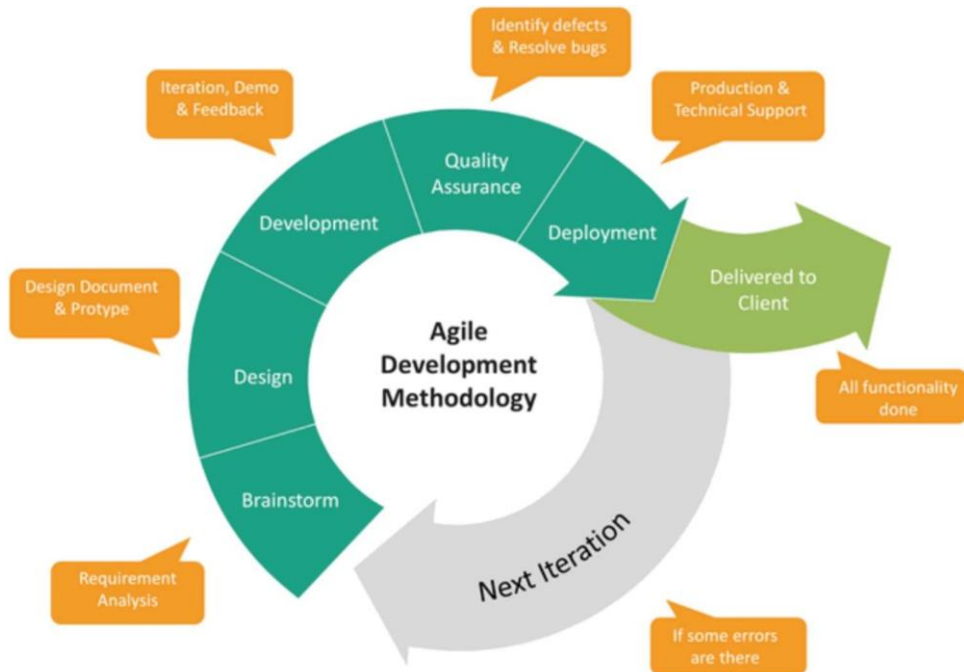
Per mantenere la logica applicativa del server indipendente dal database si è deciso di implementare la procedura di creazione delle tabelle della base di dati esternamente tramite uno script di Apache ANT.

SALVATAGGIO SUL DATABASE SOLO DI PASSWORD CRIPTATE:

Per garantire una maggior privacy/sicurezza dai dati personali degli utenti, nel database e nel server non verrà passata alcuna password in chiaro. Di queste viene sempre calcolato la rispettiva funzione di hash di tipo MD5.

PROGETTAZIONE LAVORATIVA

Inizialmente per la realizzazione del progetto ci siamo basati sul classico modello a cascata o WATERFALL per poi confluire in un modello ITERATIVE AGILE perché meglio si adattava al metodo TOP-DOWN che stavamo seguendo. Lo scopo era, una volta giunti ad una release stabile, provare a migliorarla ulteriormente.



L'organizzazione lavorativa ha previsto la suddivisione mista permettendo ad ogni membro della squadra di procedere per la parte scelta, fermo restando la possibilità di assist su un componente.

Per i requisiti funzionali ci siamo basati sulle richieste del committente.







Prima di mettere mano alla stesura del codice abbiamo redatto la rappresentazione grafica attraverso i diagrammi UML.

La realizzazione è avvenuta in ambiente Microsoft Windows, e abbiamo inoltre verificato la portabilità su ambiente Linux.

L'interfaccia è stata realizzata in modo da essere subito utilizzabile dall'operatore finale (*user-friendly*). In nessuno dei nostri test il sistema è stato forzatamente arrestato da un crash. Pur avendo intenzionalmente provocato errori non abbiamo riscontrato anomalie e malfunzionamenti.

Le Password vengono registrate nel sistema con codifica MD5.

I requisiti del software sono molto bassi quindi adatti a macchine non necessariamente aggiornate.

Nome	Stato	CPU	Memoria	Disco	Rete	GPU
Applicazioni (14)						
>  Blocco note		0%	2,4 MB	0 MB/s	0 Mbps	0%
>  Cattura e annota (3)		1,8%	23,4 MB	0,1 MB/s	0 Mbps	0,2%
>  Esplora risorse (6)		2,9%	262,5 MB	0,1 MB/s	0 Mbps	0%
>  HeidiSQL 11.3.0.6295 64 Bit		0%	20,8 MB	0 MB/s	0 Mbps	0%
>  Java(TM) Platform SE binary (2)		0%	80,3 MB	0 MB/s	0 Mbps	0%

PROGETTAZIONE DELLA BASE DI DATI

RACCOLTA E ANALISI DEI REQUISITI

SPECIFICHE SUI DATI IN LINGUAGGIO NATURALE

I dati relativi al cittadino che si desidera memorizzare nella base di dati implementata sull'applicazione "Centri vaccinali" prevede la permanenza di:

- Cf varchar
- Nome varchar
- Cognome varchar
- Indirizzo_posta varchar, unique
- Userid varchar, unique
- Password varchar
- Idvaccinazione varchar, unique
- Id_centro varchar

I dati relativi al centro vaccinale che si desidera memorizzare nella base di dati implementata sull'applicazione "Centri vaccinali" prevede la permanenza di:

- Id_centro varchar
- Nome_centro varchar
- Tipologia varchar
- Qualificatore varchar
- Nome_via varchar
- Num_civ varchar
- Comune varchar
- Provincia varchar
- Cap int
- Username varchar, unique
- Password varchar

Un centro vaccinale può inserire un vaccinato nella base di dati implementata sull'applicazione "Centri vaccinali". La tabella avrà il nome vaccinati_(nome del centro che sta inserendo il vaccinato) e conterrà i seguenti dati:

- Cf varchar
- Nome varchar
- Cognome varchar
- Data_vaccinazione varchar
- Vaccino_somministrato varchar
- Idvaccinazione varchar, unique
- Id_centro varchar

Un cittadino vaccinato registrato presso un centro può inserire gli eventi avversi. La tabella conterrà i seguenti dati:

- Id_vaccinazione varchar
- Id_centro varchar

- Nome_evento varchar
- Note_opzionali varchar
- Severita int
- Cf varchar

CARICO DI LAVORO

Si prevede che gli utenti possano essere un numero molto elevato. Sapendo che i cittadini residenti in Italia sono 59.257.566, non tutti aderiranno al piano vaccinale per motivi di scelta (10%) e altri per motivi di salute (5%), si potranno vaccinare solo 50.368.931 persone.

La nostra applicazione funzionerà sulle città a maggior densità di popolazione di ogni regione d'Italia. Supponiamo solo i comuni dove la popolazione superi le 50.000 unità (ove possibile):

- Lombardia (9.966.992 residenti):
 - Milano: (1.397.715)
 - Brescia: (195.102)
 - Monza: (123.983)
 - Bergamo: (119.684)
 - Como: (84.808)
 - Busto Arsizio: (83.121)
 - Sesto San Giovanni: (80.589)
 - Varese: (80.039)
 - Cinisello balsamo: (73.537)
 - Cremona: (71.523)
 - Pavia: (70.971)
 - Vigevano: (62.108)
 - Legnano: (58.938)
 - Gallarate: (53.960)
 - Rho: (49.609)
 - Mantova: (48.523)
 - Paderno Dugnano: (47.980)
 - Lecco: (47.415)
 - Cologno Monzese: (46.687)
 - Lissone: (45.998)
 - Seregno: (44.742)
 - Lodi: (44.574)
 - ...
- Lazio (5.720.796 residenti):
 - Roma: (2.783.809)
 - Latina: (126.612)
 - Guidonia Montecelio: (86.732)
 - Fiumicino: (76.246)
 - Aprilia: (73.296)
 - Viterbo: (65.050)
 - Pomezia: (61.298)
 - Tivoli: (54.046)
 - Velletri: (52.943)

- Anzio: (52.373)
- Civitavecchia: (51.548)
- Ardea: (48.365)
- Nettuno: (47.576)
- Rieti: (46.193)
- Marino: (44.905)
- Frosinone: (44.485)
- ...
- Campania (5.679.759 residenti):
 - Napoli: (940.940)
 - Salerno: (130.240)
 - Giugliano in Campania: (118.906)
 - Torre del Greco: (82.329)
 - Pozzuoli: (78.870)
 - Casoria: (74.416)
 - Caserta: (73.398)
 - Castellamare di Stabia: (63.824)
 - Afragola: (62.271)
 - Acerra: (58.961)
 - Benevento: (57.778)
 - Marano di Napoli: (57.750)
 - Portici: (53.254)
 - Avellino: (53.064)
 - ...
- Veneto (4.852.453 residenti):
 - Verona: (257.838)
 - Venezia: (255.609)
 - Padova: (208.306)
 - Vicenza: (109.290)
 - Treviso: (84.930)
 - Rovigo: (49.844)
 - Chioggia: (48.130)
 - Bassano del Grappa: (42.168)
 - San Donà di Piave: (42.131)
 - Schio: (38.775)
 - Mira: (37.974)
 - Belluno: (35.505)
 - ...
- Sicilia (4.840.876 residenti):
 - Palermo: (640.720)
 - Catania: (294.298)
 - Messina: (225.546)
 - Siracusa: (118.093)

- Marsala: (80.369)
- Gela: (71.535)
- Ragusa: (71.281)
- Trapani: (65.249)
- Vittoria: (62.271)
- Caltanissetta: (59.425)
- Agrigento: (56.045)
- ...
- Emilia-Romagna (4.445.549 residenti):
 - Bologna: (394.463)
 - Parma: (200.218)
 - Modena: (187.977)
 - Reggio Emilia: (169.803)
 - Ravenna: (157.422)
 - Rimini: (148.688)
 - Ferrara: (132.288)
 - Forlì: (117.479)
 - Piacenza: (103.294)
 - Cesena: (97.254)
 - ...
- Piemonte (4.273.210 residenti):
 - Torino: (848.196)
 - Novara: (101.916)
 - Alessandria: (92.104)
 - Asti: (74.348)
 - Moncalieri: (56.522)
 - Cuneo: (55.980)
 - Collegno: (49.099)
 - Rivoli: (47.318)
 - Settimo Torinese: (46.699)
 - Nichelino: (46.535)
 - Vercelli: (45.875)
 - Bielli: (43.663)
 - Grugliasco: (36.700)
 - Chieri: (35.912)
 - Pinerolo: (35.546)
 - Casale Monferrato: (33.213)
 - Venaria Reale: (32.818)
 - Alba: (31.215)
 - Verbania: (30.104)
 - ...
- Puglia (3.926.931 residenti):
 - Bari: (313.003)

- Taranto: (189.366)
- Foggia: (148.301)
- Andria: (97.942)
- Lecce: (93.673)
- Barletta: (92.927)
- Brindisi: (83.848)
- Altamura: (69.818)
- Molfetta: (57.682)
- Trani: (55.175)
- ...
- Toscana (3.668.333 residenti):
 - Firenze: (359.755)
 - Prato: (193.568)
 - Livorno: (156.031)
 - Arezzo: (97.842)
 - Pistoia: (90.479)
 - Pisa: (89.155)
 - Lucca: (88.822)
 - Grosseto: (81.621)
 - Massa: (66.886)
 - Carrara: (60.685)
 - Viareggio: (60.144)
 - Siena: (54.195)
 - ...
- Calabria (1.877.728 residenti):
 - Reggio Calabria: (173.456)
 - Catanzaro: (86.590)
 - Corigliano Rossano: (74.850)
 - Lamezia Terme: (67.713)
 - Cosenza: (65.197)
 - Crotone: (60.112)
 - Rende: (35.692)
 - Vibo Valentia: (31.097)
 - ...
- Sardegna (1.598.225 residenti):
 - Cagliari: (149.474)
 - Sassari: (124.111)
 - Quartu Sant'Elena: (67.823)
 - Olbia: (60.491)
 - Alghero: (42.295)
 - Nuoro: (34.536)
 - Oristano: (30.383)
 - Selargius: (28.631)

- Carbonia: (26.472)
- ...
- Liguria: (1.509.805 residenti):
 - Genova: (558.930)
 - La Spezia: (91.877)
 - Savona: (58.566)
 - Sanremo: (52.327)
 - Imperia: (41.976)
 - ...
- Marche: (1.501.406 residenti):
 - Ancona: (99.108)
 - Pesaro: (96.238)
 - Fano: (60.105)
 - San Benedetto del T.: (47.485)
 - Ascoli Piceno: (46.760)
 - Senigallia: (44.183)
 - Civitanova Marche: (41.989)
 - Macerata: (40.762)
 - Jesi: (39.302)
 - Fermo: (36.386)
 - ...
- Abruzzo: (1.285.256 residenti):
 - Pescara: (119.455)
 - L'Aquila: (69.941)
 - Teramo: (53.481)
 - Montesilvano: (53.113)
 - Chieti: (50.008)
 - ...
- Friuli-Friuli-Venezia Giulia: (1.198.753 resi-denti):
 - Trieste: (199.773)
 - Udine: (99.736)
 - Pordenone: (51.568)
 - Gorizia: (33.479)
 - ...
- Trentino-Alto Adige: (1.078.460 resi- denti):
 - Trento: (120.491)
 - ...
- Umbria: (865.013 residenti):
 - Perugia: (164.057)

- Terni: (109.255)
- ...
- Basilicata: (547.579 residenti):
 - Potenza: (65.988)
 - Matera: (60.295)
 - ...
- Molise: (296.547 residenti):
 - Campobasso: (47.849)
 - Termoli: (32.949)
 - Isernia: (20.972)
 - ...
- Valle d'Aosta: (123.895 residenti):
 - Aosta: (33.409)
 - ...

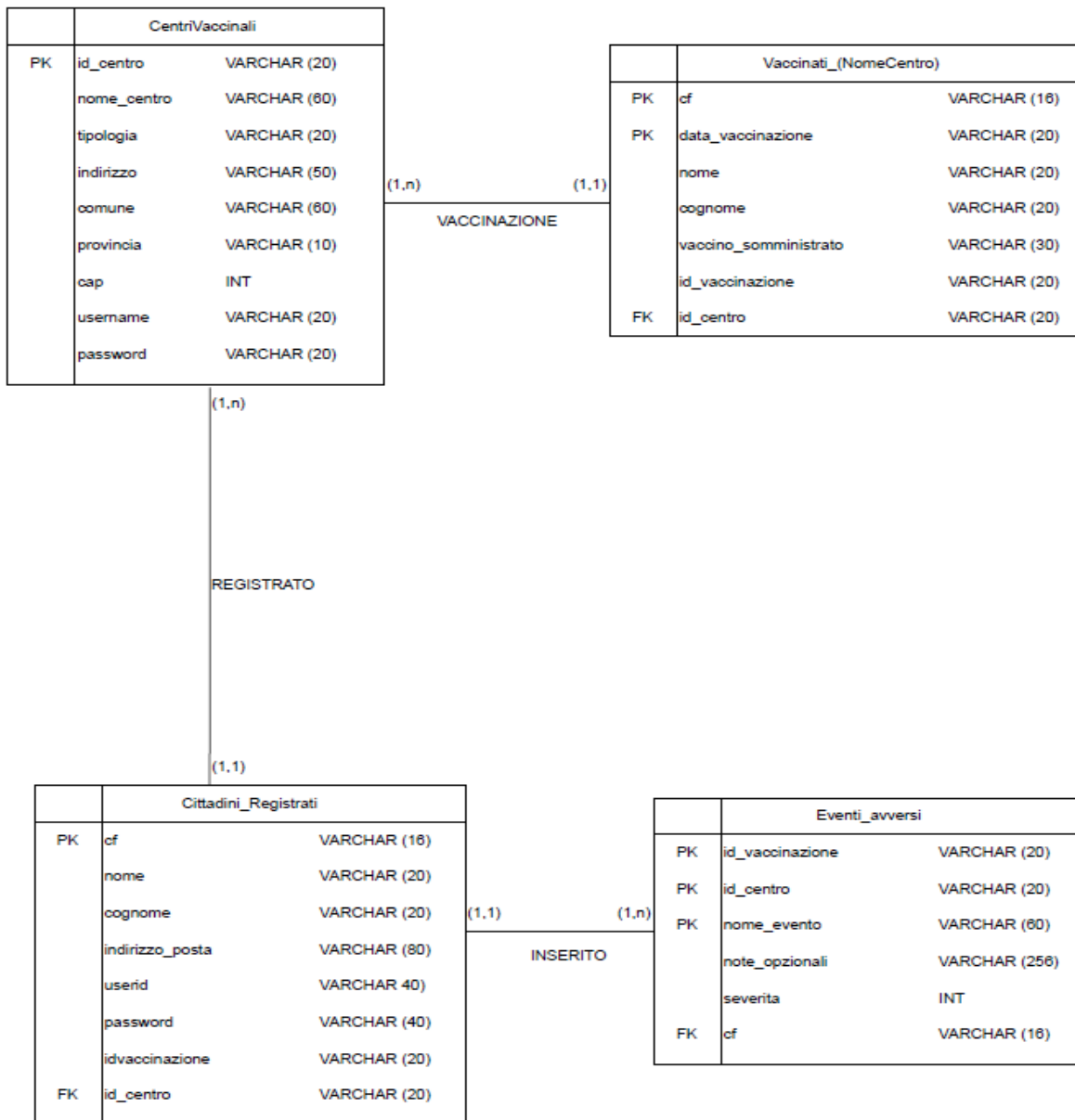
Ipotizzando un carico di utenti pari al 85% della popolazione italiana, abbiamo dedotto che la media regionale della popolazione che non effettuerà una vaccinazione è di circa lo 0.75%.

Il numero di segnalazioni per eventi avversi è una per cittadino vaccinato quindi, immaginando una lenta adozione della nostra applicazione e altri fattori popolari, per ogni centro vaccinale non avremo più di 1000 vaccini al giorno, ovvero al massimo 1000 segnalazioni di eventi avversi.

SCELTE PROGETTUALI

- Data_vaccinazione messa come varchar per questioni di conversioni nell'applicazione

SCHEMA CONCETTUALE



DOCUMENTAZIONE

VINCOLI D'INTEGRITÀ

Centrivaccinali:

- Id_centro primary key
- Nome_centro not null
- Tipologia not null
- Qualificatore not null
- Nome_via not null
- Num_civ not null
- Comune not null
- Provincia not null

- Cap not null
- Username unique e not null
- Password not null

Cittadini_registrati:

- Cf primary key
- Nome not null
- Cognome not null
- Indirizzo_posta unique e not null
- Userid unique e not null
- Password not null
- Idvaccinazione unique e not null
- Id_centro foreign key e not null

Vaccinati_(nome_centro):

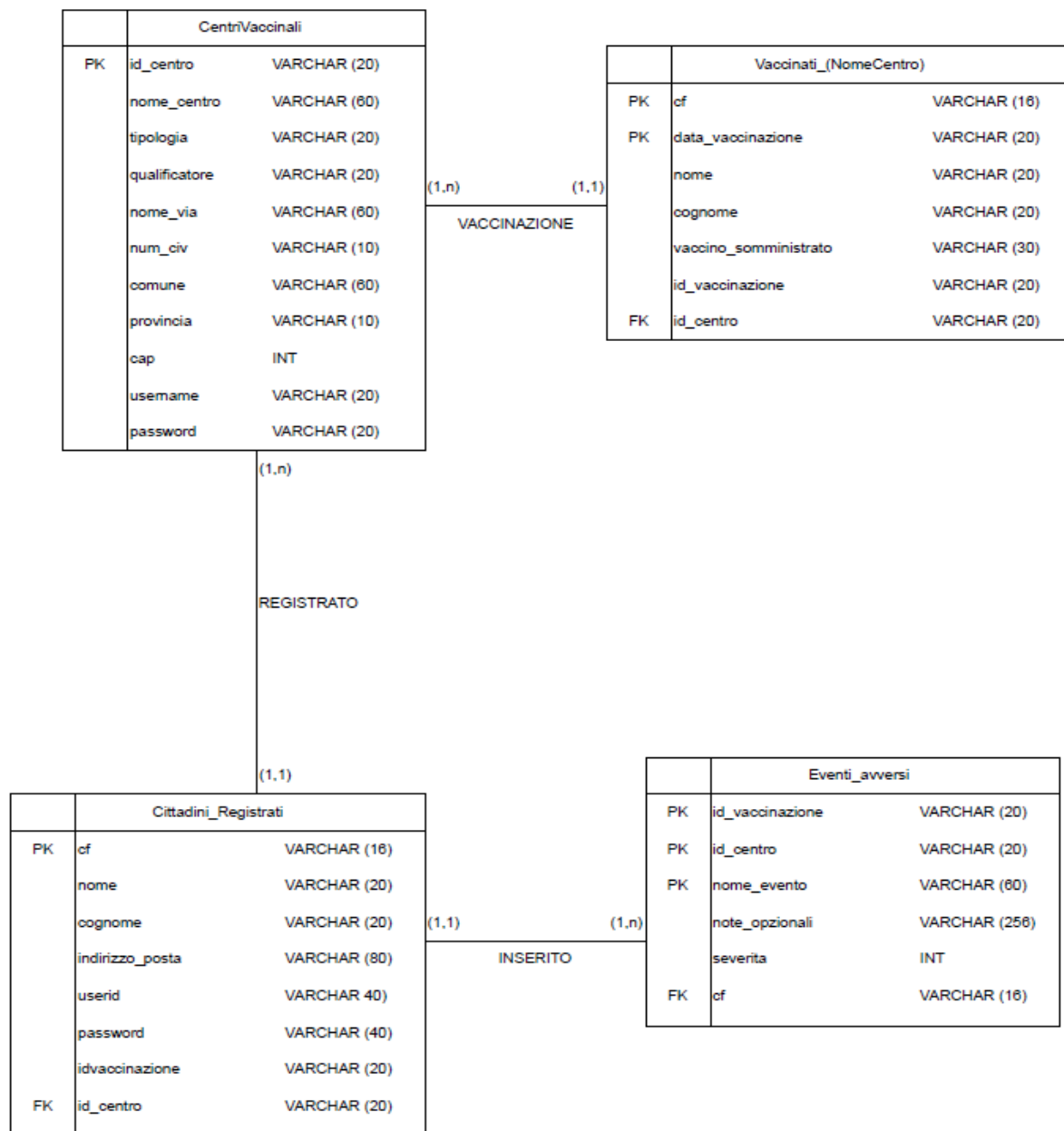
- Cf e Data_vaccinazione primary key
- Nome not null
- Cognome not null
- Vaccino_somministrato not null
- Idvaccinazione unique e not null
- Id_centro foreign key

Eventi_Avversi:

- Id_vaccinazione, id_centro e nome_evento primary key
- Note_opzionali not null
- Severita not null
- Cf foreign key e not null

SCHEMA CONCETTUALE RISTRUTTURATO

- L'indirizzo contiene troppi campi e viene suddiviso in qualificatore, nome via e numero civico



PROGETTAZIONE LOGICA

Centrivaccinali{

Id_centro varchar

Nome_centro varchar

Tipologia varchar

Qualificatore varchar

Nome_via varchar

Num_civ varchar

```

    Comune varchar
    Provincia varchar
    Cap varchar
    Username varchar
    Password varchar
)
Cittadini_registrati(
    Cf varchar
    Nome varchar
    Cognome varchar
    Indirizzo_posta varchar
    Userid varchar
    Password varchar
    Idvaccinazione varchar
    Id_centroCentrivaccinali varchar
)
Vaccinati_(nome_centro)(
    Cf varchar
    Nome varchar
    Cognome varchar
    Data_vaccinazione varchar
    Vaccino_somministrato varchar
    Idvaccinazione varchar
    Id_centroCentrivaccinali varchar
)
Eventi_Aversivi(
    Id_vaccinazione varchar
    Id_centro varchar
    Nome_evento varchar
    Note_opzionali varchar
    Severita int
    CfCittadini_registrati varchar
)

```

NORMALIZZAZIONE

Data la ridotta complessità della struttura del database, non si ritiene necessaria.

PROGETTAZIONE PRATICA

CREAZIONE TABELLE

Centrivaccinali:

```

CREATE TABLE IF NOT EXISTS CentriVaccinali (
\tid_centro VARCHAR(20) PRIMARY KEY,
\tnome_centro VARCHAR(60) NOT NULL,
\ttipologia VARCHAR(20) NOT NULL,
\tqualificatore VARCHAR(20) NOT NULL,
\tnome_via VARCHAR(60) NOT NULL,
\tnum_civ VARCHAR(10) NOT NULL,
\tcomune VARCHAR(60) NOT NULL,
\tprovincia VARCHAR(10) NOT NULL,

```

```

\tcap INT NOT NULL,
\tusername VARCHAR(20) UNIQUE NOT NULL,
\tpassword VARCHAR(20) NOT NULL
);

```

Cittadini_registrati:

```

CREATE TABLE IF NOT EXISTS Cittadini_Registrati (
\tcf VARCHAR(30) PRIMARY KEY,
\tnome VARCHAR(20) NOT NULL,
\tcognome VARCHAR(20) NOT NULL,
\tindirizzo_posta VARCHAR(80) UNIQUE NOT NULL,
\tuserid VARCHAR(40) UNIQUE NOT NULL,
\tpassword VARCHAR(40) NOT NULL,
\tidvaccinazione VARCHAR(20) UNIQUE,
\tid_centro VARCHAR(20) NOT NULL,
\tFOREIGN KEY (id_centro) REFERENCES CentriVaccinali(id_centro)
);

```

Vaccinati_(nome_centro):

```

CREATE TABLE IF NOT EXISTS Vaccinati_"+nome+" (\n" +
\tcf VARCHAR(30),\n" +
\tnome VARCHAR(20) NOT NULL,\n" +
\tcognome VARCHAR(20) NOT NULL,\n" +
\tdata_vaccinazione VARCHAR(20) ,\n" +
\tvaccino_somministrato VARCHAR(30) NOT NULL,\n" +
\tidvaccinazione VARCHAR(20) UNIQUE NOT NULL,\n" +
\tid centro VARCHAR(20) NOT NULL,\n" +
\tPRIMARY KEY(cf,data_vaccinazione),\n" +
\tFOREIGN KEY (id_centro) REFERENCES CentriVaccinali(id_centro)\n" +
");

```

Eventi_avversi:

```

CREATE TABLE IF NOT EXISTS Eventi_Avversi (
\tid_vaccinazione VARCHAR(20) NOT NULL,
\tid_centro VARCHAR(60) NOT NULL,
\tnome_evento VARCHAR(60) NOT NULL,
\tnote_opzionali VARCHAR(256),
\tseverita INT NOT NULL,
\tcf VARCHAR(16) NOT NULL,
\tPRIMARY KEY(id_vaccinazione, id_centro, nome_evento),
\tFOREIGN KEY (cf) REFERENCES Cittadini_Registrati(cf)
);

```

OPERAZIONI DI INSERT

Aggiungi centro vaccinale:

```

"INSERT INTO CentriVaccinali(id_centro, nome_centro,tipologia,qualificatore,
nome_via,num_civ, comune, provincia, cap, username, password ) VALUES
(?,?,?,?,?,?,?,?,?,?)";

```

Aggiungi cittadino registrato:

```

"INSERT INTO Cittadini_Registrati VALUES (?,?,?,?, ?, ?, ?, ?, ?)";

```

Aggiungi vaccinato:

```
"INSERT INTO Vaccinati " + Utility.getNameForQuery(nome).toLowerCase()  
+ " VALUES (?, ?, ?, ?, ?, ?, ?)";
```

Aggiungi evento avverso:

```
"INSERT INTO Eventi_Avversi (id_vaccinazione, id_centro, nome_evento, severita,  
note_opzionali, cf) VALUES (?, ?, ?, ?, ?, ?)";
```

QUERY

Ricerca centro vaccinale:

```
"SELECT * FROM CentriVaccinali WHERE nome_centro=? OR comune=? OR tipologia=? ";
```

Controllo nome centro vaccinale:

```
"SELECT nome_centro FROM CentriVaccinali WHERE nome_centro = ?";
```

Controllo username centro vaccinale:

```
"SELECT nome_centro FROM CentriVaccinali WHERE username = ?";
```

Controllo username centro vaccinale (restituisce lo username):

```
"SELECT username FROM CentriVaccinali WHERE username = ?";
```

Controllo login centro vaccinale:

```
"SELECT username, password FROM CentriVaccinali WHERE username = ? AND password =  
?";
```

Controllo cf cittadino:

```
"SELECT cf FROM Vaccinati " + Utility.getNameForQuery(centro).toLowerCase()  
+ " WHERE cf = ?";
```

Controllo id_centro:

```
"SELECT nome_centro FROM CentriVaccinali WHERE id_centro=?";
```

Controllo idvaccinazione:

```
"SELECT idvaccinazione from Vaccinati " + centro + " WHERE idvaccinazione = ?";
```

Conteggio del numero di centri vaccinali:

```
"SELECT COUNT(*) AS count centri FROM CentriVaccinali";
```

Restituzione del nome e del cognome del cittadino:

```
"SELECT nome, cognome FROM Cittadini Registrati WHERE cf = ?";
```

Controllo dello user e del cf del cittadino:

```
"SELECT * FROM Cittadini Registrati WHERE userid = ? and cf =?";
```

Controllo dello user e della password del cittadino:

```
"SELECT userid, password FROM Cittadini_Registrati WHERE userid = ? AND password =  
?";
```

Controllo user del cittadino:

```
"SELECT userid FROM Cittadini_Registrati WHERE userid = ?";
```

Restituzione cf del cittadino:

```
"SELECT cf FROM Cittadini_Registrati WHERE userid = ?";
```

Controllo cf del cittadino:

```
"SELECT cf FROM Cittadini_Registrati WHERE cf = ?";
```

Aggiornamento dell'idvaccinazione:

```
"UPDATE Cittadini_Registrati SET idvaccinazione = ? WHERE cf = ?";
```

Controllo idvaccinazione:

```
"SELECT idvaccinazione FROM Cittadini_Registrati WHERE idvaccinazione = ?";
```

Restituzione idvaccinazione:

```
"SELECT idvaccinazione , id_centro FROM Cittadini_Registrati WHERE cf = ?";
```

Conteggio cittadini vaccinati:

```
"SELECT COUNT(idvaccinazione) AS count_vaccinazioni FROM Cittadini_Registrati WHERE (idvaccinazione IS NOT NULL)";
```

Conteggio numero segnalazioni per un centro vaccinale:

```
"SELECT count(id_centro) AS count_segnalazioni FROM eventi_avversi WHERE id_centro = ?";
```

Calcolo valore medio dell'intensità dell'evento:

```
"SELECT avg(severita) AS media_valori FROM Eventi_Avversi WHERE id_centro = ? AND nome_evento = ?";
```

Controllo se idvaccinazione è già presente (per gli eventi avversi):

```
"SELECT id_vaccinazione FROM Eventi_Avversi WHERE id_vaccinazione = ?";
```

Controllo se esiste centro vaccinale con quel nome:

```
"SELECT id_centro FROM CentriVaccinali WHERE nome_centro=?";
```

IMPLEMENTAZIONE

FUNZIONALITA' AGGIUNTIVE

- Calendario: Siccome la libreria *Java.Swing* non mette a disposizione un'interfaccia grafica per il calendario, abbiamo deciso di utilizzare una libreria esterna che mettesse a disposizione un'interfaccia soddisfacente.
- Menù laterale: Pur non essendo una funzionalità aggiuntiva vera e propria, abbiamo comunque optato per la creazione di un piccolo menù laterale nelle pagine dell'utente loggato.
- Visualizzazione statistiche: Mostra nella Home Page il numero di centri vaccinali esistenti nell'applicazione e il quantitativo di persone che si sono già vaccinate.

PROBLEMI RISCONTRATI

Pur non essendo strettamente necessario è però utile aprire due finestre del terminale per la corretta esecuzione dell'applicazione. Questo può essere ovviato utilizzando uno script cmd/bat per Windows o uno script sh per Linux.