

Varese

Autori:

Aleksandar Evgeniev Ivanov

Berti Davide

Mazza Serghej

Rizzi Silvio

MANUALE TECNICO

INDICE:

- LINGUAGGI UTILIZZATI p 3
- BREVE DESCRIZIONE p 3
- STRUTTURA p 4
 - cartella Resources p 4
 - Package centrivaccinali p 5
 - Package cittadini p 6
 - Package server p 7
 - Package common p 8
 - Package DBManagement p 9
- DIAGRAMMI p 10
 - Package centrivaccinali p 10
 - Package cittadini p 11
 - Package server p 12
 - Package common p 13
 - Package DBManagement p 14
 - DATABASE p 15
- JAVADOC p 17
 - JAVADOC..... p 17

LINGUAGGI UTILIZZATI

Lo sviluppo di questa applicazione ha previsto l'utilizzo dei linguaggi riportati nella tabella seguente:

Nome del linguaggio	Ambito di utilizzo
Java	core dell'applicazione
SQL	gestione della base di dati
FXML	definizione delle interfacce grafiche

Tabella 1. Lista dei linguaggi utilizzati nello sviluppo dell'applicazione

BREVE DESCRIZIONE

L'applicazione sviluppata prevede l'interazione tra il lato client, che è rappresentato da centri vaccinali e cittadini, e il lato server, che offre i servizi necessari per il trasferimento delle informazioni fornite dai client nella base di dati associata all'applicazione.

L'architettura client-server è stata implementata utilizzando Java Remote Method Invocation (Java RMI), una API core del linguaggio che ha permesso la semplificazione delle logiche di connessione e l'ottimizzazione del multithreading.

Le interfacce grafiche sono state implementate utilizzando JavaFX, un framework del linguaggio che ha permesso la separazione tra la logica di progettazione grafica e la logica di controllo determinando un miglioramento nella leggibilità e nella manutenibilità del codice prodotto.

La gestione della base di dati e l'ottimizzazione delle interrogazioni è stata affidata PostgreSQL.

STRUTTURA

CARTELLA RESOURCES

La cartella resources contiene i file in formato FXML in cui sono definite le GUI utilizzate dall'applicazione. Di seguito, sono elencati i file contenuti in questa cartella:

- CvCerca
- CvConnection
- CvHome
- CvRegCentro
- CvRegVaccinato
- CercaCittadini
- ConnectionCittadini
- HomeCittadini
- LoginCittadini
- RegistrazioneCittadini
- SegnalazioneCittadini
- VisualizzaCittadini

PACKAGE CENTRIVACCINALI

Il package centrivaccinali contiene le classi necessarie per le operazioni di registrazione di centri vaccinali e cittadini vaccinati. Di seguito, sono descritte le classi contenute in questo package:

- **CvMain**: contiene il metodo main che permette l'avvio dell'applicazione e il lancio del pannello di gestione della connessione con il server.
- **ServerConnection**: contiene il riferimento a un oggetto che permette di effettuare il collegamento con il server indicato dall'utente.
- **CvCntrlConnection**: rappresenta il controller dell'interfaccia grafica del pannello di gestione della connessione con il server tramite cui è possibile specificare un indirizzo IP, verificare lo stato di connessione e lanciare la pagina iniziale; la GUI associata a questo controller è definita in CvConnection.fxml.
- **CvHomePage** e **CvCntrlHome**: rappresentano rispettivamente implementazione e controller dell'interfaccia grafica della pagina iniziale tramite cui è possibile decidere se proseguire con la registrazione di un nuovo centro vaccinale o di un nuovo cittadino vaccinato; la GUI associata a questo controller è definita in CvHome.fxml.
- **CvRegCentro** e **CvCntrlCentro**: rappresentano rispettivamente implementazione e controller dell'interfaccia grafica del modulo per la registrazione di un nuovo centro vaccinale; la GUI associata a questo controller è definita in CvRegCentro.fxml.
- **CvRegVaccinato** e **CvCntrlVaccinato**: rappresentano rispettivamente implementazione e controller dell'interfaccia grafica del modulo per la registrazione di un nuovo cittadino vaccinato; la GUI associata a questo controller è definita in CvRegVaccinato.fxml.
- **CvCerca** e **CvCntrlCerca**: rappresentano rispettivamente implementazione e controller dell'interfaccia grafica del modulo per la selezione di un centro vaccinale registrato nella base di dati; la GUI associata a questo controller è definita in CvCerca.fxml.
- **CvUtil**: contiene metodi statici di utilità.

Il diagramma delle classi in linguaggio UML di questo package è riportato nella sezione "Diagrammi UML". ([Clicca qui per visualizzare](#))

PACKAGE CITTADINI

Il package cittadini contiene le classi necessarie per le operazioni di registrazione e/o autenticazione di un utente, la ricerca di informazioni su centri vaccinali registrati e la segnalazione di eventi avversi. Di seguito, sono descritte le classi contenute in questo package:

- **ClientGUI:** contiene il metodo main che permette l'avvio dell'applicazione e il lancio del pannello di gestione della connessione con il server.
- **ControllerConnection:** rappresenta il controller dell'interfaccia grafica del pannello di gestione della connessione con il server tramite cui è possibile specificare un indirizzo IP, verificare lo stato di connessione e lanciare la pagina iniziale; la GUI associata a questo controller è definita in ConnectionCittadini.fxml.
- **ControllerHome:** rappresenta il controller dell'interfaccia grafica della pagina iniziale tramite cui è possibile decidere se proseguire con la ricerca di informazioni su centri vaccinali registrati, la segnalazione di eventuali eventi avversi, la registrazione e/o autenticazione di un utente; la GUI associata a questo controller è definita in HomeCittadini.fxml.
- **ControllerCerca:** rappresenta il controller dell'interfaccia grafica del modulo per la ricerca di un centro vaccinale registrato sulla base di dati associata all'applicazione; la GUI associata a questo controller è definita in CercaCittadini.fxml.
- **ControllerVisualizza:** rappresenta il controller dell'interfaccia grafica del modulo per la visualizzazione delle informazioni disponibili per il centro vaccinale selezionato; la GUI associata a questo controller è VisualizzaCittadini.fxml.
- **ControllerRegistrazione:** rappresenta il controller dell'interfaccia grafica del modulo per la registrazione di un nuovo utente sulla piattaforma; la GUI associata a questo controller è definita in RegistrazioneCittadini.fxml.
- **ControllerLogin:** rappresenta il controller dell'interfaccia grafica del modulo per l'autenticazione di un utente registrato sulla piattaforma; la GUI associata a questo controller è definita in LoginCittadini.fxml.

- **ControllerSegnalazioni:** rappresenta il controller dell'interfaccia grafica del modulo per la segnalazione di eventi avversi in risposta alla somministrazione del vaccino; la GUI associata a questo controller è definita in SegnalazioneCittadini.fxml.

Il diagramma delle classi in linguaggio UML di questo package è riportato nella sezione "Diagrammi UML". ([Clicca qui per visualizzare](#))

PACKAGE SERVER

Il package server contiene le interfacce e le classi necessarie per la corretta gestione delle connessioni con il server. Di seguito, sono descritte le classi e le interfacce contenute in questo package:

- **ServerInterface:** rappresenta l'interfaccia in cui è definita la segnatura dei metodi offerti dal server ai client dell'applicazione.
- **ServerImpl:** rappresenta la classe in cui sono implementati i metodi definiti nell'interfaccia citata al punto precedente.

Il diagramma delle classi in linguaggio UML di questo package è riportato nella sezione "Diagrammi UML". ([Clicca qui per visualizzare](#))

PACKAGE COMMON

Il package common contiene le classi comuni a server e client che sono necessarie alla rappresentazione e alla gestione degli attori presenti nel sistema. Di seguito, sono descritte le classi contenute in questo package:

- **CentroVaccinale**: descrive un centro vaccinale.
- **Vaccinato**: descrive un utente vaccinato registrato nella sezione "centri vaccinali" dell'applicazione.
- **Cittadino**: descrive un utente vaccinato registrato nella sezione "cittadini" dell'applicazione.
- **ReportEventoAvverso**: descrive un report degli eventi avversi che si sono verificati in seguito alla somministrazione del vaccino presso un centro vaccinale.
- **Segnalazione**: descrive una segnalazione effettuata da un utente riguardo a un evento avverso percepito.
- **TipologiaCentro**: enumerativo delle tipologie di centro vaccinale.
- **Vaccino**: enumerativo delle tipologie di vaccino.

Il diagramma delle classi in linguaggio UML di questo package è riportato nella sezione "Diagrammi UML". ([Clicca qui per visualizzare](#))

PACKAGE DBMANAGEMENT

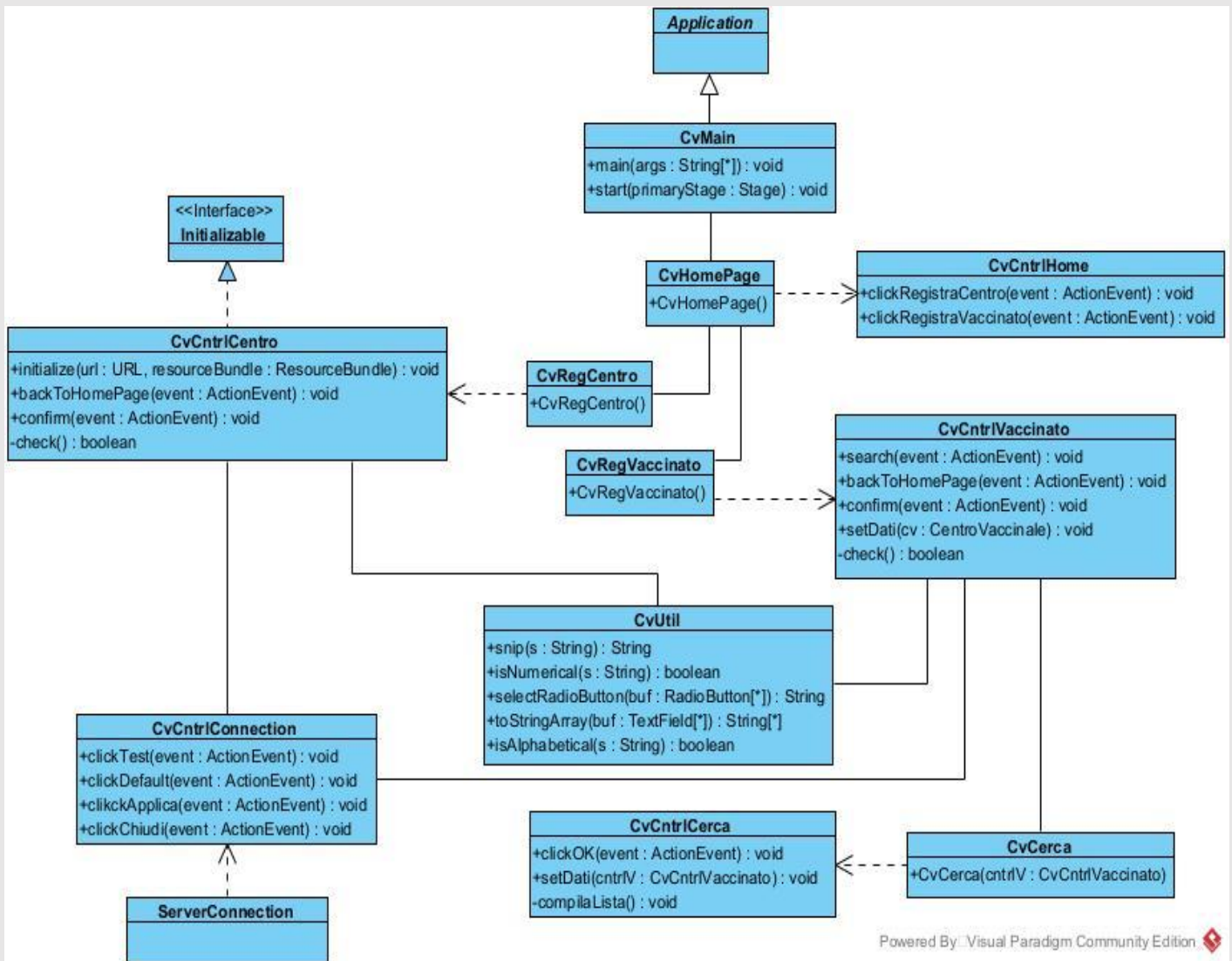
Il package DBManagement contiene le classi necessarie per la connessione alla base di dati e l'esecuzione delle interrogazioni. Di seguito, sono descritte le classi contenute in questo package:

- **DBManager**: offre servizi di base per la connessione e l'esecuzione delle interrogazioni su una base di dati.
- **DBVaccinazioniManagement**: offre servizi specifici per la gestione della base di dati associata all'applicazione.
- **DBTester**: contiene un metodo main che permette di effettuare un test preliminare per verificare il corretto funzionamento della base di dati dell'applicazione.

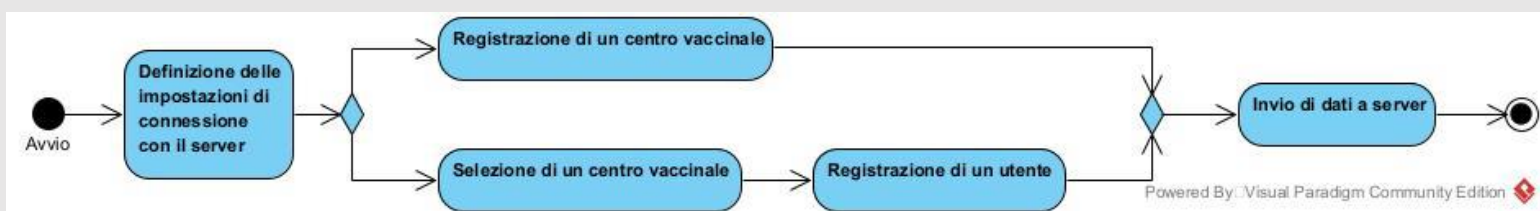
Il diagramma delle classi in linguaggio UML di questo package è riportato nella sezione "Diagrammi UML". ([Clicca qui per visualizzare](#))

DIAGRAMMI UML

Package CENTRIVACCINALI

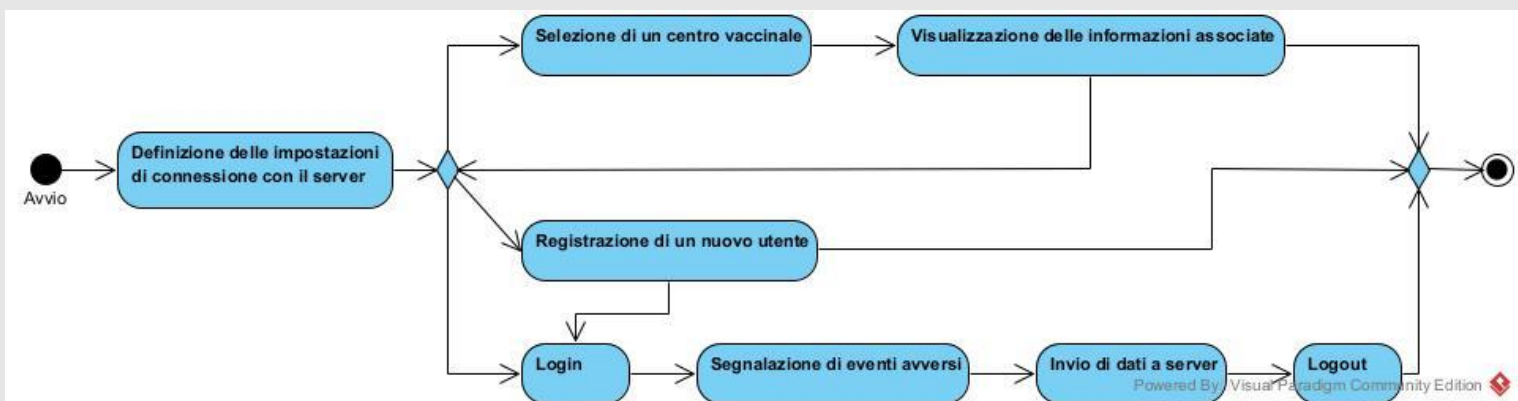
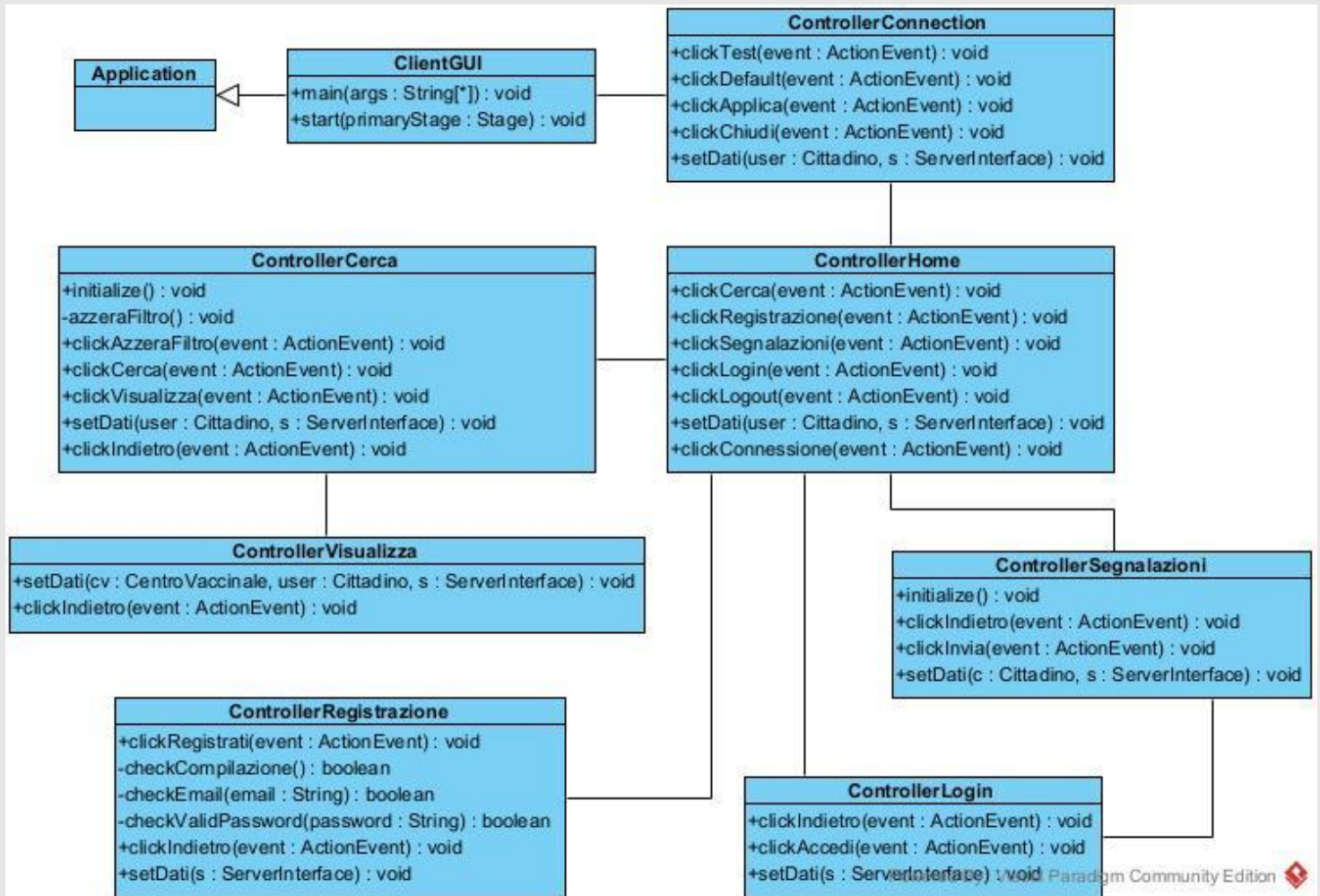


Powered By: Visual Paradigm Community Edition

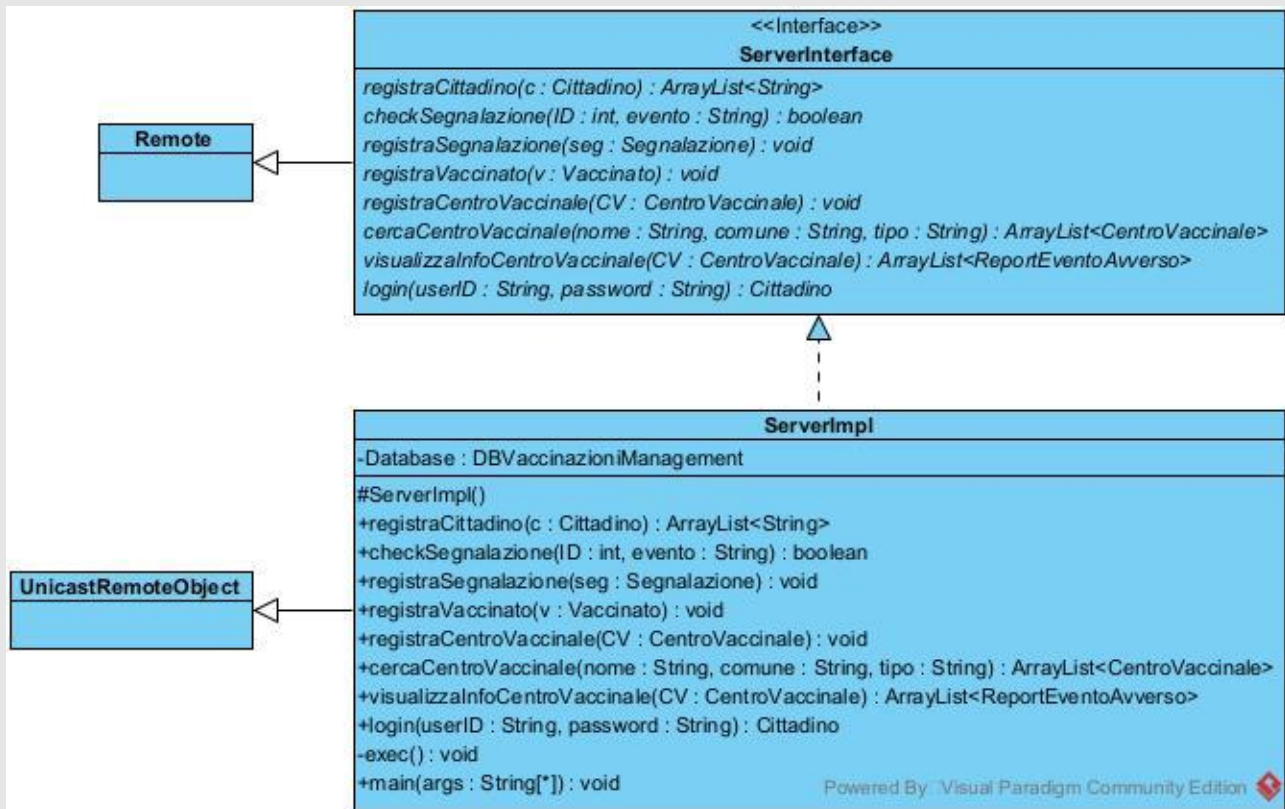


Powered By: Visual Paradigm Community Edition

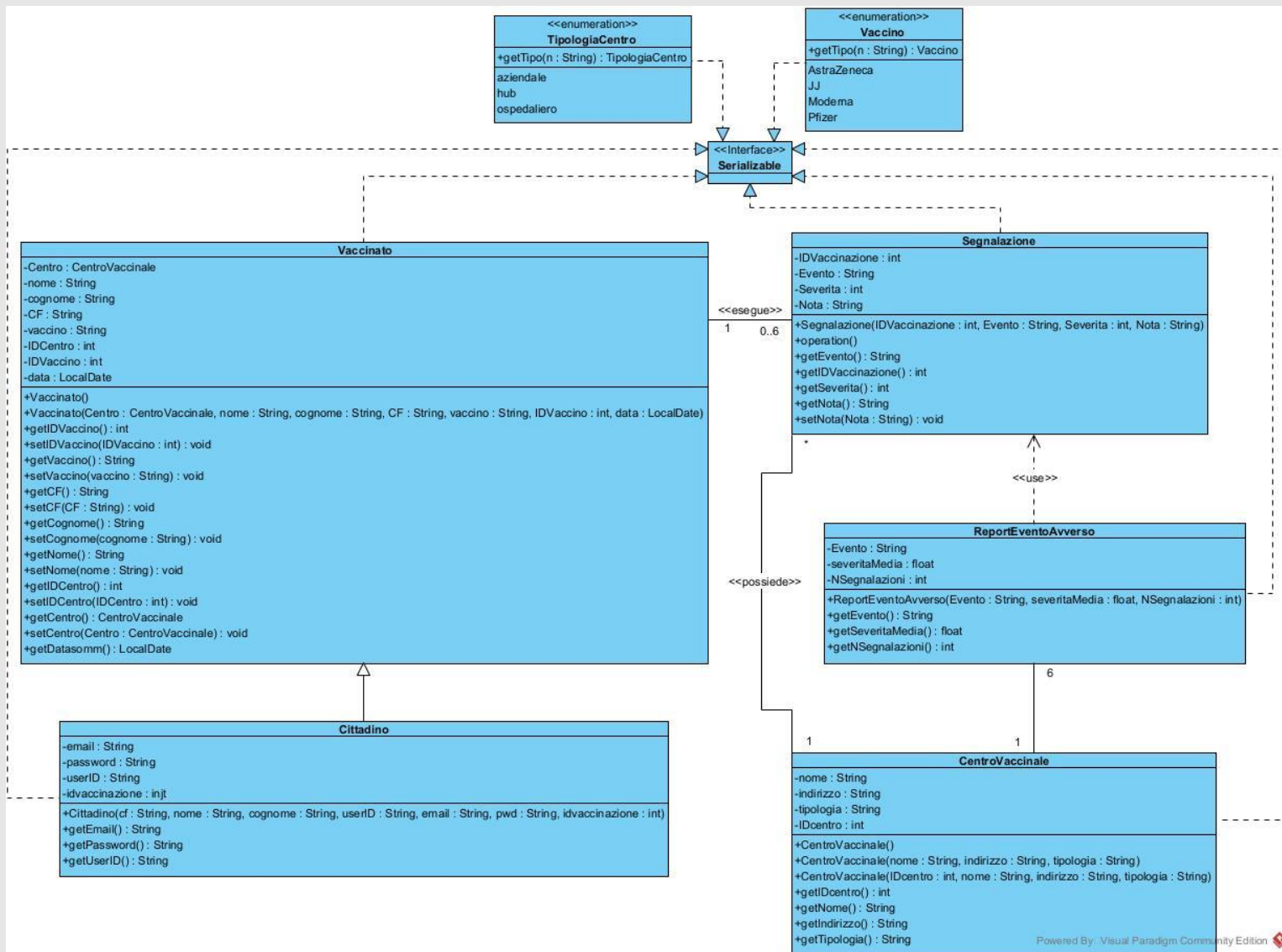
Package CITTADINI



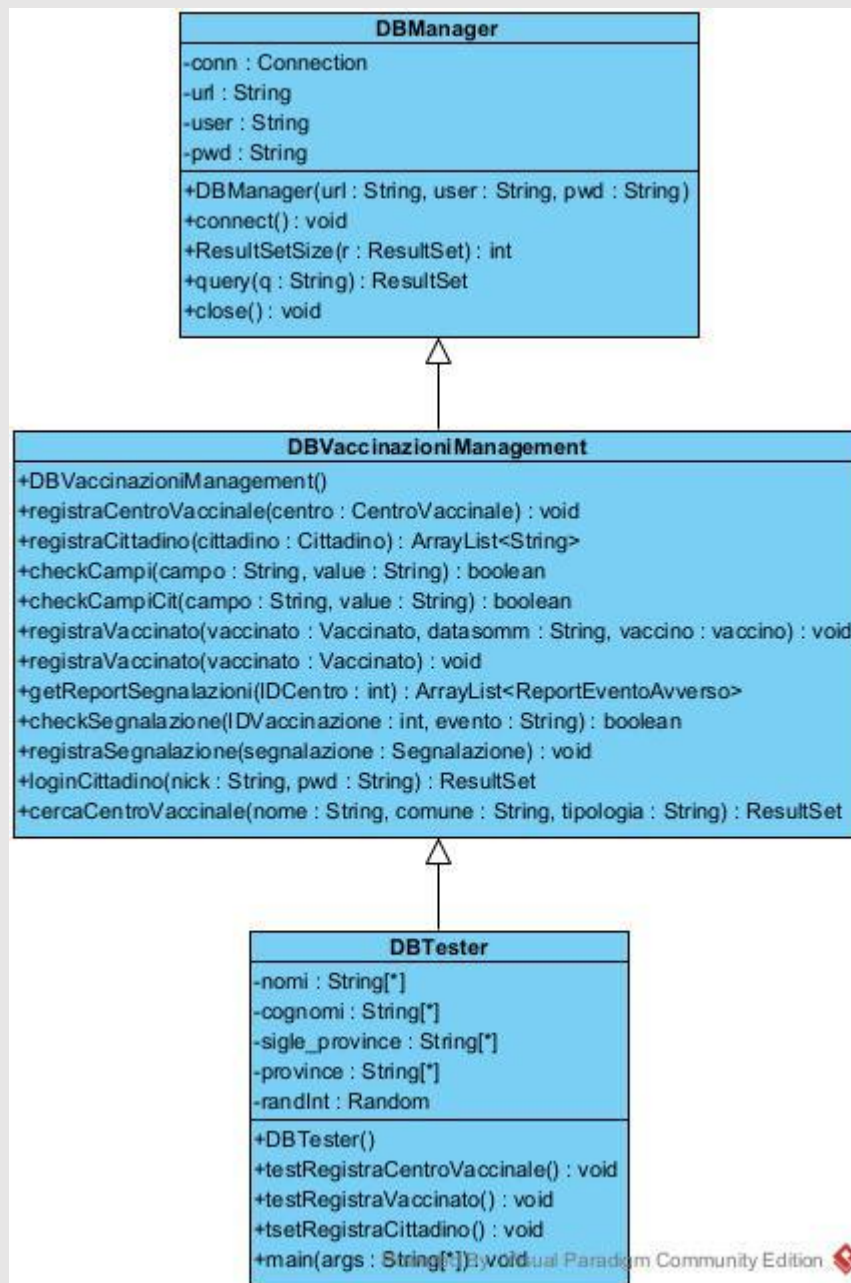
Package SERVER



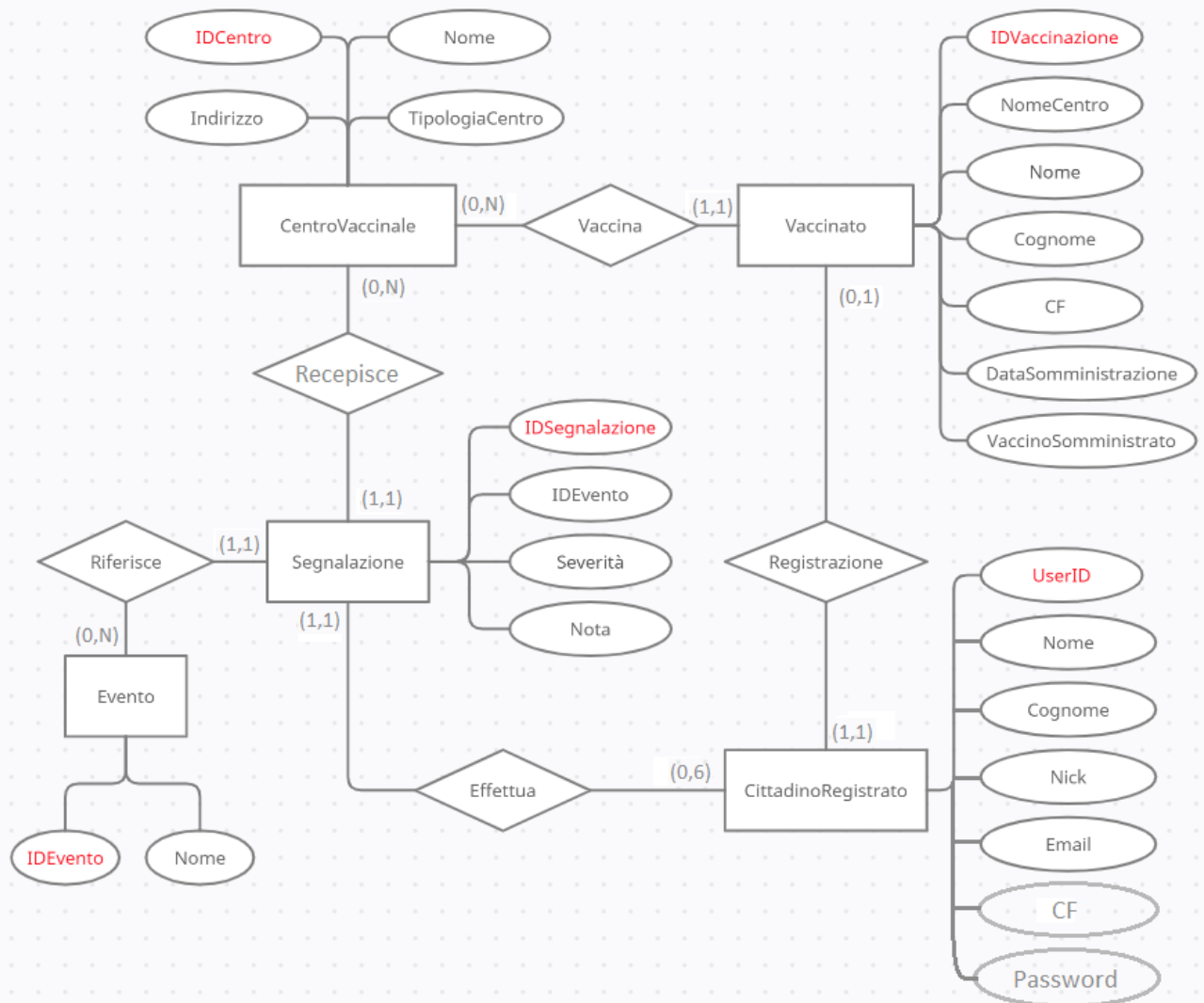
Package COMMON



Package DBMANAGEMENT



DATABASE



CentroVaccinale (IDCentro, Nome, Indirizzo, TipologiaCentro)

Vaccinato(IDVaccinazione, NomeCentro, Nome, Cognome, CF, DataSomministrazione, VaccinoSomministrazione, IDCentro^{CentroVaccinale})

CittadinoRegistrato (UserID, Nome, Cognome, Nick, Email, CF, Password, IDVaccinazione^{Vaccinato})

Segnalazione(IDSegnalazione, Severità, Nota, IDVaccinazione^{CittadinoRegistrato}, IDEvento^{Evento})

Evento(IDEvento, Nome)

centrivaccinali	
	idcentro integer NOT NULL nome character varying(60) NOT NULL indirizzo character varying(100) NOT NULL tipologia tipologiacentro NOT NULL

cittadiniregistrati	
	idutente integer NOT NULL nome character varying(30) NOT NULL cognome character varying(30) NOT NULL email character varying(30) NOT NULL password character(64) NOT NULL cf character(16) NOT NULL idcentro integer NOT NULL nick character varying(20) NOT NULL idvaccinazione integer NOT NULL

eventoavverso	
	idevento integer NOT NULL evento character varying(50) NOT NULL

segnalazione	
	idsegnalazione integer NOT NULL idevento integer NOT NULL severita integer NOT NULL nota character varying(256) idvaccinazione integer NOT NULL

vaccinati	
	idcentro integer NOT NULL nome character varying(30) NOT NULL cognome character varying(30) NOT NULL cf character(16) NOT NULL datasomministrazione date NOT NULL vaccinosomministrato vaccino idvaccinazione integer NOT NULL nomecentro character varying(60) NOT NULL

JAVADOC

È presente la possibilità di visualizzare in dettaglio tutti le componenti del progetto tramite la funzionalità di JAVADOC.

Per accedere a quest'ultima basta cliccare qui sotto.

-----> *JAVADOC* <-----

Se il collegamento non dovesse funzionare basta aprire il file denominato "index" nella cartella LabB\doc\JavaDOC\index