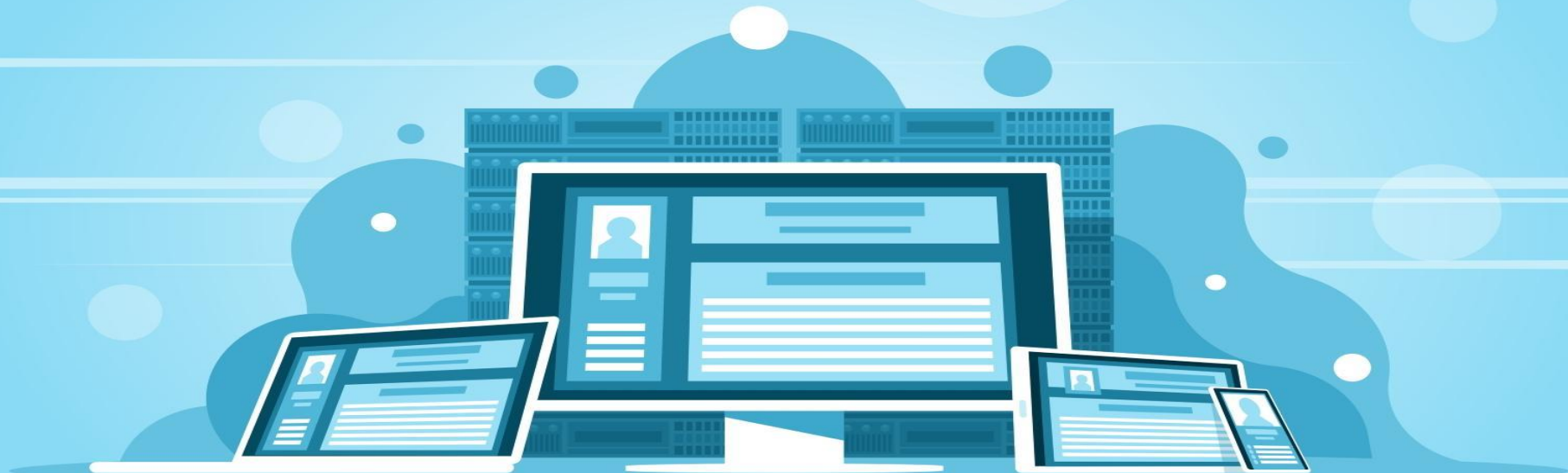


APP MESSAGGISTICA 2023/2024

Prod by:

Davide Rossitto(Programmatore Java)

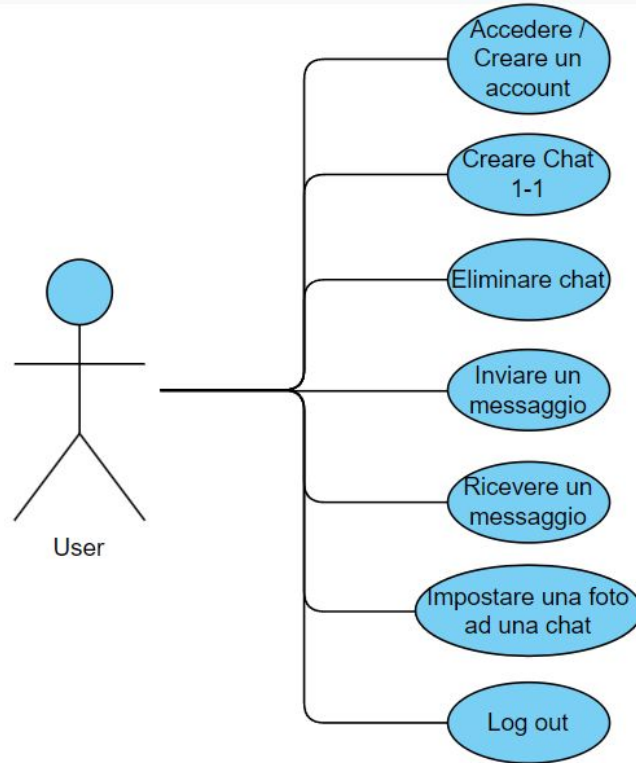
Edoardo Stanzani(Database manager)



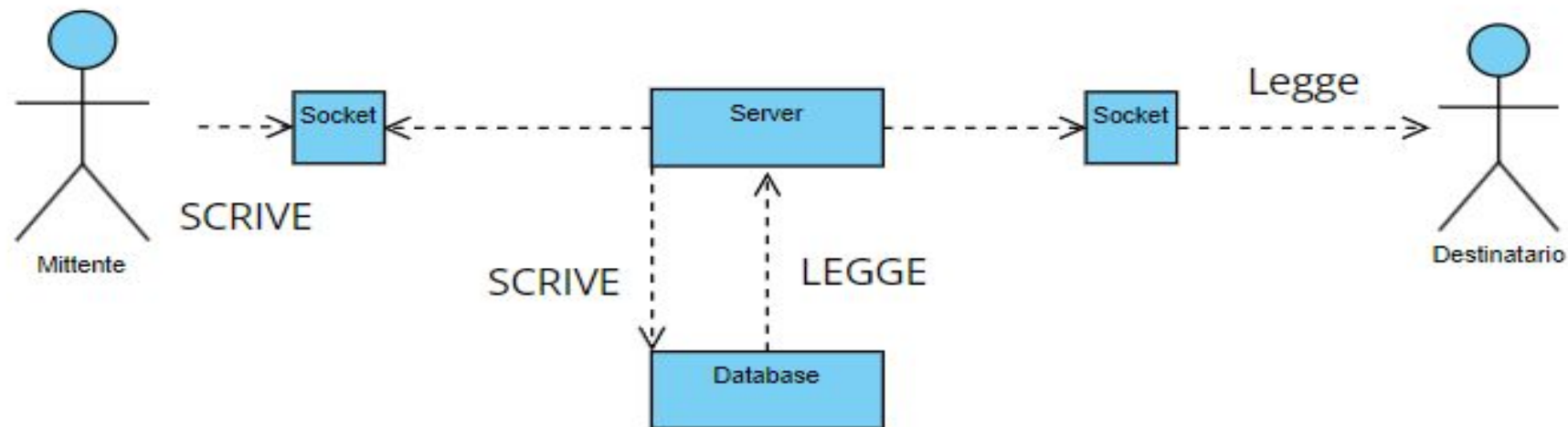
Tecnologie usate

Scene Builder	JAVAFX	JAVA	JDBC	MySQL	Socket TCP
Scene Builder è un software per creare file fxml JavaFX in modo visuale, utilizzando il linguaggio FXML.	JavaFX è una libreria grafica per la creazione di interfacce utente moderne e interattive.	Java è un linguaggio di programmazione ad alto livello, orientato agli oggetti e multi-piattaforma.	JDBC è un'API Java che consente ai programmatori di connettersi e interagire con i database relazionali.	MySQL è un sistema di gestione di database relazionale open-source e ampiamente utilizzato.	Per una comunicazione affidabile ci siamo prestati ad usare le socket che si basano sul protocollo TCP
Front-end	Front-end	Back-end	Back-end	Database	Comunicazione Client-Server

USE CASE USER



Interazione generale



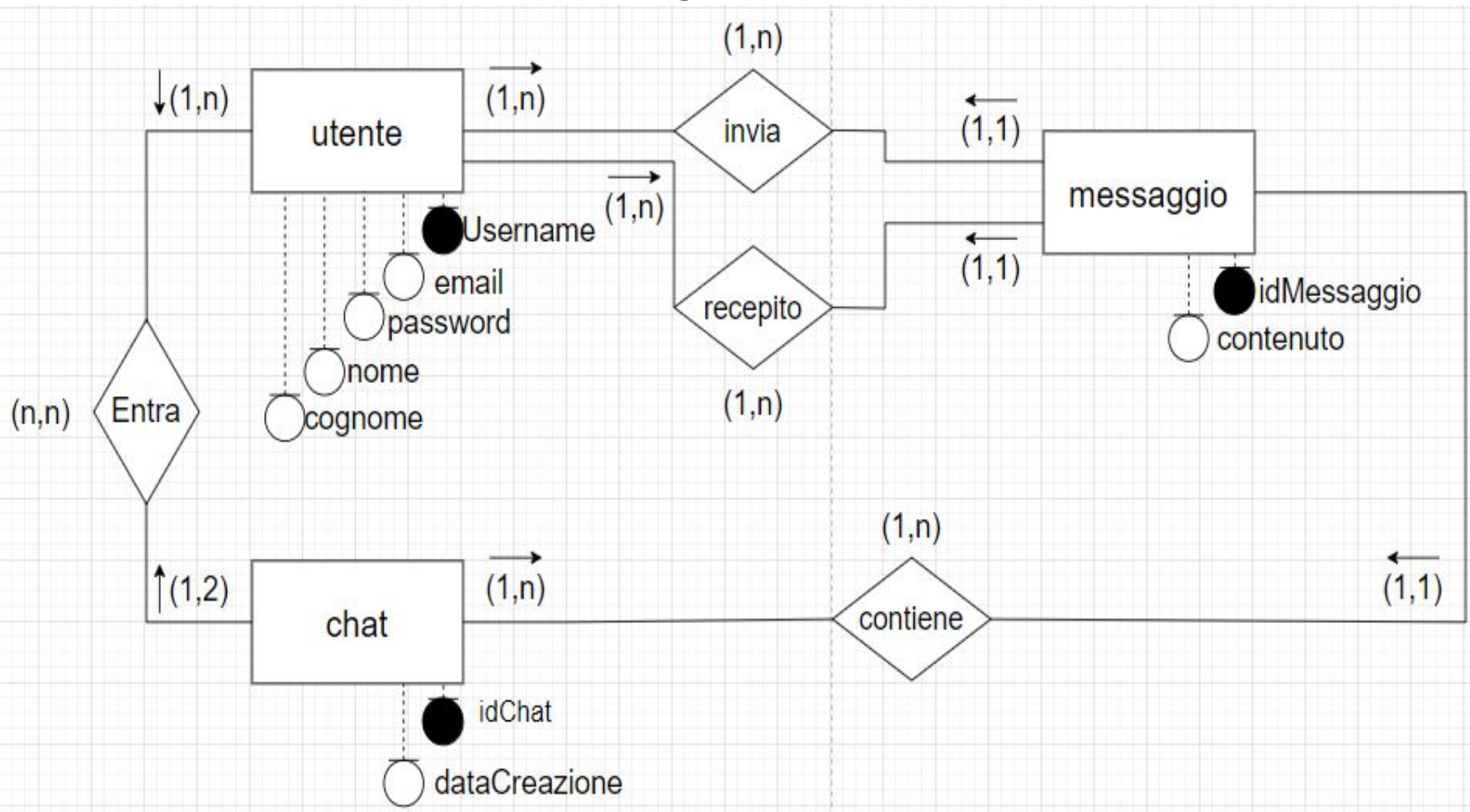
Protocolli lato server

TIPO MESSAGGIO	SERVER	REGEX
<i>Correct Login</i>	true,username,email,nome,cognome	£\$%&
<i>Incorrect Login</i>	false	£\$%&
<i>Messaggio Generico</i>	<u>mittente,destinatario,contenuto,id</u> <u>Messaggio, id Chat</u>	£\$%&
<i>Registrazione effettuata</i>	true	£\$%&
<i>Errore Registrazione generico</i>	error	£\$%&
<i>Errore già esiste user</i>	exist	£\$%&
<i>Risposta se esiste username corretta:</i>	true <u>usernameDestinatario</u>	£\$%&
<i>Risposta se esiste username non corretta</i>	false	£\$%&

Protocolli lato client

TIPO MESSAGGIO	CLIENT	REGEX
<i>Login</i>	username password	£\$%&
<i>Registrazione</i>	username password nome cognome email	£\$%&
<i>Messaggio Generico</i>	mittente destinatario contenuto id Messaggio id Chat	£\$%&
<i>Se esiste uno user</i>	<u>usernameMittente</u> <u>usernameDestinatario</u>	£\$%&

Database - progettazione concettuale



Database - progettazione logica

utente (username, email, password, nome , cognome)

chat (idChat , dataCreazione)

entra(idChat*, attore1*, attore2*)

messaggio(idMessaggio, contenuto, mittente*, destinatario*, idChat*)

entra [attore1] \subseteq utente[username]

entra [attore2] \subseteq utente[username]

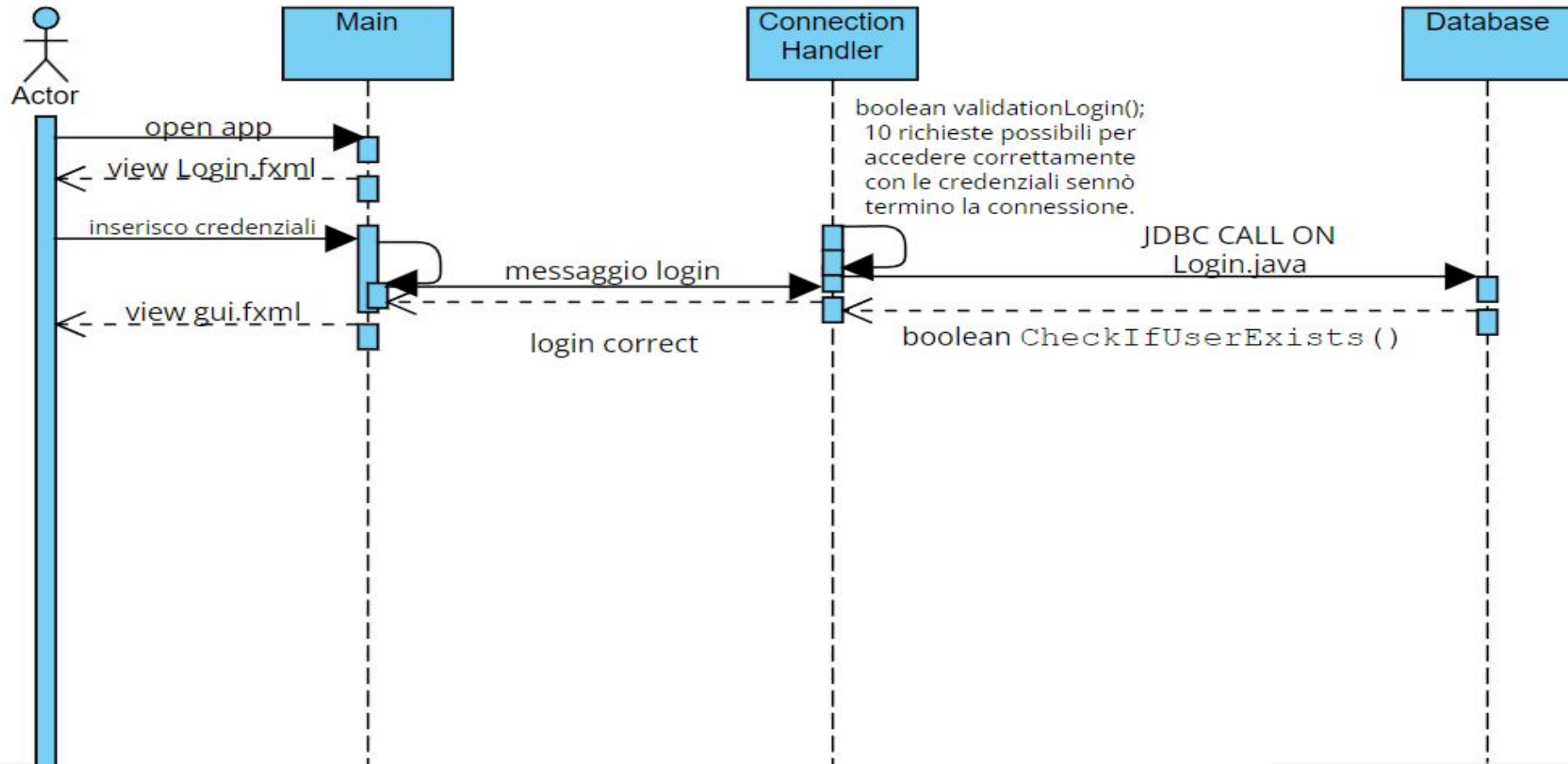
entra[idChat] \subseteq chat[idChat]

messaggio[mittente] \subseteq utente[username]

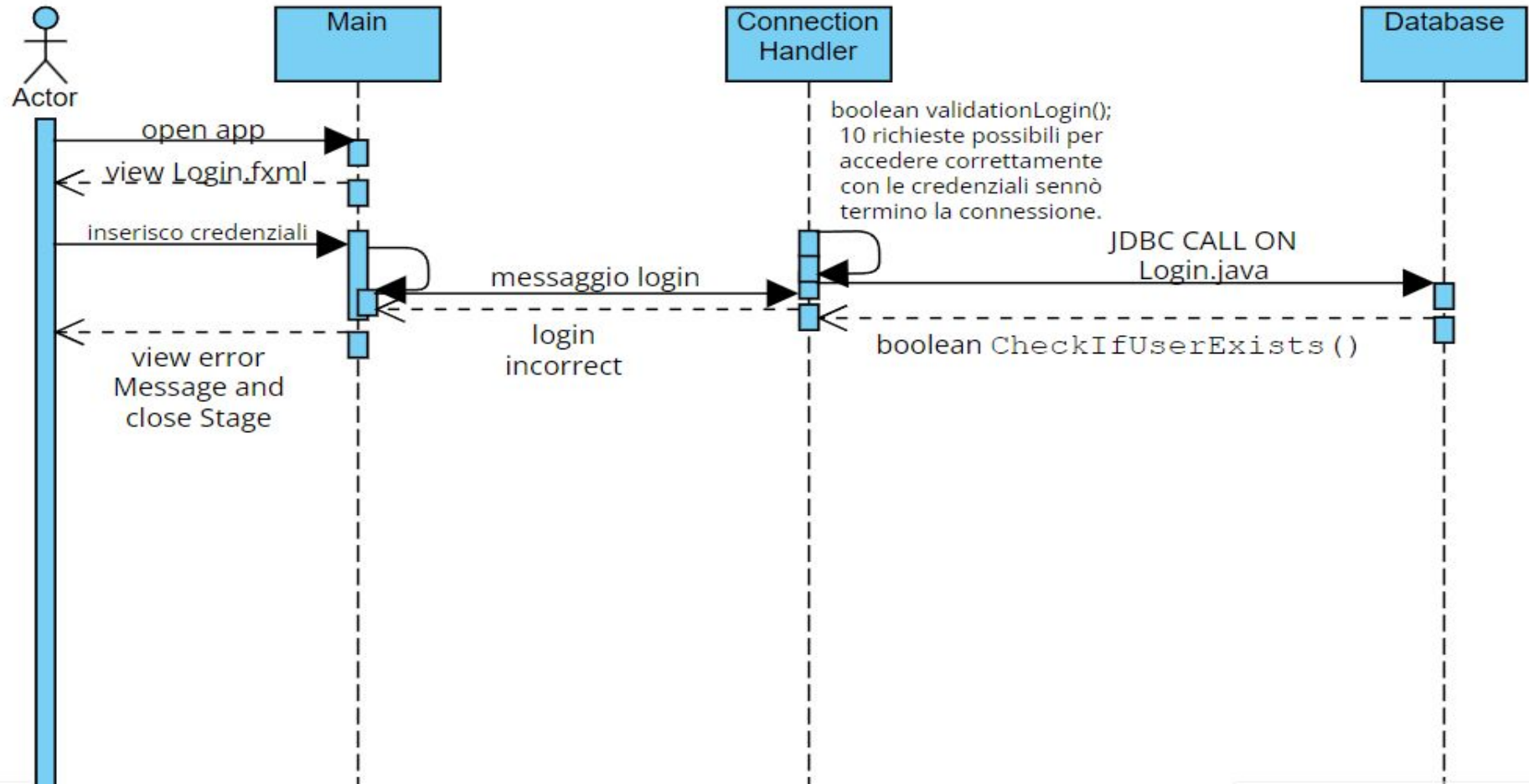
messaggio[destinatario] \subseteq utente[username]

messaggio[idChat] \subseteq chat[idChat]

Login Correct



Login Incorrect



Login



Username

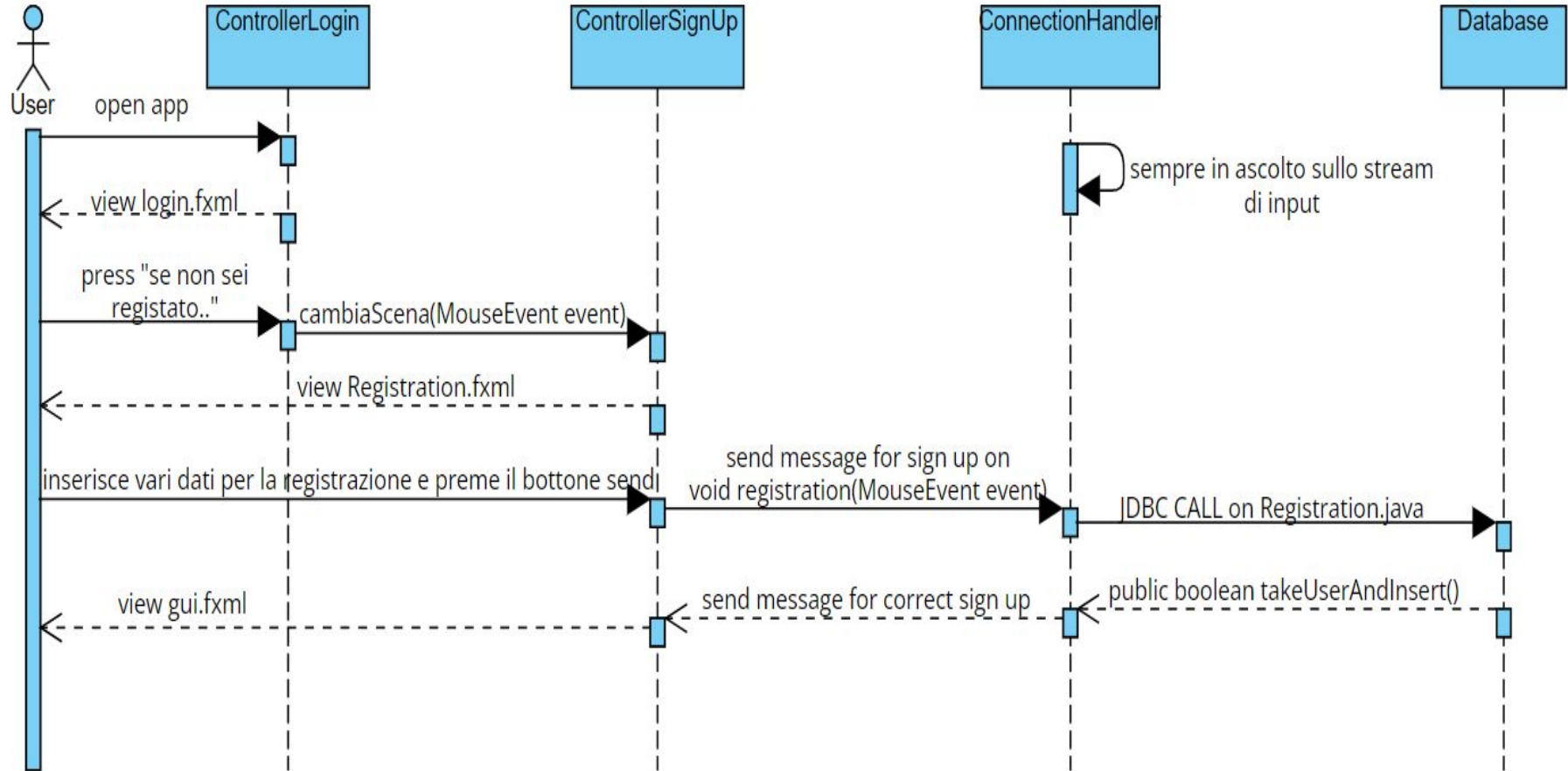


Password

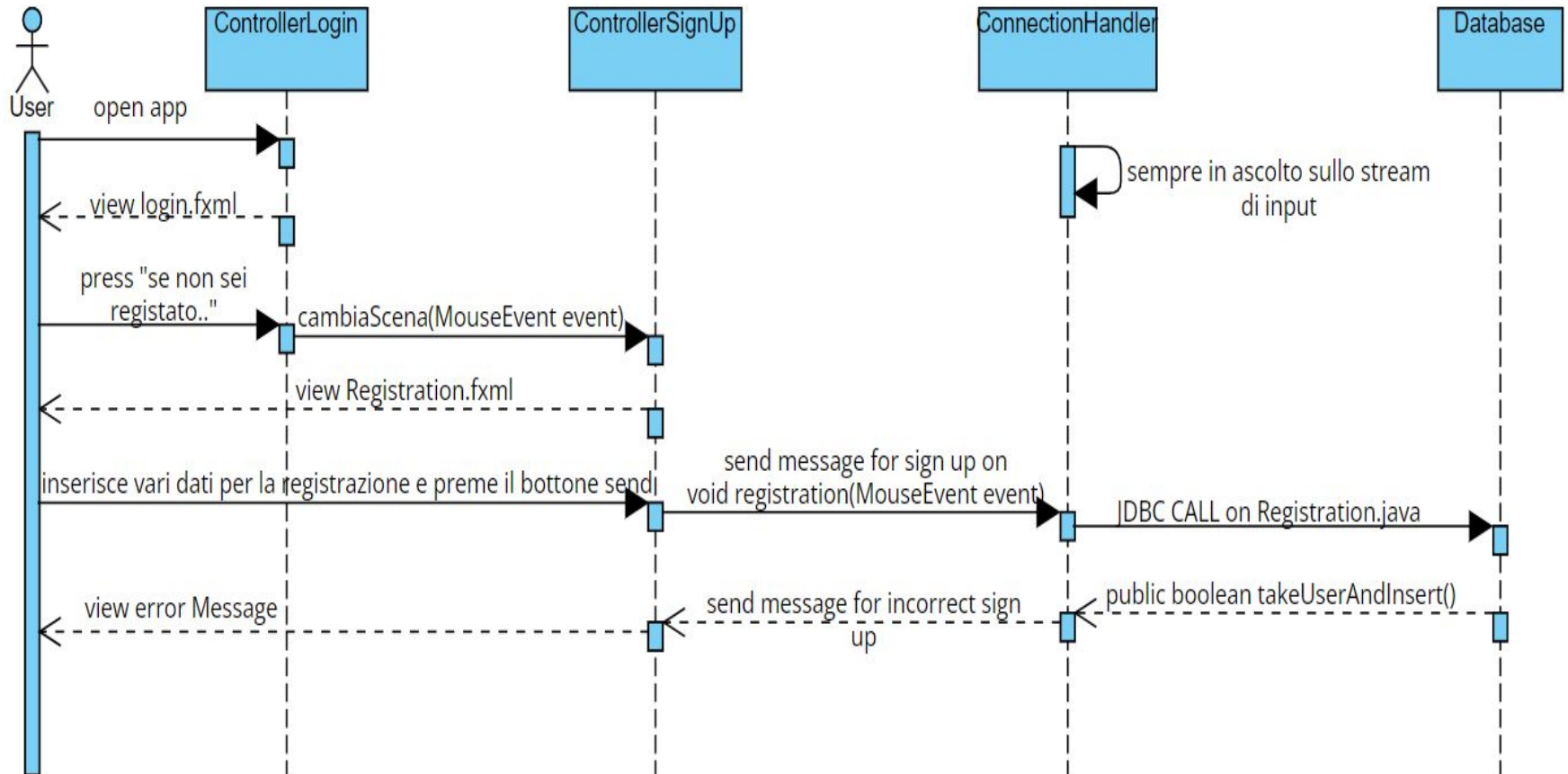
Login

[Se non sei registrato clicca qui](#)

Registrazione corretta



Registrazione non corretta



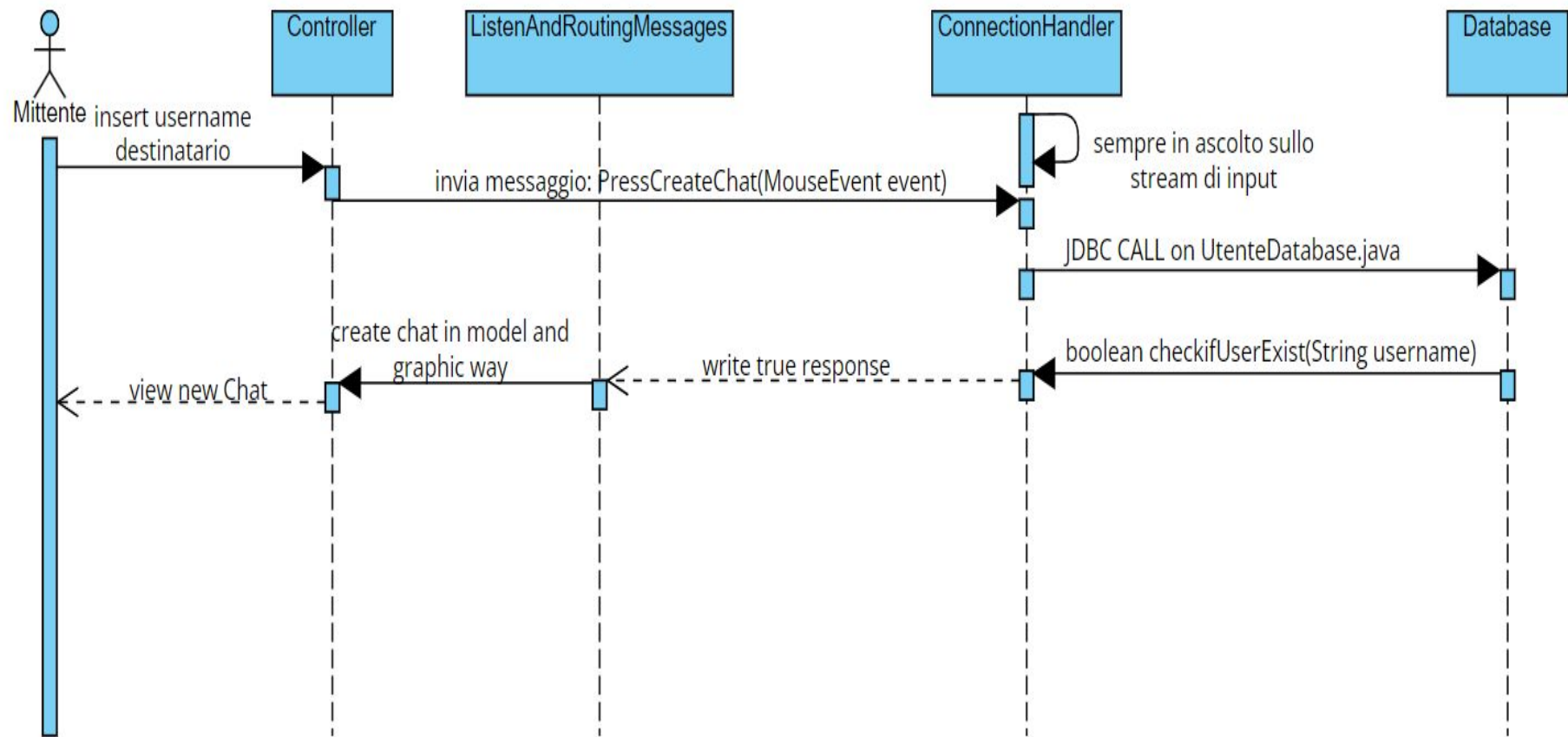
Registrazione



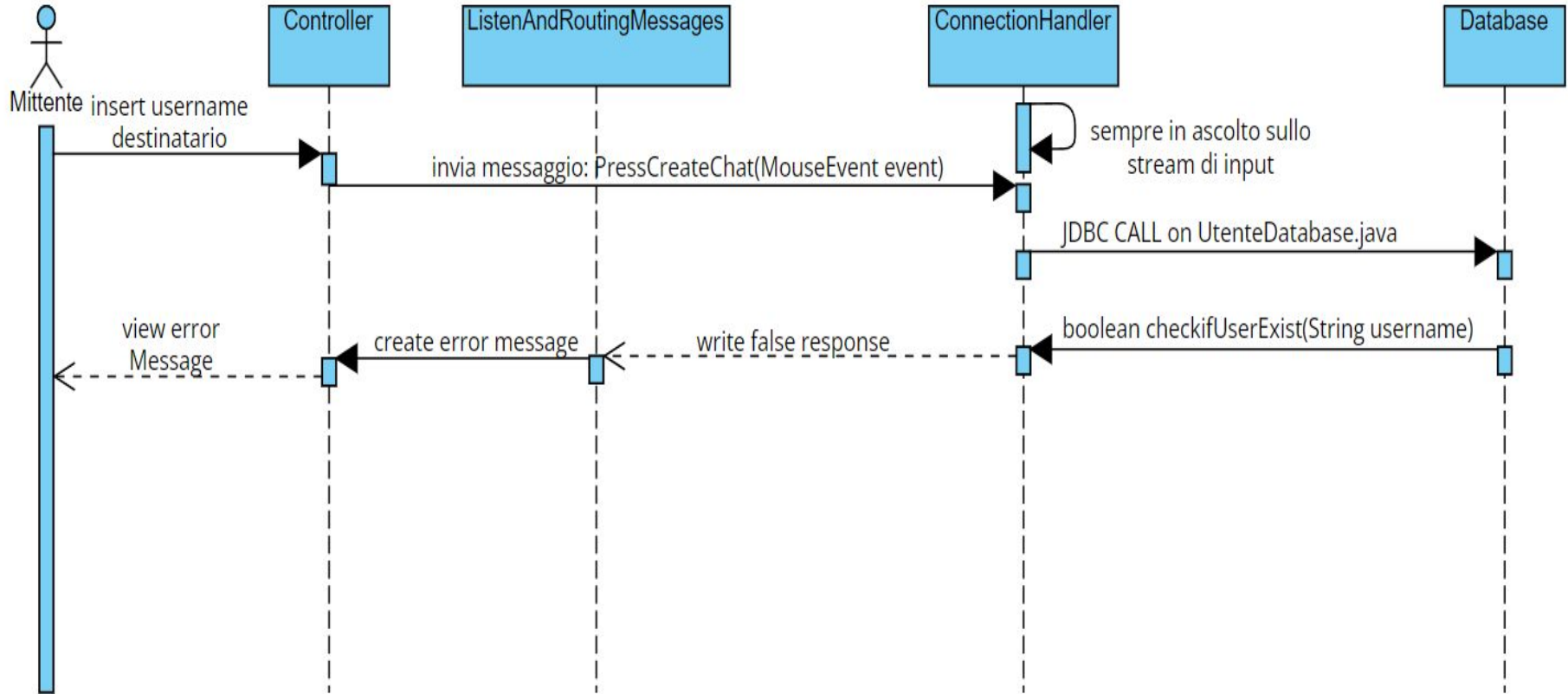
Send

[Se già sei registrato clicca qui](#)

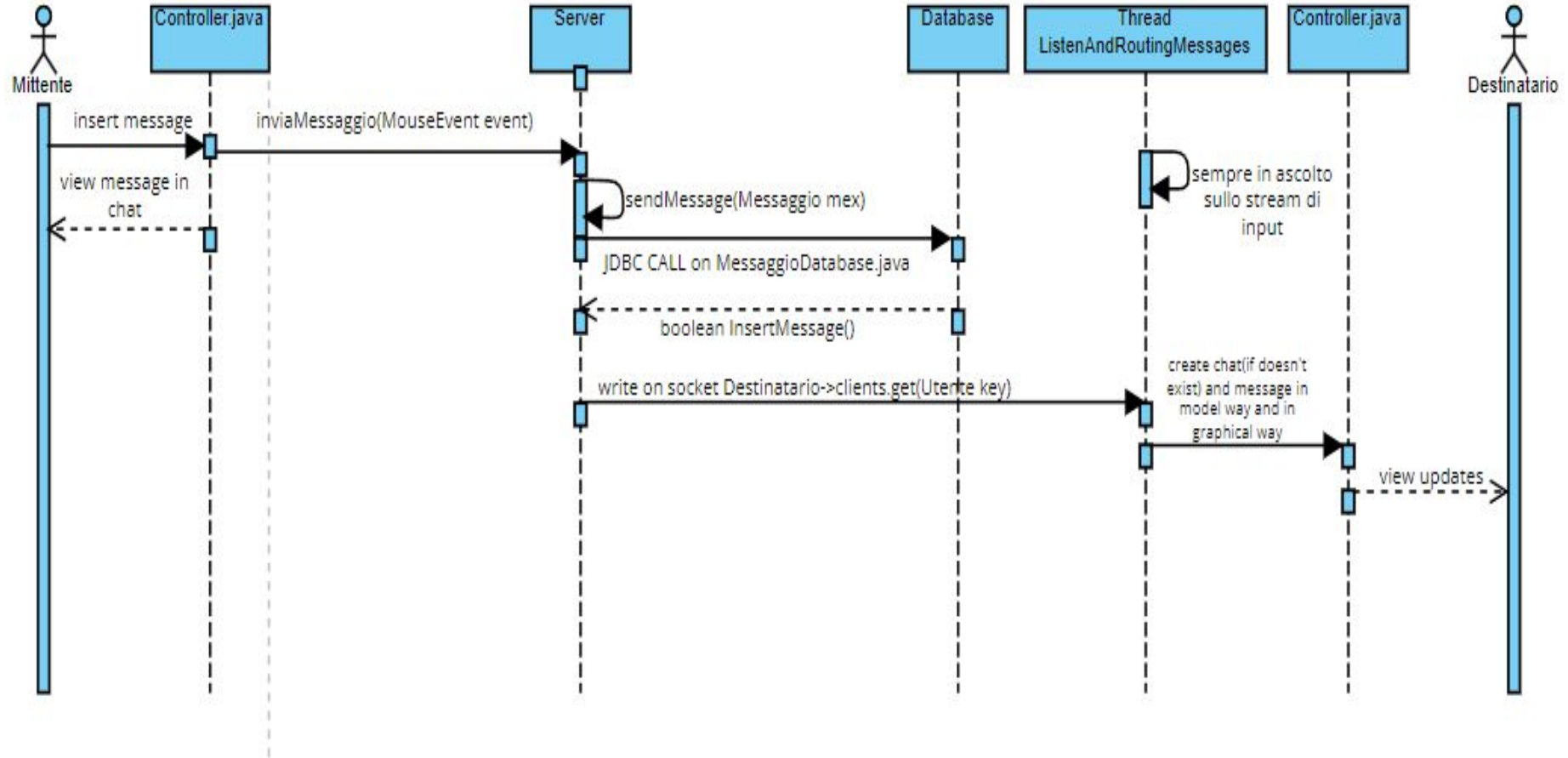
Aggiungere una nuova chat inserendo un username



Aggiungere una nuova chat inserendo un username



Invio Messaggio generico





Search

**Theo**

ciao theo

**Eva**

ciao

**Mario Rossi**

avverti anche gli altri!

**Mario Rossi**

A che ora ci vediamo?

alle 8, in piazza come al solito !

avverti anche gli altri!


Type something



Profilo

Profilo

Profilo



Username: Mario Rossi

Email: mariorossi@gmail.com

Nome: Mario

Cognome: Rossi

Argomenti che abbiamo studiato e che abbiamo applicato in questo progetto:

- Thread e l'implementazione della mutua esclusione(specificatamente monitor) + implementazione di variabili atomiche
- Socket TCP in Java
- Database
- Implementazione della crittografia per la sicurezza dei dati

Cosa abbiamo imparato:

- Creare oggetti in maniera dinamica in JAVAFX
- MVC(Model, View, Control) Pattern Architeturale.
- Connettersi tramite Java al database
- Riuscire a padroneggiare meglio la OOP con le sue potenzialità.

Extra

Link repository progetto: