



# Host2Host

Membri del team di sviluppo:

Yuri Noviello 0000873239

Alessio Di Domenico 0000882684

Jacopo Di Battista 0000880390

## Sommario

Abstract	3
Analisi dei requisiti	4
Requisiti del Sistema	4
Analisi del dominio	5
Tabella dei requisiti	5
Vocabolario	6
Analisi del dominio applicativo	8
Sistemi esterni	8
Analisi Dei Requisiti	9
Casi d'uso	9
Scenari	10
Analisi Del Rischio	21
Valutazione Dei Beni	21
Analisi minacce e controlli	22
Analisi Tecnologica della Sicurezza	23
Security Use Case e Misuse Case	24
Requisiti di Protezione dei Dati	27
Descrizione delle interfacce grafiche	27
Analisi Del Problema	31
Analisi delle funzionalità	31
Tabella delle funzionalità	31
Tabelle Informazioni/Flusso	32
Analisi Dei Vincoli	38
Tabella Dei Vincoli	38
Scomposizione del problema	39
Tabella Scomposizione Funzionalità	39
Tabelle Sotto-Funzionalità	39
Analisi Delle Interazioni	40
Tabella maschere	40
Tabella dei sistemi esterni	41
Analisi dei ruoli e delle responsabilità	42
Tabella Ruoli	42
Tabelle Ruolo-Informazioni	43
Modello del Dominio	44

Architettura Logica	46
Struttura	46
Interazione	50
Comportamento	56
Piano di Lavoro	56
Sviluppi Futuri	57
Piano di Collaudo	57
Progettazione	59
Requisiti Non Funzionali	59
Scelta dell'Architettura	60
Pattern & Design Principle	61
Progettazione di Dettaglio	62
Struttura	62
Interazione	70
Comportamento	76
Progettazione della Persistenza	76
Progettazione del Collaudo	77
Progettazione del Deployment	78
Artefatti	78
Deployment Type-Level	79

# Abstract

Da sempre una delle attività ricreative preferite dai giovani (e non) è quella di viaggiare, per conoscere nuove culture, allargare le proprie vedute o semplicemente per concedersi un momento di svago e di distacco dalla monotonia.

Sicuramente effettuare un viaggio è dispendioso e non alla portata di tutti, c'è anche il pericolo che non si conosca a priori cosa sia meglio visitare, quali attività intraprendere ecc.

Lo scopo del progetto è quello di realizzare uno strumento che permetta di abbattere i costi di un viaggio e di essere certi che si avrà una guida una volta giunti a destinazione.

Ogni Utente di *Host2Host*, dopo avere registrato la propria Abitazione e i servizi che intende mettere a disposizione ad un eventuale Ospite, avrà la possibilità di esprimere delle preferenze nei confronti di altre Abitazioni, situate nelle città che più gli interesserebbe visitare. Quando due Utenti avranno espresso una preferenza reciproca, potranno decidere di prenotare lo *Scambio*. Così facendo, ciascuno dei due sarà sia *Ospite* che *Ospitante* del suo Pari.



# Analisi dei requisiti

## Requisiti del Sistema

- L'*Utente non Registrato* si registra inserendo *email, password* e *dati anagrafici*
- Gli *Utenti* vengono autenticati tramite *username* e *password*
- Il *Sistema* permette la registrazione di un'*Abitazione* per ogni *Utente*
- Per ogni *Abitazione* vengono registrati *Città, Via, Stato, i Servizi* offerti con i relativi *costi*
- Il *Sistema* permette agli *Utenti* la ricerca di *Abitazioni* filtrando in base alla città desiderata
- Gli *Utenti* possono esprimere *interesse* per un'*Abitazione*
- Quando due *Utenti* si scambiano un *Like* reciprocamente possono prenotare lo *Scambio*
- Il *Sistema* permette ad un *Utente* di prenotare uno *Scambio* con un suo *Pari* (*Utente* con cui ha effettuato un *Match*)
- Dopo la *Prenotazione*, il *Sistema* permette agli *Utenti* di procedere al pagamento
- L'*Utente* può consultare lo *Storico* dei viaggi
- I privilegi sono basati su due tipi di account: *Utente* e *Amministratore*
- L'*Amministratore* deve poter consultare la lista delle *Abitazioni* e degli *Utenti*
- La valuta di riferimento è l'euro
- Gli *Utenti* hanno la possibilità di recensire l'esperienza dello *Scambio* effettuato
- Ogni *Recensione* contiene un voto e un testo



# Analisi del dominio

## Tabella dei requisiti

Id	Requisito	Tipo
R1F	Gli <i>Utenti non registrati</i> possono registrarsi al <i>Sistema</i> inserendo <i>email</i> , <i>password</i> e <i>dati anagrafici</i>	Funzionale
R2F	Agli <i>Amministratori</i> vengono fornite le credenziali di accesso al <i>Sistema</i>	Funzionale
R3F	Gli <i>Utenti</i> sono identificati con <i>username</i> (email)	Funzionale
R4F	Gli <i>Utenti</i> sono autenticati tramite <i>username</i> e <i>password</i>	Funzionale
R5F	Gli <i>Utenti</i> devono registrare un' <i>Abitazione</i> con relativi servizi per poter prenotare uno Scambio	Funzionale
R6F	L'approvazione delle <i>Abitazioni</i> è a carico di un <i>Amministratore</i>	Funzionale
R7F	Gli <i>Utenti</i> possono effettuare ricerche di <i>Abitazioni</i>	Funzionale
R8F	Gli <i>Utenti</i> possono esprimere <i>interesse</i> per un' <i>Abitazione</i>	Funzionale
R9F	Il <i>Sistema</i> comunica agli <i>Utenti</i> eventuali <i>Match</i>	Funzionale
R10F	Gli <i>Utenti</i> , terminato lo <i>Scambio</i> , possono rilasciare una <i>Recensione</i>	Funzionale
R11F	Gli <i>Utenti</i> possono segnalare <i>Utenti malintenzionati</i>	Funzionale
R12F	Il ruolo degli <i>Amministratori</i> riguarda la verifica delle <i>Abitazioni</i> , delle <i>Segnalazioni</i> e la rimozione degli <i>Utenti</i>	Funzionale

R13F	Il <i>Sistema</i> tiene traccia dello <i>Storico</i> dei viaggi degli <i>Utenti</i>	Funzionale
R14F	Il Sistema permette agli <i>Utenti</i> di prenotare uno <i>Scambio</i> ed effettuare il <i>pagamento</i>	Funzionale
R1NF	La password deve essere almeno di 8 caratteri	Non Funzionale
R2NF	L'email degli <i>Utenti</i> deve essere valida e verificata	Non Funzionale
R3NF	La <i>ricerca</i> deve essere veloce	Non Funzionale
R4NF	Il trattamento dei dati degli <i>Utenti</i> deve rispettare le normative	Non Funzionale
R5NF	L'interfaccia grafica deve essere intuitiva e di facile utilizzo	Non Funzionale

## Vocabolario

Voce	Definizione	Sinonimi
Sistema	Applicazione che fornisce i servizi di <i>Host2Host</i>	
Utente non registrato	Persona non ancora registrata al <i>Sistema</i>	
Utente	Persona registrata al <i>Sistema</i>	Viaggiatore
Utente non abilitato	<i>Utente</i> che non ha registrato un' <i>Abitazione</i>	
Utente abilitato	<i>Utente</i> che ha registrato un' <i>Abitazione</i> e può effettuare Scambi	
Amministratore	Persona abilitata a verificare <i>Abitazioni</i> , <i>Segnalazioni</i> ed eliminare <i>Utenti</i>	Admin, Amministratore di Sistema

Like	<i>Preferenza espressa da un Utente nei confronti di un'Abitazione in un dato periodo</i>	Preferenza, Interesse
Match	<i>Corrispondenza tra Utenti che hanno espresso interesse alle reciproche Abitazioni</i>	Corrispondenza
Ospite	<i>Utente che soggiorna nell'Abitazione di un altro Utente</i>	
Ospitante	<i>Utente che offre soggiorno ad un altro Utente</i>	
Abitazione	<i>Struttura che l'Ospitante mette a disposizione dell'Ospite, che comprende dei Servizi</i>	
Servizio	<i>Attività (a pagamento) che l'Ospitante offre durante il soggiorno dell'Ospite</i>	
Pari	<i>Persona con cui un Utente ha effettuato un Match</i>	
Recensione	<i>Valutazione che un Utente può esprimere nei confronti del Pari al termine dello Scambio</i>	Valutazione
Ricerca	<i>Metodo a disposizione di un Utente per cercare Abitazioni</i>	
Viaggio Ospite	<i>Viaggio di un Utente nel ruolo di Ospite</i>	
Viaggio Ospitante	<i>Attività che l'Utente svolge nel ruolo di Ospitante</i>	
Scambio	<i>Combinazione dei Viaggi Ospite e Viaggi Ospitante di due Utenti</i>	



Segnalazione	Notifica di presunta attività non conforme effettuata da un <i>Utente</i>	
DataInizio Disponibilità	Giorno dal quale si rende disponibile un' <i>Abitazione</i>	
DataFine Disponibilità	Giorno dal quale un' <i>Abitazione</i> non è più disponibile	
DataInizio Viaggio	Giorno di inizio <i>Viaggio</i> indicato nella <i>Ricerca</i>	
DataFine Viaggio	Giorno di fine <i>Viaggio</i> indicato nella <i>Ricerca</i>	

\*\*L'intervallo [DataInizioViaggio, DataFineViaggio] deve essere contenuto nell'intervallo [DataInizioDisponibilità, DataFineDisponibilità]

## Analisi del dominio applicativo

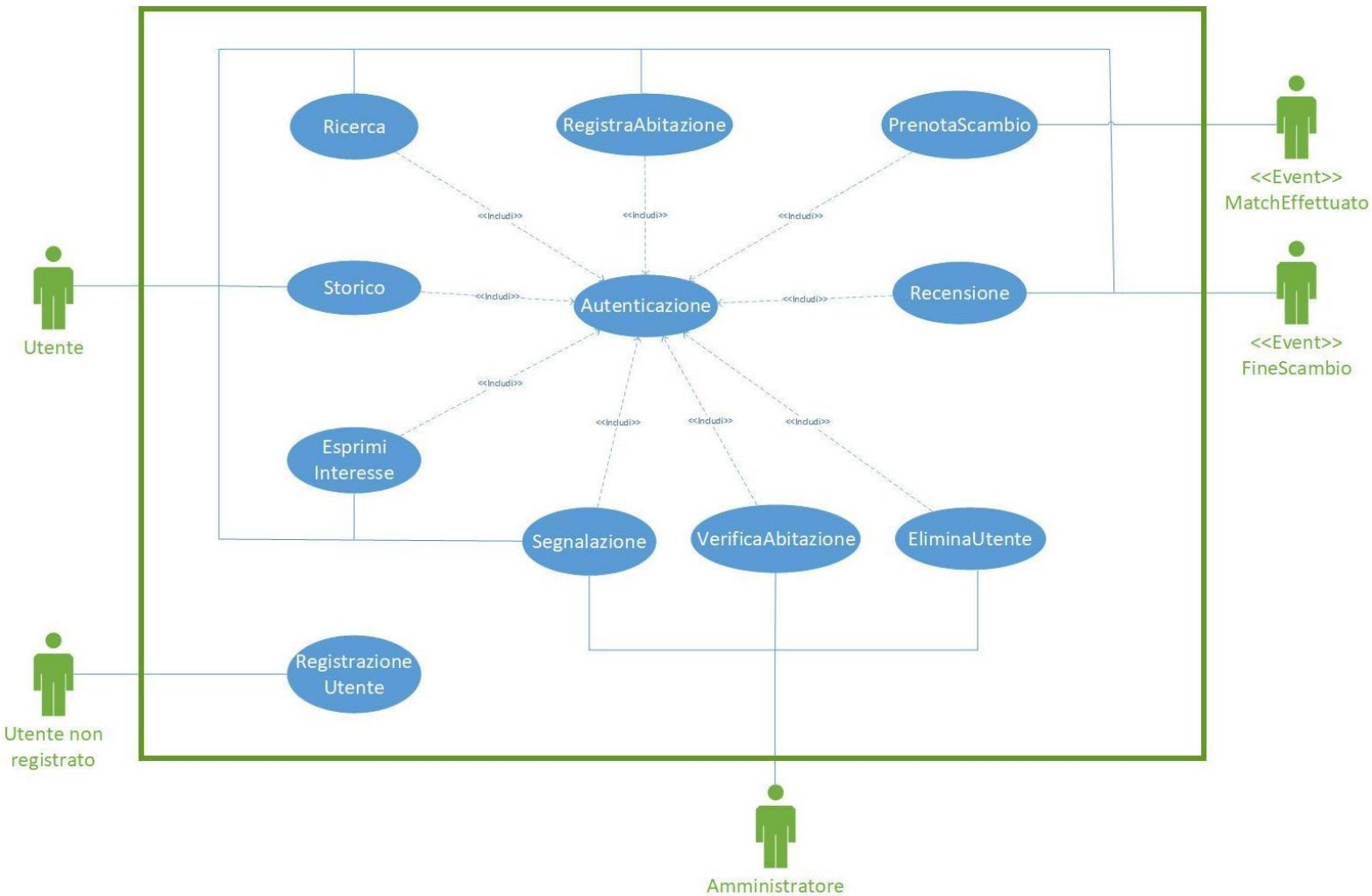
A differenza dei comuni strumenti che permettono di prenotare un alloggio nei luoghi destinati ad un viaggio, *Host2Host* ha la peculiarità di trasformare un viaggio in un vero e proprio scambio culturale. Si ritiene che i costi saranno sensibilmente più bassi dello standard in quanto le spese coprono i soli servizi che l'*Ospitante* mette a disposizione del *Pari*.

## Sistemi esterni

Host2Host dovrà interfacciarsi con un sistema esterno per la gestione dei pagamenti.

# Analisi Dei Requisiti

## Casi d'uso



L'*Autenticazione* non è un requisito da effettuare ogni volta prima di svolgere ciascuna delle operazioni che la includono. E', invece, necessario essere già stati autenticati prima di *effettuare una ricerca, prenotare uno scambio ecc.*

## Scenari

Titolo	Autenticazione
Descrizione	Azione mediante la quale un <i>Utente</i> o un <i>Amministratore</i> si autentica presso il <i>Sistema</i>
Attori	<i>Utente, Amministratore</i>
Relazioni	<i>EsprimiInteresse, Storico, Ricerca, RegistraAbitazione, Recensione, PrenotaScambio, VerificaAbitazione, Segnalazione, EliminaUtente</i>
Precondizioni	L' <i>Utente</i> o l' <i>Amministratore</i> è già registrato
Postcondizioni	L' <i>Utente</i> o l' <i>Amministratore</i> risulta autenticato
Scenario Principale	<ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> presenta la schermata di autenticazione all'<i>Utente/Amministratore</i></li> <li>2. L'<i>Utente/Amministratore</i> inserisce <i>username</i> e <i>password</i></li> <li>3. L'<i>Utente/Amministratore</i> preme il bottone di login</li> <li>4. Il <i>Sistema</i> verifica le credenziali</li> <li>5. L'<i>Utente/Amministratore</i> viene reindirizzato sulla pagina di sua competenza</li> </ol>
Scenari Alternativi	<p>Scenario a: L'<i>username</i> non è presente nel <i>Sistema</i></p> <ol style="list-style-type: none"> <li>1. All'<i>Utente/Amministratore</i> viene mostrato un avviso di errore</li> <li>2. L'<i>Utente/Amministratore</i> viene reindirizzato sulla pagina di accesso</li> </ol> <p>Scenario b: La <i>password</i> non corrisponde a quella corretta</p> <ol style="list-style-type: none"> <li>1. All'<i>Utente/Amministratore</i> viene mostrato un avviso di errore</li> <li>2. L'<i>Utente/Amministratore</i> viene reindirizzato sulla pagina di accesso.</li> </ol>
Requisiti non funzionali	La <i>password</i> non deve essere esplicitamente visibile durante l'autenticazione dell' <i>Utente/Amministratore</i>
Punti aperti	

Titolo	EliminaUtente
Descrizione	L'Amministratore rimuove un <i>Utente</i> dal <i>Sistema</i>
Attori	Amministratore
Relazioni	Autenticazione
Precondizioni	L'Amministratore è stato autenticato e l' <i>Utente</i> è un <i>Utente registrato</i>
Postcondizioni	L' <i>Utente</i> e i suoi dati vengono rimossi dal <i>Sistema</i>
Scenario Principale	<ol style="list-style-type: none"> <li>1. L'Amministratore seleziona l'<i>Utente</i> da eliminare</li> <li>2. Viene chiesta la conferma dell'operazione</li> <li>3. L'Amministratore conferma l'operazione</li> <li>4. Il <i>Sistema</i> elimina l'<i>Utente</i> e i dati a lui relativi</li> <li>5. Viene notificato il successo dell'operazione</li> </ol>
Scenari Alternativi	Scenario a: L'Amministratore non conferma l'operazione <ol style="list-style-type: none"> <li>1. L'Amministratore declina la conferma di eliminazione</li> <li>2. Il <i>Sistema</i> notifica l'annullamento dell'operazione</li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

Titolo	Segnalazione
Descrizione	Un <i>Utente</i> effettua una <i>Segnalazione</i> nei confronti di una presunta attività non conforme

Attori	<i>Utente, Amministratore</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	<i>L'Amministratore e l'Utente sono stati Autenticati</i>
Postcondizioni	<i>La Segnalazione è stata valutata dall'Amministratore</i>
Scenario Principale	<ol style="list-style-type: none"> <li>1. L'Utente seleziona l'area relativa alle <i>Segnalazioni</i></li> <li>2. L'Utente scrive l'username dell'Utente da segnalare e una breve descrizione della presunta attività non conforme e invia la <i>Segnalazione</i></li> <li>3. L'Amministratore incaricato valuta la <i>Segnalazione</i></li> <li>4. L'Amministratore procede ad eliminare l'Utente segnalato</li> </ol>
Scenari Alternativi	<p>Scenario a: L'Amministratore non elimina l'Utente</p> <ol style="list-style-type: none"> <li>1. L'Utente seleziona l'area relativa alle <i>Segnalazioni</i></li> <li>2. L'Utente scrive l'username dell'Utente da segnalare e una breve descrizione della presunta attività non conforme e invia la <i>Segnalazione</i></li> <li>3. L'Amministratore incaricato valuta la <i>Segnalazione</i></li> <li>4. L'Amministratore non ritenendo appropriata la <i>Segnalazione</i> non intraprende alcuna contromisura</li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

Titolo	RegistrazioneUtente
Descrizione	Nel <i>Sistema</i> viene registrato un nuovo <i>Utente</i>
Attori	<i>UtenteNonRegistrato</i>
Relazioni	
Precondizioni	La persona non è già registrata nel <i>Sistema</i>
Postcondizioni	L' <i>Utente</i> risulta registrato nel <i>Sistema</i> e potrà accedere con le proprie credenziali
Scenario Principale	<ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> presenta una schermata contenente i campi necessari per la registrazione</li> <li>2. L'<i>UtenteNonRegistrato</i> compila i campi</li> <li>3. L'<i>UtenteNonRegistrato</i> preme sul pulsante di registrazione</li> <li>4. Il <i>Sistema</i> controlla i dati inseriti e chiede di confermare l'email entro 24h</li> <li>5. L'<i>UtenteNonRegistrato</i> conferma l'email</li> <li>6. Il <i>Sistema</i> conferma la registrazione</li> <li>7. L'<i>UtenteNonRegistrato</i> diventa un <i>Utente</i> che può autenticarsi al <i>Sistema</i></li> </ol>
Scenari Alternativi	<p>Scenario a: Non sono stati inseriti dei dati obbligatori</p> <ol style="list-style-type: none"> <li>1. All'<i>UtenteNonRegistrato</i> viene mostrato un avviso di errore</li> <li>2. L'<i>UtenteNonRegistrato</i> è rimandato alla pagina di registrazione</li> </ol> <p>Scenario b: Non è stata verificata l'email prima del tempo massimo</p> <ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> disabilita la conferma dell'email per l'<i>UtenteNonRegistrato</i></li> <li>2. Il <i>Sistema</i> non conferma la registrazione</li> </ol>
Requisiti non funzionali	La <i>password</i> deve essere di almeno 8 caratteri e non deve essere visibile durante l'inserimento. Integrità e protezione dei dati
Punti aperti	

Titolo	RegistraAbitazione
Descrizione	Nel <i>Sistema</i> viene registrata una nuova <i>Abitazione</i>
Attori	<i>Utente</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	<i>L'Utente</i> deve essere stato autenticato dal <i>Sistema</i>
Postcondizioni	All' <i>Utente</i> viene associata una nuova <i>Abitazione</i> che può essere ricercata da altri <i>Utenti</i> e viene abilitato ad effettuare <i>Scambi</i>
Scenario Principale	<ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> presenta una schermata contenente i campi necessari per la registrazione dell'<i>Abitazione</i></li> <li>2. L'<i>Utente</i> compila i campi</li> <li>3. L'<i>Utente</i> preme sul pulsante di registrazione dell'<i>Abitazione</i></li> <li>4. L'<i>Utente</i> resta in attesa fino alla corretta verifica dell'<i>Abitazione</i></li> <li>5. L'<i>Amministratore</i> verifica con successo l'<i>Abitazione</i></li> <li>6. L'<i>Utente</i> riceve un messaggio di conferma</li> <li>7. Il <i>Sistema</i> associa l'<i>Abitazione</i> all'<i>Utente</i></li> <li>8. L'<i>Utente</i> viene abilitato ad effettuare <i>Scambi</i></li> </ol>
Scenari Alternativi	<p>Scenario a: Non sono stati inseriti dei dati obbligatori</p> <ol style="list-style-type: none"> <li>1. All'<i>Utente</i> viene mostrato un avviso di errore</li> <li>2. L'<i>Utente</i> è rimandato alla pagina di <i>Registrazione</i></li> </ol> <p>Scenario b: L'<i>Abitazione</i> non viene approvata dall'<i>Amministratore</i></p> <ol style="list-style-type: none"> <li>1. Viene comunicato all'<i>Utente</i> che non è stata approvata l'<i>Abitazione</i></li> <li>2. Il <i>Sistema</i> annulla la registrazione</li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

Titolo	VerificaAbitazione
Descrizione	<i>L'Amministratore verifica un'Abitazione registrata da un Utente non ancora confermata</i>
Attori	<i>Amministratore</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	<i>L'Amministratore è stato autenticato e un Utente ha richiesto la verifica di un'Abitazione</i>
Postcondizioni	<i>L'Abitazione è stata valutata dall'Amministratore</i>
Scenario Principale	<ol style="list-style-type: none"> <li>1. <i>L'Amministratore seleziona l'Abitazione da verificare</i></li> <li>2. <i>L'Amministratore effettua gli opportuni controlli</i></li> <li>3. <i>L'Amministratore conferma l'Abitazione</i></li> <li>4. <i>Il Sistema notifica all'Utente l'approvazione della sua Abitazione</i></li> <li>5. <i>La registrazione dell'Abitazione da parte dell'Utente è confermata</i></li> </ol>
Scenari Alternativi	<p>Scenario a:</p> <p><i>L'Amministratore ritiene non idonea l'Abitazione</i></p> <ol style="list-style-type: none"> <li>1. <i>L'Amministratore declina la richiesta dell'Utente</i></li> <li>2. <i>Il Sistema notifica all'Utente l'esito negativo della verifica</i></li> <li>3. <i>La registrazione dell'Abitazione da parte dell'Utente è annullata</i></li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	



Titolo	Storico
Descrizione	Il <i>Sistema</i> mostra all' <i>Utente</i> i viaggi effettuati e le relative recensioni
Attori	<i>Utente</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	L' <i>Utente</i> è registrato ed autenticato
Postcondizioni	
Scenario Principale	<ol style="list-style-type: none"> <li>1. L'<i>Utente</i> clicca sul pulsante per visualizzare lo storico dei viaggi</li> <li>2. Viene visualizzata la lista dei viaggi</li> <li>3. L'<i>Utente</i> seleziona il viaggio di suo interesse</li> <li>4. Il <i>Sistema</i> mostra i dettagli relativi al viaggio</li> </ol>
Scenari Alternativi	Scenario a: Lo storico dei viaggi è vuoto <ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> comunica che non ci sono viaggi da visualizzare</li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

Titolo	Recensione
Descrizione	L' <i>Utente</i> scrive una recensione sullo <i>Scambio</i> avvenuto con il <i>Pari</i>

Attori	<i>Utente</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	L' <i>Utente</i> è registrato, autenticato ed ha terminato uno <i>Scambio</i>
Postcondizioni	Il <i>Sistema</i> associa la nuova recensione al <i>Pari</i>
Scenario Principale	<ol style="list-style-type: none"> <li>1. L'<i>Utente</i> seleziona lo <i>Scambio</i> che vuole recensire</li> <li>2. L'<i>Utente</i> assegna un voto da 1 a 5</li> <li>3. L'<i>Utente</i> aggiunge un commento</li> <li>4. L'<i>Utente</i> conferma la recensione</li> <li>5. Il <i>Sistema</i> controlla che la recensione non contenga parole vietate</li> <li>6. La Recensione viene pubblicata</li> </ol>
Scenari Alternativi	<p>Scenario a: L'<i>Utente</i> non compila correttamente i campi</p> <ol style="list-style-type: none"> <li>1. Viene segnalato l'errore all'<i>Utente</i></li> <li>2. L'<i>Utente</i> è reindirizzato sulla pagina di recensione</li> </ol> <p>Scenario b: Il <i>Sistema</i> rileva parole vietate</p> <ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> rileva l'anomalia</li> <li>2. La recensione viene annullata</li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

Titolo	PrenotaScambio
Descrizione	Il <i>Sistema</i> permette all' <i>Utente</i> di prenotare lo <i>Scambio</i>

Attori	<i>Utente</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	L' <i>Utente</i> è autenticato ed è stato effettuato un <i>Match</i> con un altro <i>Utente</i>
Postcondizioni	
Scenario Principale	<ol style="list-style-type: none"> <li>1. L'<i>Utente</i> seleziona l'<i>Abitazione</i> dell'<i>Utente Pari</i> con il quale ha effettuato il <i>Match</i></li> <li>2. L'<i>Utente</i> invia la richiesta di <i>Scambio</i></li> <li>3. L'<i>Utente</i> resta in attesa di conferma da parte del <i>Pari</i></li> <li>4. Il <i>Sistema</i> notifica all'<i>Utente</i> che la sua richiesta è stata accettata</li> <li>5. L'<i>Utente</i> procede al pagamento</li> </ol>
Scenari Alternativi	<p>Scenario a: Il <i>Pari</i> rifiuta la richiesta di <i>Scambio</i> dell'<i>Utente</i></p> <ol style="list-style-type: none"> <li>1. All'<i>Utente</i> viene notificato che il <i>Pari</i> ha rifiutato la richiesta</li> <li>2. Il <i>Match</i> tra gli <i>Utenti</i> è annullato</li> </ol> <p>Scenario b: L'<i>Utente</i> ha ricevuto la richiesta da parte del <i>Pari</i></p> <ol style="list-style-type: none"> <li>1. Il <i>Sistema</i> notifica all'<i>Utente</i> che il <i>Pari</i> ha inviato una richiesta di <i>Scambio</i></li> <li>2. L'<i>Utente</i> conferma la richiesta del <i>Pari</i></li> <li>3. Il <i>Sistema</i> notifica all'<i>Utente Pari</i> che lo scambio è confermato</li> <li>4. L'<i>Utente</i> procede al pagamento</li> </ol> <p>Scenario c: L'<i>Utente</i> rifiuta la richiesta di scambio del <i>Pari</i></p> <ol style="list-style-type: none"> <li>1. L'<i>Utente</i> rifiuta la richiesta del <i>Pari</i></li> <li>2. Il <i>Sistema</i> notifica al <i>Pari</i> che l'<i>Utente</i> ha rifiutato la richiesta e il <i>Match</i> è annullato</li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

Titolo	Ricerca
Descrizione	L' <i>Utente</i> effettua una ricerca sulle <i>Abitazioni</i>
Attori	<i>Utente</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	L' <i>Utente</i> è autenticato dal <i>Sistema</i>
Postcondizioni	
Scenario Principale	<ol style="list-style-type: none"> <li>1. L'<i>Utente</i> effettua una ricerca inserendo nella barra dedicata il nome della città di interesse e le date di permanenza</li> <li>2. L'<i>Utente</i> conferma la ricerca premendo sul pulsante apposito</li> <li>3. Il <i>Sistema</i> mostra all'<i>Utente</i> la lista delle <i>Abitazioni</i> disponibili presso la città di interesse nel periodo indicato</li> </ol>
Scenari Alternativi	
Requisiti non funzionali	La ricerca deve essere veloce, integrità e protezione dei dati
Punti aperti	

Titolo	EsprimiInteresse
Descrizione	Un <i>Utente</i> esprime interesse nei confronti di un' <i>Abitazione</i> durante una <i>Ricerca</i>

Attori	<i>Utente</i>
Relazioni	<i>Autenticazione</i>
Precondizioni	<i>L'Utente deve essere stato autenticato dal Sistema, deve aver registrato un'Abitazione</i>
Postcondizioni	<i>L'Utente prova interesse per l'Abitazione</i>
Scenario Principale	<ol style="list-style-type: none"> <li>1. <i>L'Utente</i> seleziona l'<i>Abitazione</i> che gli interessa</li> <li>2. <i>L'Utente</i> preme sul pulsante "<i>Like</i>" esprimendo il proprio interesse</li> <li>3. Il <i>Sistema</i> memorizza il <i>Like</i> dell'<i>Utente</i></li> </ol>
Scenari Alternativi	<p>Scenario a:  <i>L'Utente Pari</i> aveva già messo <i>Like</i> all'<i>Abitazione</i> dell'<i>Utente</i></p> <ol style="list-style-type: none"> <li>1. Viene immediatamente notificato il <i>Match</i> ad entrambi gli <i>Utenti</i></li> <li>2. Viene abilitata l'operazione <i>PrenotaScambio</i> per gli <i>Utenti</i></li> </ol>
Requisiti non funzionali	Integrità e protezione dei dati
Punti aperti	

# Analisi Del Rischio

## Valutazione Dei Beni

Bene	Valore	Esposizione
Sistema Informativo	Alto. Supporto alla gestione dell'intera applicazione. Critico dal punto di vista della sicurezza.	Alta. Perdita finanziaria e di immagine. Costi per il ripristino del <i>Sistema</i> .
Informazioni relative agli Utenti	Alto. Informazioni personali degli <i>Utenti</i> , comprese le credenziali.	Alta. Perdita d'immagine se gli <i>Utenti</i> vengono compromessi.
Informazioni relative alle Abitazioni	Alto. Informazioni sensibili relative alle <i>Abitazioni</i> .	Alta. Perdita d'immagine e possibili ripercussioni legali in caso di divulgazione di dati.
Record Scambio	Alto. Informazioni riguardanti uno specifico <i>Scambio</i> .	Alta. Perdita dei dati di uno <i>Scambio</i> .
Informazioni relative agli Amministratori	Alto. Informazioni relative agli amministratori, comprese le loro credenziali che permettono di accedere e modificare parti critiche del <i>Sistema</i> .	Alta. Perdita d'immagine: un attaccante con permessi amministrativi può compromettere seriamente il <i>Sistema</i> .

## Analisi minacce e controlli

Minaccia	Probabilità	Controllo	Fattibilità
Furto credenziali Utente	Alta	Log di tutte le operazioni effettuate dall' <i>Utente</i> .	Basso costo di implementazione.
Furto credenziali Amministratore	Media	Accesso amministratore solo da locale. Log degli accessi e delle operazioni.	Basso costo, limita i punti di attacco.
DoS	Bassa	Progettazione adeguata, controllo e limitazione accessi.	Basso costo. Impossibile prevenire un attacco DoS.
Intercettazione Comunicazioni/ Sniffing	Alta. Sistema distribuito con comunicazioni client/server.	Cifratura.	Costo variabile a seconda del tipo di cifratura scelta.
Richiesta o invio di dati di prenotazione provenienti da fonti non autorizzate	Bassa	Log operazioni per identificare la provenienza di richieste malevole.	Basso costo di implementazione.
Man in the Middle	Media	Autenticazione end-to-end tramite certificati.	Costo basso in aggiunta all'implementazione SSL.

## Analisi Tecnologica della Sicurezza

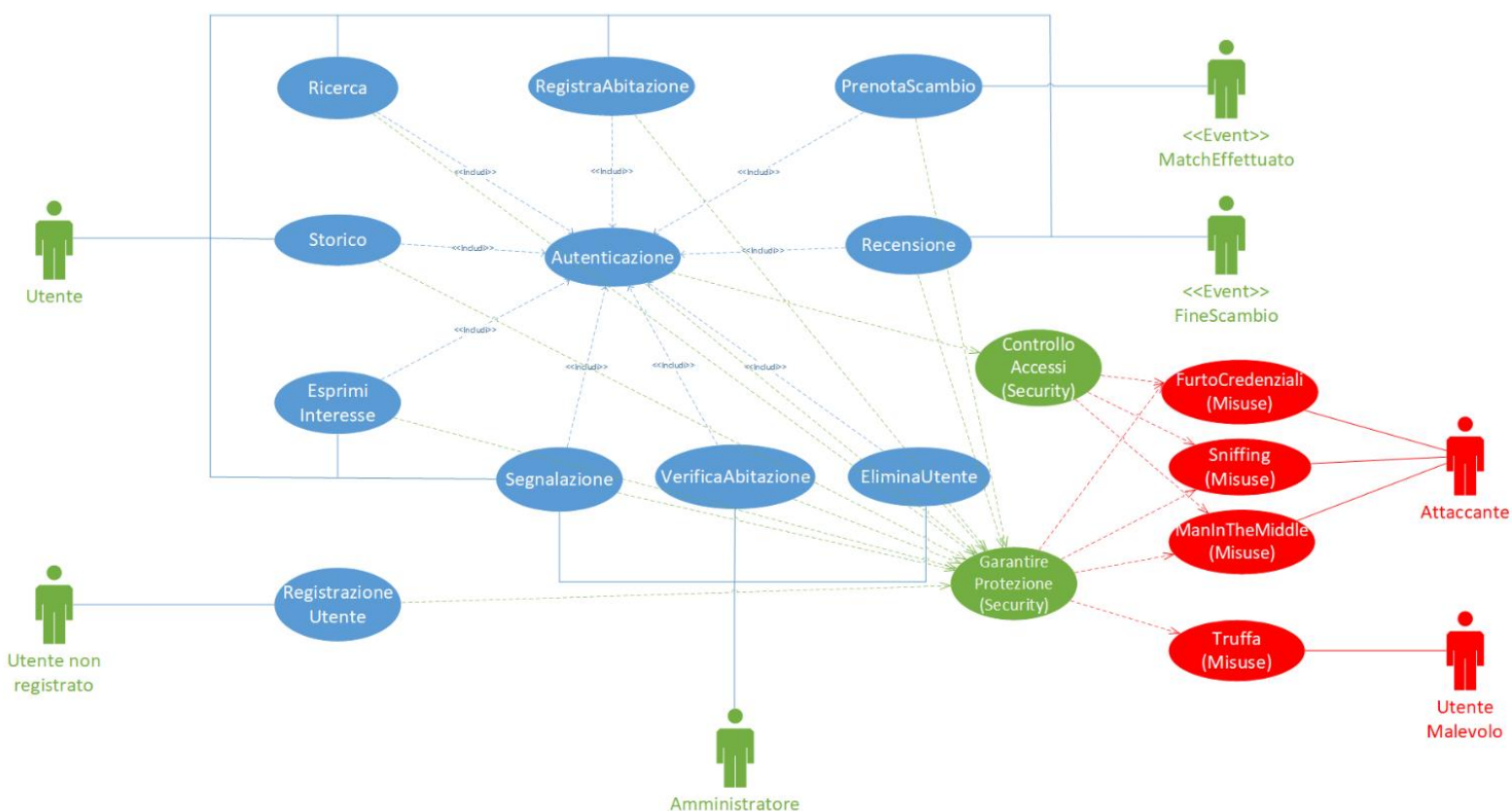
Tecnologia	Vulnerabilità
Cifratura delle comunicazioni	<p>Per ottenere una comunicazione sicura, bisogna utilizzare sia una cifratura simmetrica che asimmetrica. Ognuna presenta delle vulnerabilità:</p> <ul style="list-style-type: none"><li>-Cifratura Simmetrica:<ol style="list-style-type: none"><li>1. Tempo di vita della chiave</li><li>2. Memorizzazione della chiave</li><li>3. Lunghezza della chiave</li></ol></li><li>-Cifratura Asimmetrica:<ol style="list-style-type: none"><li>1. Lunghezza della chiave</li><li>2. Memorizzazione della chiave privata</li></ol></li></ul> <p>L'utilizzo di cifrature standard non recenti potrebbero causare vulnerabilità ulteriori</p>
Architettura Client/Server	<ul style="list-style-type: none"><li>-Man in the Middle: un malintenzionato potrebbe riuscire a sottrarre informazioni o effettuare operazioni illecite.</li><li>-Attacco DoS/DDoS: l'obiettivo dell'attacco è quello di negare il servizio offerto agli <i>Utenti</i>.</li><li>-Intercettazione della comunicazione (sniffing): un malintenzionato potrebbe riuscire ad intercettare una comunicazione e a sottrarre informazioni.</li></ul>



## Autenticazione tramite credenziali

- Phishing: Vengono sottratte le credenziali all'*Utente* tramite metodi di phishing.
- Password Banali: l'*Utente* sceglie una password facile da indovinare.
- Social Engineering: un *Utente* potrebbe essere vittima di un attacco di social engineering, volto al furto delle credenziali.
- Password Cracking: attraverso metodologie di cracking delle password, l'attaccante risale alla password di un *Utente* o *Amministratore di Sistema*.
- Negligenza: L'*Utente* volontariamente o per sbaglio fornisce le credenziali a terzi.

## Security Use Case e Misuse Case



Titolo	ControlloAccesso	
Descrizione	Tutti gli accessi al <i>Sistema</i> devono essere controllati	
Misuse Case	FurtoCredenziali, Sniffing, ManInTheMiddle	
Relazioni		
Precondizioni	L'Attaccante entra in possesso delle credenziali di un <i>Utente</i>	
Postcondizioni	L'accesso dell'Attaccante viene bloccato dal <i>Sistema</i>	
Scenario Principale	Sistema	Attaccante
		L'Attaccante è in possesso dell' <i>username</i> di un <i>Utente</i> registrato al <i>Sistema</i> e tenta un attacco di <i>password cracking</i>
	Il <i>Sistema</i> rileva il tentativo di attacco e sospende l'account dell' <i>Utente</i>	
Scenario di attacco avvenuto con successo	Sistema	Attaccante
		L'Attaccante effettua il login
	Il <i>Sistema</i> consente l'accesso all'Attaccante	
		L'Attaccante naviga nel <i>Sistema</i> con l'obiettivo di ottenere i dati dell' <i>Utente</i>
	Il <i>Sistema</i> trascrive nei file di log le operazioni effettuate dall'Attaccante	

Titolo	Garantire Protezione	
Descrizione	Integrità dei dati salvati dal <i>Sistema</i>	
Misuse Case	Furto Credenziali, Sniffing, ManInTheMiddle, Truffa	
Relazioni		
Precondizioni	Il <i>Sistema</i> ha registrato dati sensibili che non devono essere danneggiati o modificati da chi non è autorizzato	
Postcondizioni	Il <i>Sistema</i> monitora le attività sospette e notifica la possibile alterazione dei dati	
Scenario Principale	Sistema	Attaccante
		L' <i>Attaccante</i> cerca di decifrare e alterare i dati del <i>Sistema</i>
	Il <i>Sistema</i> rileva il tentativo di modifica e impedisce la manomissione di dati sensibili	
Scenario di attacco avvenuto con successo	Sistema	Attaccante
		L' <i>Attaccante</i> riesce a decifrare e alterare dati sensibili
	Tutte le operazioni effettuate vengono salvate nei log di <i>Sistema</i> . Operazione potenzialmente malevole sono notificate all' <i>Amministratore</i>	

## Requisiti di Protezione dei Dati

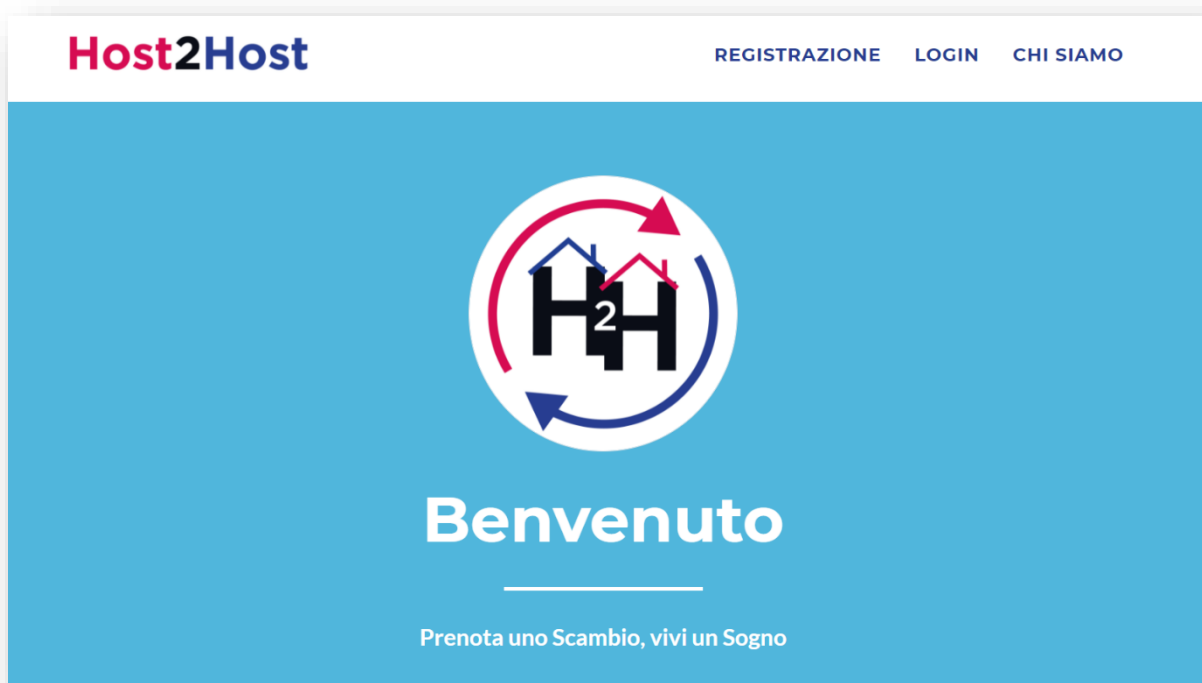
Dall'analisi del rischio sono emersi altri requisiti riguardanti la protezione dei dati:

1. Creazione di un *Sistema* di Log per tracciare tutte le operazioni del *Sistema*. Ogni interazione viene tracciata e registrata in maniera persistente. L'*Amministratore* è l'unico in grado di poter visionare i Log
2. I dati memorizzati dal *Sistema* devono essere protetti da un eventuale *Attaccante*, eventualmente adottando una cifratura dei dati
3. I dati scambiati nel *Sistema* devono essere protetti

## Descrizione delle interfacce grafiche

Il sistema presenta diverse sezioni raggiungibili dalla barra in alto a destra. La pagina visualizzata dopo l'accesso è dipendente dal ruolo di chi lo effettua. L'*UtenteNonAbilitato* non potrà accedere a tutte le funzionalità dell'applicazione fino alla registrazione di un'*Abitazione*. All'*Amministratore* verrà mostrata una sezione dedicata per poter effettuare le funzionalità a lui riservate. Di seguito sono riportate alcune tra delle interfacce dell'applicazione.


Home principale



## Autenticazione

**Host2Host**REGISTRAZIONE

### LOGIN



Login

## Home Amministratore

**Host2Host**SEGNALAZIONIABITAZIONIUTENTILOGOUT

### CIAO ADMIN

Concluse	Da fare
Utenti eliminati: 5	Utenti da eliminare: 2
Segnalazioni verificate: 9	Segnalazioni da verificare: 4
Abitazioni verificate: 22	Abitazioni da verificare: 7



## Registra Abitazione

**Host2Host** **LOGOUT**

Registra la tua Abitazione

Stato

Ex. Italy

Città

Ex. Bologna

Indirizzo

Ex. Viale del Risorgimento, 2

Data di inizio disponibilità

gg/mm/aaaa

Data di fine disponibilità

gg/mm/aaaa

Foto

UPLOAD FOTO

Servizi

Aggiungi

Registra Abitazione


## Home Utente Abilitato

**Host2Host** **STORICO** **RICERCA** **SCAMBIO** **LOGOUT**


**CIAO ALESSIO**

**Bologna, Italy**

Via San Mamolo, 14 (BO)

 Alessio Di Domenico, 22 anni

**Disponibile**  
dal 22/05/2021 al 10/08/2021



**Servizi**

**Gita al parco** Passeggiata nel verde dei colli bolognesi **5€**

**Giro in vespa** Giro in vespa sui colli bolognesi a mangiare tortellini **10€**

**Visita al Museo** Visita al museo di arte contemporanea al MAMbo **25€**

# Analisi Del Problema

## Analisi delle funzionalità

### Tabella delle funzionalità

Funzionalità	Tipo	Grado complessità
Autenticazione	Gestione di dati	Semplice
Ricerca	Gestione dei dati	Semplice
Storico	Gestione e memorizzazione dei dati	Semplice
EsprimiInteresse	Gestione e memorizzazione dei dati	Semplice
RegistraAbitazione	Gestione e memorizzazione dei dati	Semplice
Recensione	Memorizzazione dei dati	Semplice
PrenotaScambio	Gestione e memorizzazione dei dati, interazione con l'esterno	Complessa
RegistrazioneUtente	Gestione e memorizzazione dei dati	Semplice
VerificaAbitazione	Gestione e memorizzazione dei dati	Semplice
EliminaUtente	Gestione e memorizzazione dei dati	Semplice

Segnalazione	Gestione e memorizzazione dei dati	Complessa
Visualizza log	Gestione dei dati	Semplice
Scrittura log	Memorizzazione dei dati	Semplice

## Tabelle Informazioni/Flusso

### Autenticazione – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Username	Semplice	Molto alta	Input	Non più di 128 caratteri
Password	Semplice	Molto alta	Input	Non più di 128 caratteri

### Ricerca – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Città	Semplice	Bassa	Input	Non più di 128 caratteri
DataInizio Viaggio	Semplice	Bassa	Input	
DataFine Viaggio	Semplice	Bassa	Input	
ListaAbitazioni	Composto	Media	Output	



### Storico – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
ListaScambi	Complesso	Alta	Output	

### EsprimiInteresse – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Username Utente	Semplice	Alta	Input	
Like  Composta da: UsernameUtente AbitazionePari DataInizioViaggio DataFineViaggio	Composto	Media	Input	

\*AbitazionePari rappresenta l'Abitazione che ha ricevuto il Like

### RegistraAbitazione – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Via	Semplice	Alta	Input	Non più di 128 caratteri
Città	Semplice	Alta	Input	Non più di 128 caratteri

Stato	Semplice	Alta	Input	Non più di 128 caratteri
Foto	Semplice	Alta	Input	Non più di 20 immagini, formato JPEG/PNG
PrezzoServizi	Semplice	Alta	Input	Valore numerico, nella valuta €
ListaServizi	Composto	Alta	Input	
DataInizioDisponibilità	Semplice	Alta	Input	
DataFineDisponibilità	Semplice	Alta	Input	
UsernameUtente	Semplice	Alta	Input	

### Recensione – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Username Utente	Semplice	Alta	Input	
IdPari	Semplice	Alta	Input	
Testo	Semplice	Media	Input	Non più di 1000 caratteri
Voto	Semplice	Media	Input	Valore numerico compreso tra 1,5

## PrenotaScambio – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Username Utente	Semplice	Alta	Input	
IdPari	Semplice	Alta	Input	
DataInizio ViaggioOspite	Semplice	Bassa	Input	
DataFine ViaggioOspite	Semplice	Bassa	Input	
DataInizio ViaggioOspitante	Semplice	Bassa	Input	
DataFine ViaggioOspitante	Semplice	Bassa	Input	
PrezzoUtente	Semplice	Alta	Input	Valore numerico, nella valuta €
PrezzoPari	Semplice	Alta	Input	Valore numerico, nella valuta €

## RegistrazioneUtente – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Nome	Semplice	Alta	Input	Non più di 256 caratteri
Cognome	Semplice	Alta	Input	Non più di 256 caratteri
DataDiNascita	Semplice	Alta	Input	

Documento Identificazione	Semplice	Alta	Input	Deve essere un documento valido, valore alfanumerico
Numero di cellulare	Semplice	Alta	Input	Valore numerico, deve essere un numero valido
Indirizzo Email	Semplice	Alta	Input	Deve essere un indirizzo e-mail valido
Password	Semplice	Molto alta	Input	Almeno 8 caratteri

#### VerificaAbitazione – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione/ Privacy	Input/ Output	Vincoli
IdAbitazione	Semplice	Media	Input	
IsVerificato	Semplice	Media	Input	Valore booleano
Username Amministratore	Semplice	Alta	Input	

#### EliminaUtente – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Username Utente	Semplice	Alta	Input	
Username Amministratore	Semplice	Alta	Input	

### Segnalazione – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
IdSegnalazione	Semplice	Media	Input	
IsVerificato	Semplice	Media	Input	
Username Utente	Semplice	Alta	Input	
Username Amministratore	Semplice	Alta	Input	

### ScritturaLog – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Data	Semplice	Media	Input	
Ora	Semplice	Media	Input	
Operazione Eseguita	Composto	Media	Input	

### VisualizzaLog – Tabella Informazioni/Flusso

Informazioni	Tipo	Livello protezione / Privacy	Input/ Output	Vincoli
Data	Semplice	Media	Output	
Ora	Semplice	Media	Output	
Operazione Eseguita	Composto	Media	Output	

# Analisi Dei Vincoli

## Tabella Dei Vincoli

Requisito	Categoria	Impatto	Funzionalità
Integrità dei dati	Integrità	Garantisce maggiore protezione e qualità dei dati ma ne peggiora la velocità di scrittura e di trasmissione	Tutte
Protezione dei dati	Sicurezza	Aumenta il tempo di risposta del <i>Sistema</i> ma migliora la privacy dei dati	Tutte
Controllo accessi	Sicurezza	Causa rallentamento del Sistema ma migliora la privacy dei dati	Autenticazione
Rapidità di Ricerca	Tempo di risposta	Diminuisce il tempo di risposta del Sistema a seguito di una ricerca	Ricerca
Facilità di navigazione delle schermate	Usabilità	Rende il Sistema più semplice da utilizzare	Tutte
Sistema di pagamento esterno	Sicurezza	Massimizzare l'affidabilità in fase di pagamento, ridurre la quantità di dati sensibili da memorizzare	PrenotaScambio

# Scomposizione del problema

## Tabella Scomposizione Funzionalità

Funzionalità	Scomposizione
PrenotaScambio	RichiediScambio, ConfermaScambio
Segnalazione	EffettuaSegnalazione, VerificaSegnalazione

## Tabelle Sotto-Funzionalità

### PrenotaScambio - Tabella Sotto-funzionalità

Sotto-funzionalità	Sotto-funzionalità	Legame	Informazioni
RichiediScambio	ConfermaScambio	Le richieste hanno bisogno di essere confermate per prenotare lo <i>Scambio</i>	Tutte le informazioni <i>Utente-Pari</i> relative allo <i>Scambio</i>

### Segnalazione - Tabella Sotto-funzionalità

Sotto-funzionalità	Sotto-funzionalità	Legame	Informazioni
Effettua Segnalazione	Verifica Segnalazione	Le <i>Segnalazioni</i> effettuate dagli <i>Utenti</i> devono essere verificate da un <i>Amministratore</i>	UsernameUtente, IdSegnalazione, UsernameAmministratore

# Analisi Delle Interazioni

## Tabella maschere

Maschera	Informazioni	Funzionalità
Home Principale	Messaggio di benvenuto, possibilità di navigazione verso altre schermate	Autenticazione, RegistrazioneUtente
View Autenticazione	Form per inserire Username e Password per il login	Autenticazione
View RegistrazioneUtente	Form per procedere alla RegistrazioneUtente	RegistrazioneUtente
Home UtenteNonAbilitato	Avviso che bisogna registrare almeno un' <i>Abitazione</i> per essere abilitati	RegistraAbitazione
View RegistraAbitazione	Form per procedere alla registrazione dell' <i>Abitazione</i>	RegistraAbitazione
Home UtenteAbilitato	Lista delle informazioni relative all' <i>Utente</i> e possibilità di navigazione verso altre schermate	Ricerca, Storico, PrenotaScambio, Recensione, Segnalazione
View Ricerca	Form per la <i>Ricerca</i> di <i>Abitazioni</i> , possibilità di mettere <i>Like</i>	Ricerca, EsprimiInteresse
View Storico	Lista dei <i>Viaggi</i> effettuati	Storico
View Recensione	Lista dei <i>Viaggi</i> da poter recensire	Recensione
View RichiediScambio	Lista dei <i>Match</i> dell' <i>Utente</i> selezionabili per richiedere uno <i>Scambio</i>	PrenotaScambio



View ConfermaScambio	Lista delle richieste di <i>Scambio</i> ricevute	PrenotaScambio
View EffettuaSegnalazione	Form da compilare per inoltrare una <i>Segnalazione</i>	Segnalazione
Home Amministratore	Lista delle informazioni dell' <i>Amministratore</i> e possibilità di navigazione verso altre schermate	VerificaAbitazione, EliminaUtente, Segnalazione
View VerificaAbitazione	Lista delle <i>Abitazioni</i> da verificare	VerificaAbitazione
View EliminaUtente	Lista di tutti gli <i>Utenti</i> del <i>Sistema</i> , possibilità di eliminazione	EliminaUtente
View VerificaSegnalazione	Lista delle <i>Segnalazioni</i> da verificare	Segnalazione

## Tabella dei sistemi esterni

Sistema	Descrizione	Protocollo di interazione	Livello di sicurezza
Pagamento	Sistema che si occupa di gestire le transazioni relative ai pagamenti degli <i>Scambi</i>	Restituire un messaggio di conferma che notifica dell'avvenuto pagamento	Molto alto perché gestisce dati estremamente sensibili come i numeri delle carte di credito

# Analisi dei ruoli e delle responsabilità

## Tabella Ruoli

Ruolo	Responsabilità	Maschere	Riservatezza	Numerosità
Utente	Registra le <i>Abitazioni</i> , effettua una ricerca, esprime interesse, prenota <i>Scambi</i> , rilascia recensioni, visualizza lo <i>Storico</i>	Home Principale, View Autenticazione, Home UtenteNonAbilitato, View RegistraAbitazione, Home UtenteAbilitato, View Ricerca, View Storico, View Recensione, View RichiediScambio, View ConfermaScambio View EffettuaSegnalazio- ne	E' richiesto un grado di riservatezza molto alto	Potenzialmente infinito, limitato dalla sola disponibilità e dalle risorse del <i>Sistema</i>
Amministratore	Verifica le <i>Abitazioni</i> e le <i>Segnalazioni</i> , elimina gli <i>Utenti</i>	Home Principale, Home Amministratore, VerificaAbitazione, EliminaUtente, View VerificaSegnalazione	E' richiesto un grado di riservatezza molto alto	3-10 persone, variabile a seconda delle esigenze di gestione

## Tabelle Ruolo-Informazioni

### Utente: Tabella Ruolo-Informazioni

Informazione	Tipo di Accesso
Nome	Scrittura
Cognome	Scrittura
DataDiNascita	Scrittura
DocumentIdentificazione	Scrittura
NumeroCellulare	Scrittura
Email	Scrittura
Password	Scrittura
DataInizioViaggio/DataFineViaggio	Scrittura/Lettura
DataInizioDisponibilità/DataFineDisponibilità	Scrittura/Lettura
Like	Scrittura/Lettura
Storico	Lettura
Abitazioni	Lettura/Scrittura
Recensioni	Lettura/Scrittura

### Amministratore: Tabella Ruolo-Informazioni

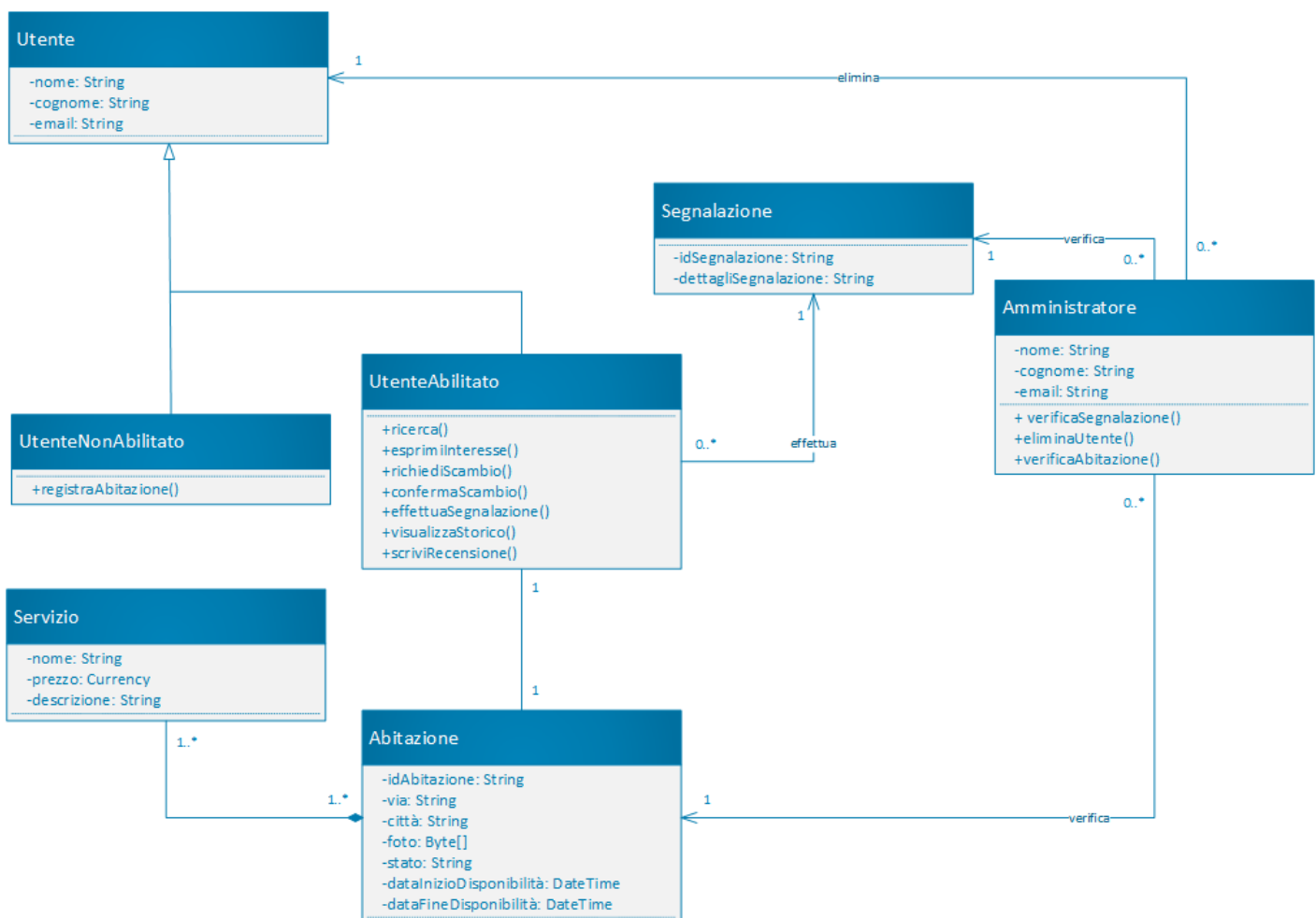
Informazione	Tipo di Accesso
Username	Scrittura
Password	Scrittura
Abitazioni	Lettura/Scrittura

Utenti	Lettura/Scrittura
Log (*)	Lettura

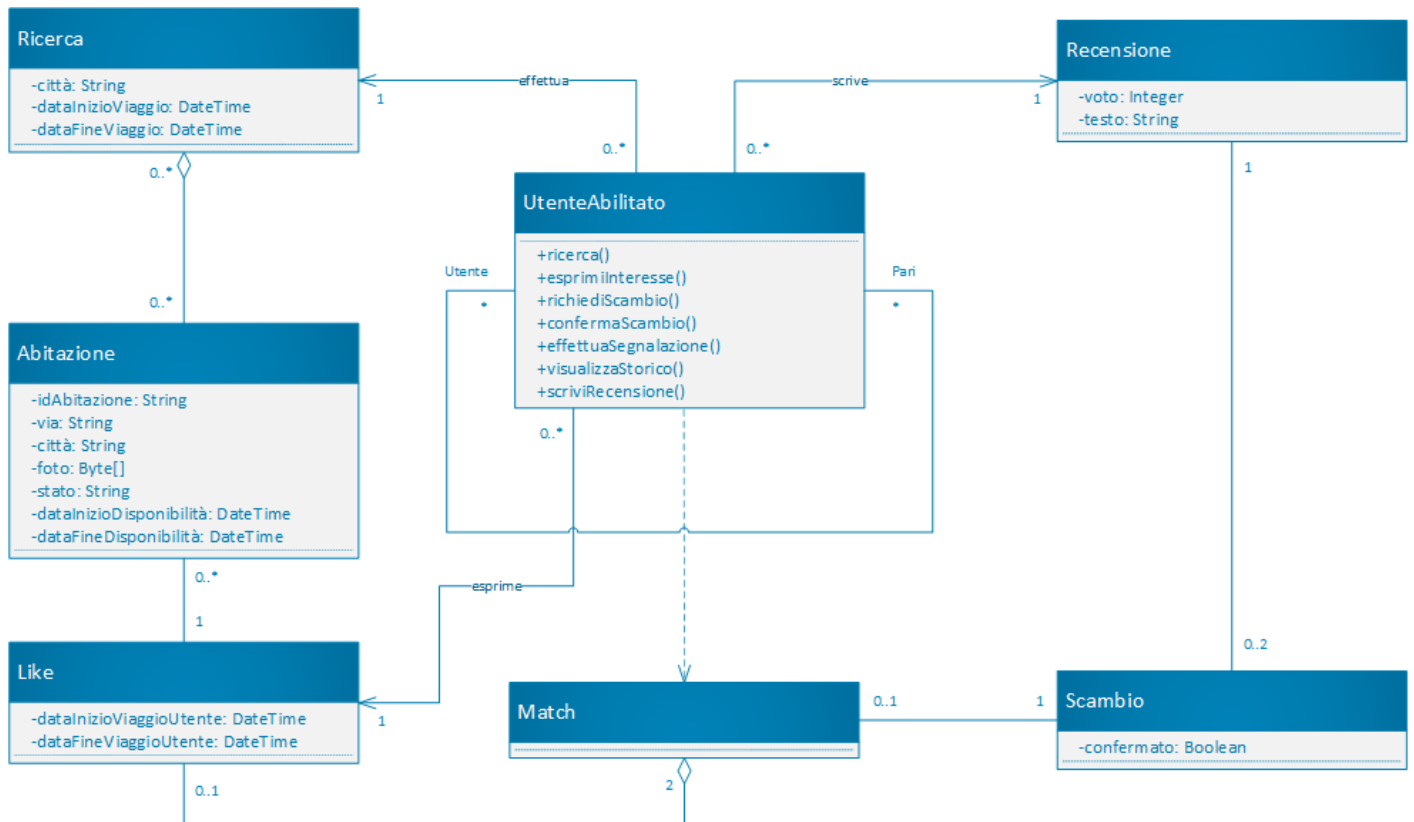
(\*) L'Amministratore ha la possibilità di visualizzare i log non da una schermata apposita del Sistema ma mediante un editor di testo esterno.

## Modello del Dominio

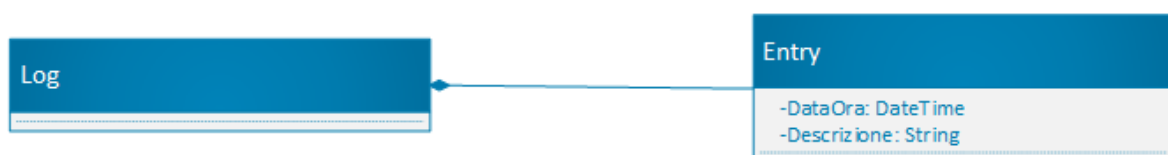
Il seguente diagramma rappresenta le interazioni degli *Utenti* e degli *Amministratori* relative alle *Abitazioni* e alle *Segnalazioni*.



Il seguente diagramma rappresenta le interazioni tra gli *Utenti* nella gestione delle *Preferenze* e degli *Scambi*. Il vincolo che un *Utente Abilitato* possa prenotare uno *Scambio* solo dopo aver effettuato un *Match* è espresso dalla relazione Match-Scambio. L'attributo “*confermato*” dello *Scambio* indica se è stato confermato o meno dal *Pari*.



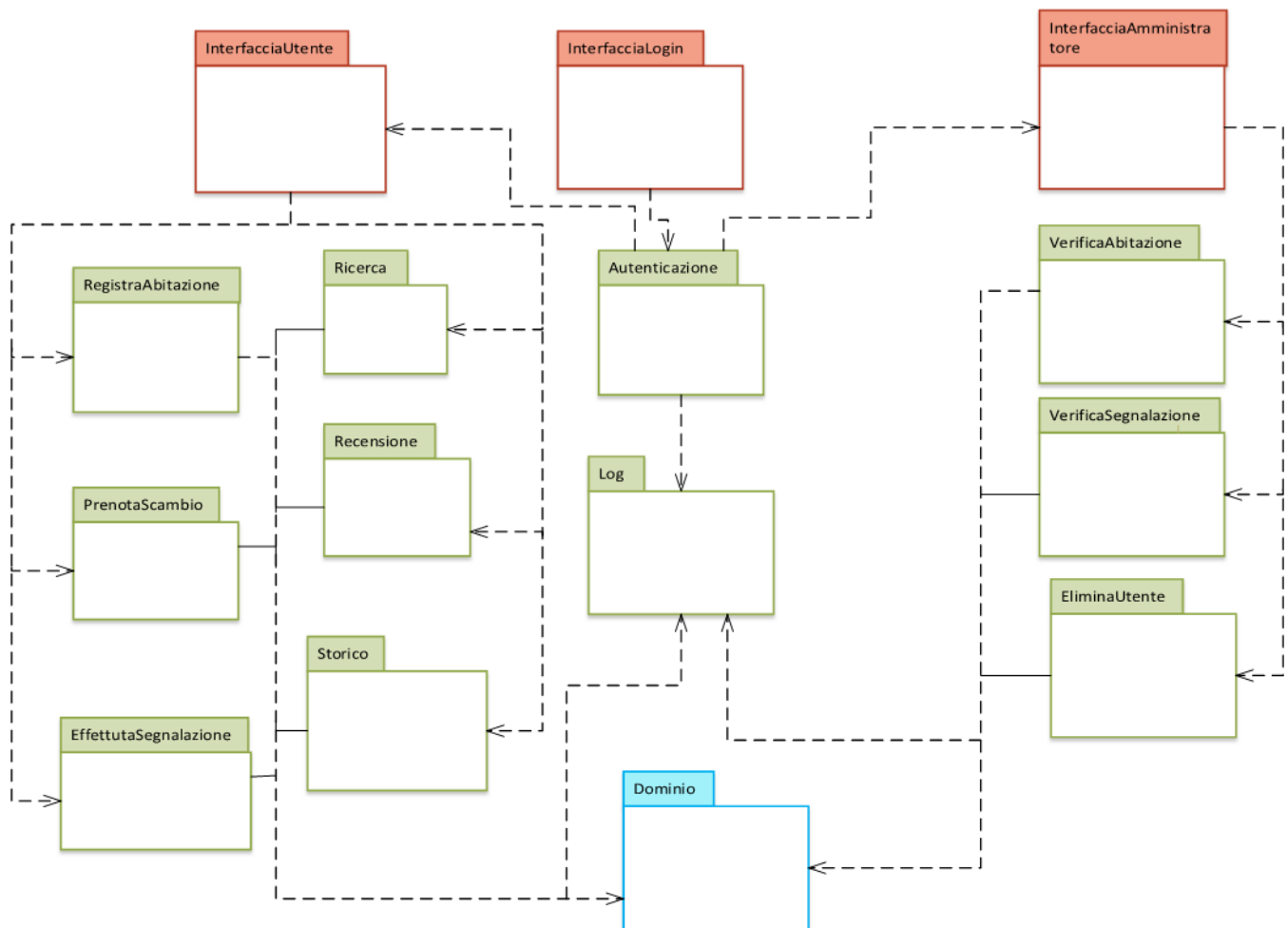
Il seguente diagramma rappresenta la parte di modello di dominio relativa alla gestione di *Log*.



# Architettura Logica

## Struttura

### Diagramma dei package

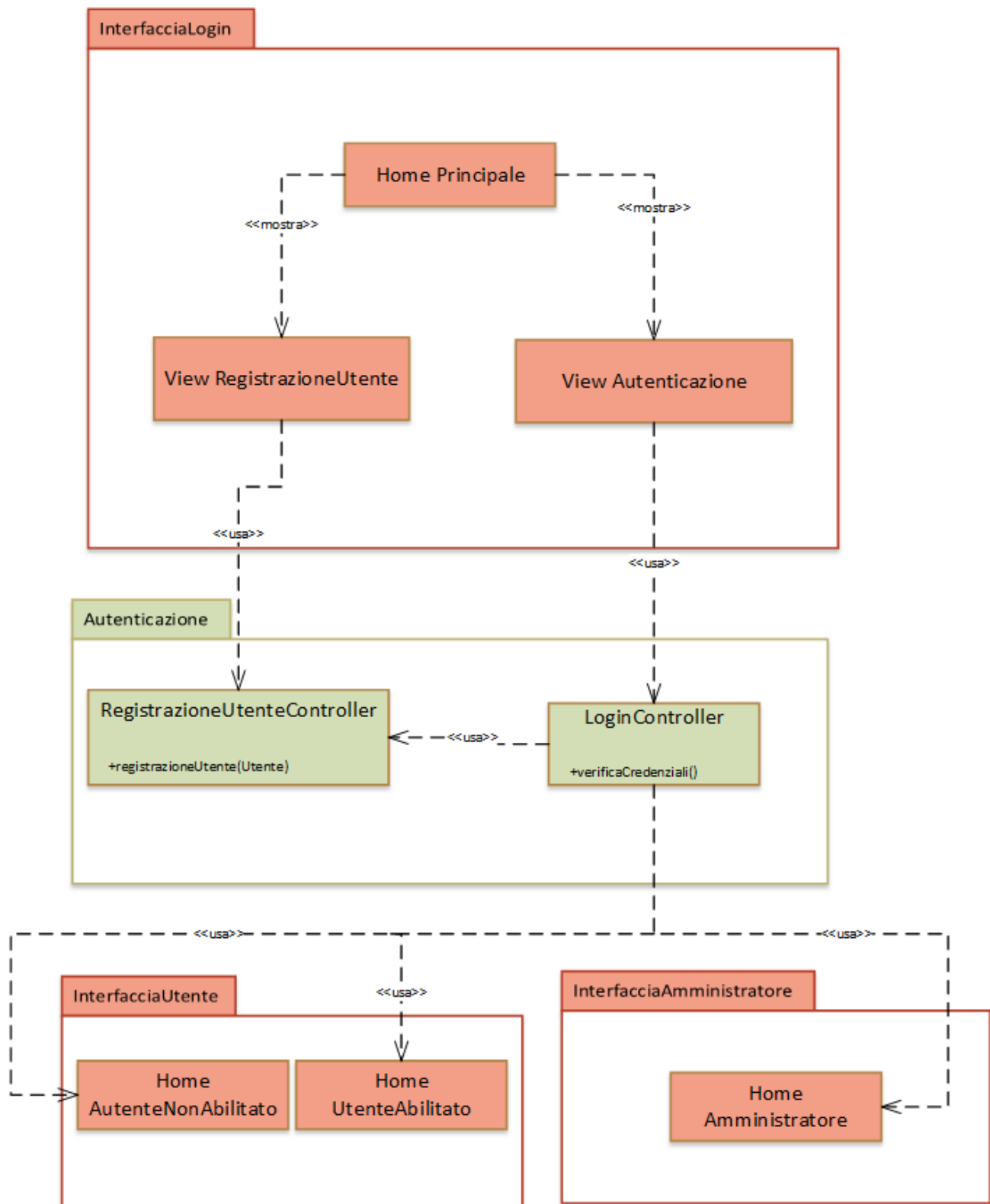


## Diagramma delle classi

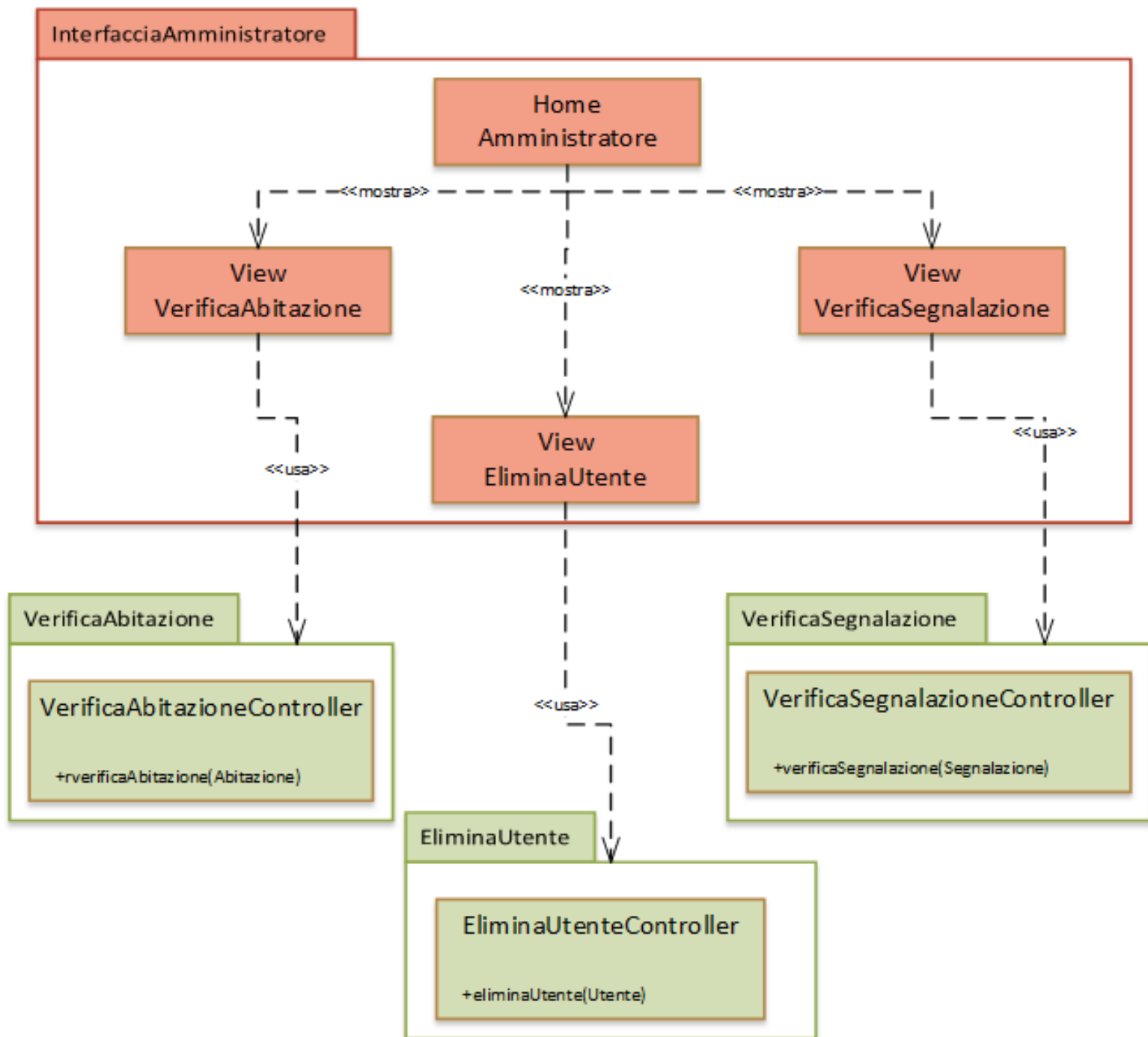
### Diagramma delle classi: package Dominio

Non viene riportato il diagramma delle classi associato al package Dominio in quanto è il modello del dominio creato nella fase precedente

## Diagramma delle classi: Interfaccia Login

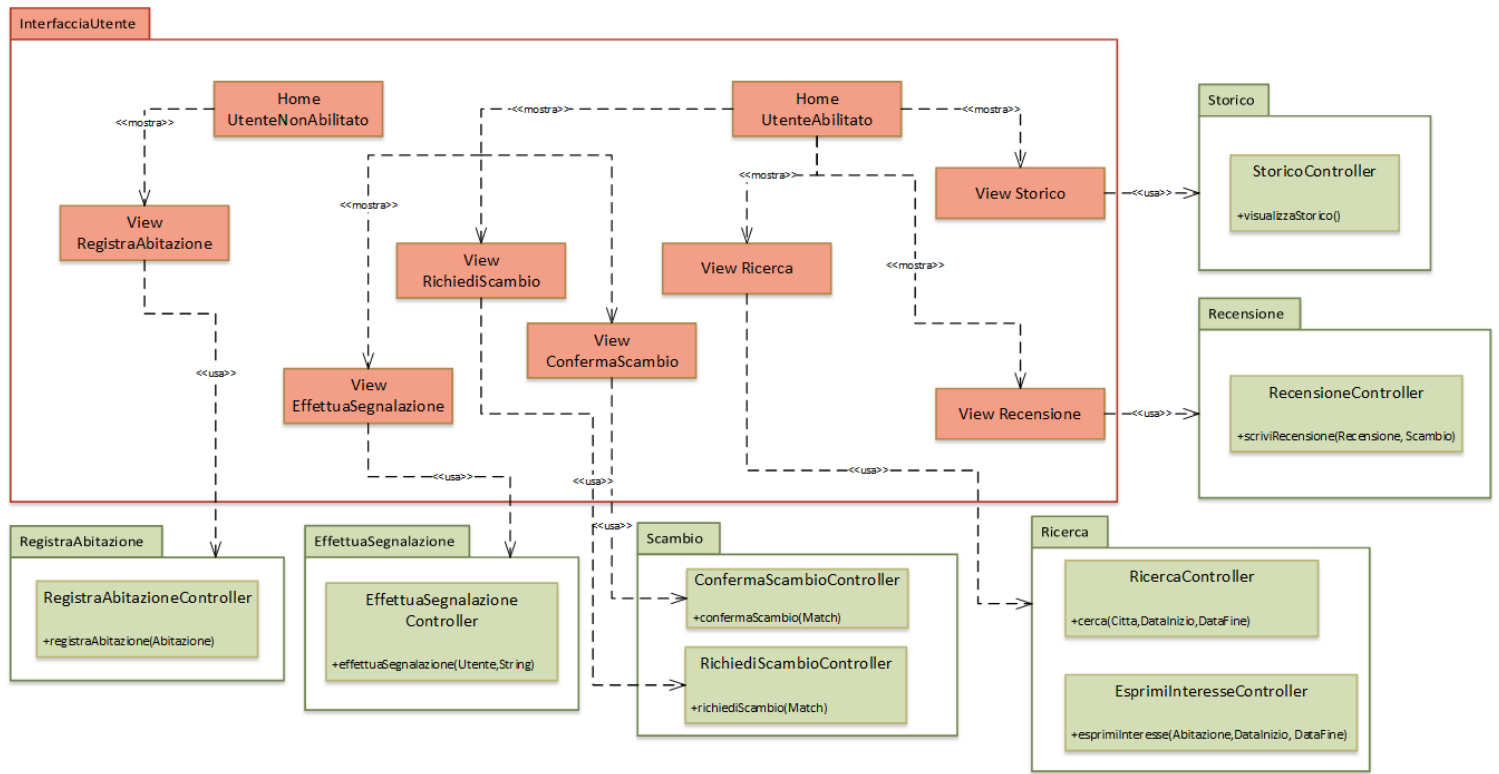


## Diagramma delle classi: Interfaccia Amministratore



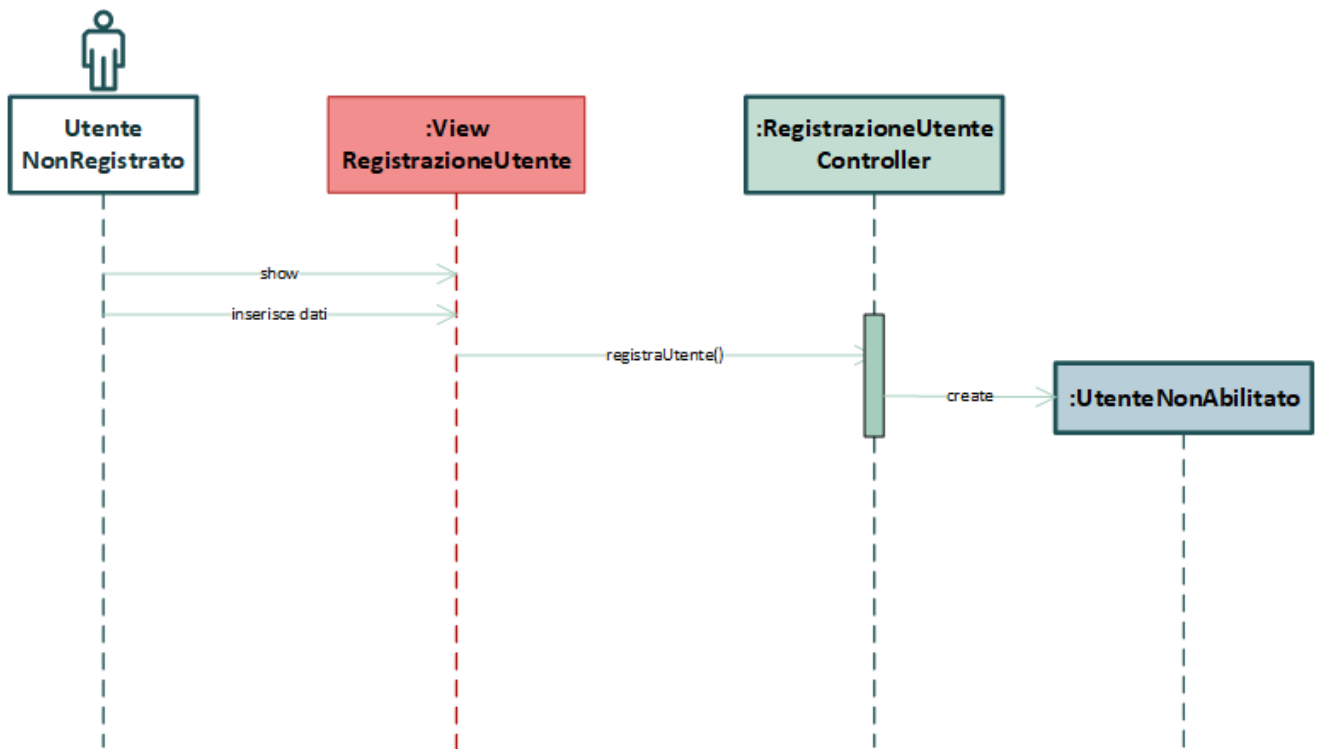


## Diagramma delle classi: Interfaccia Utente

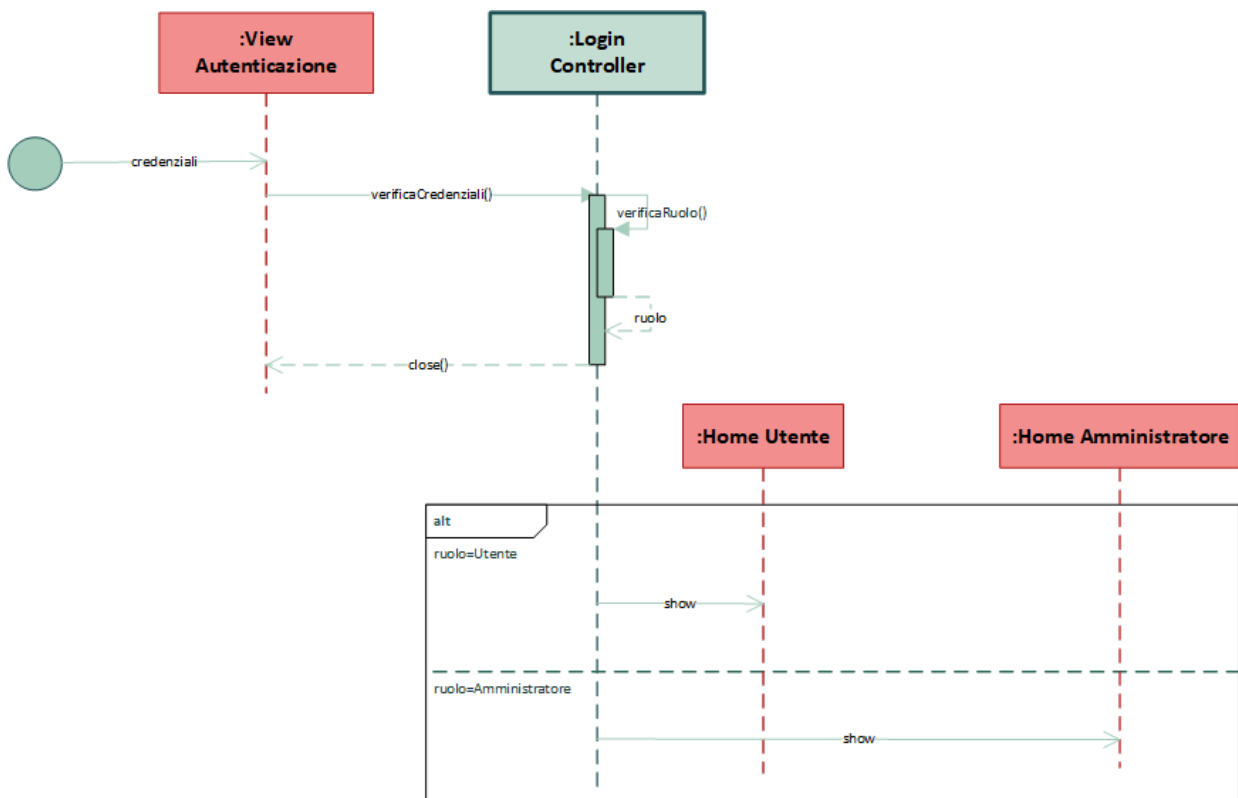


## Interazione

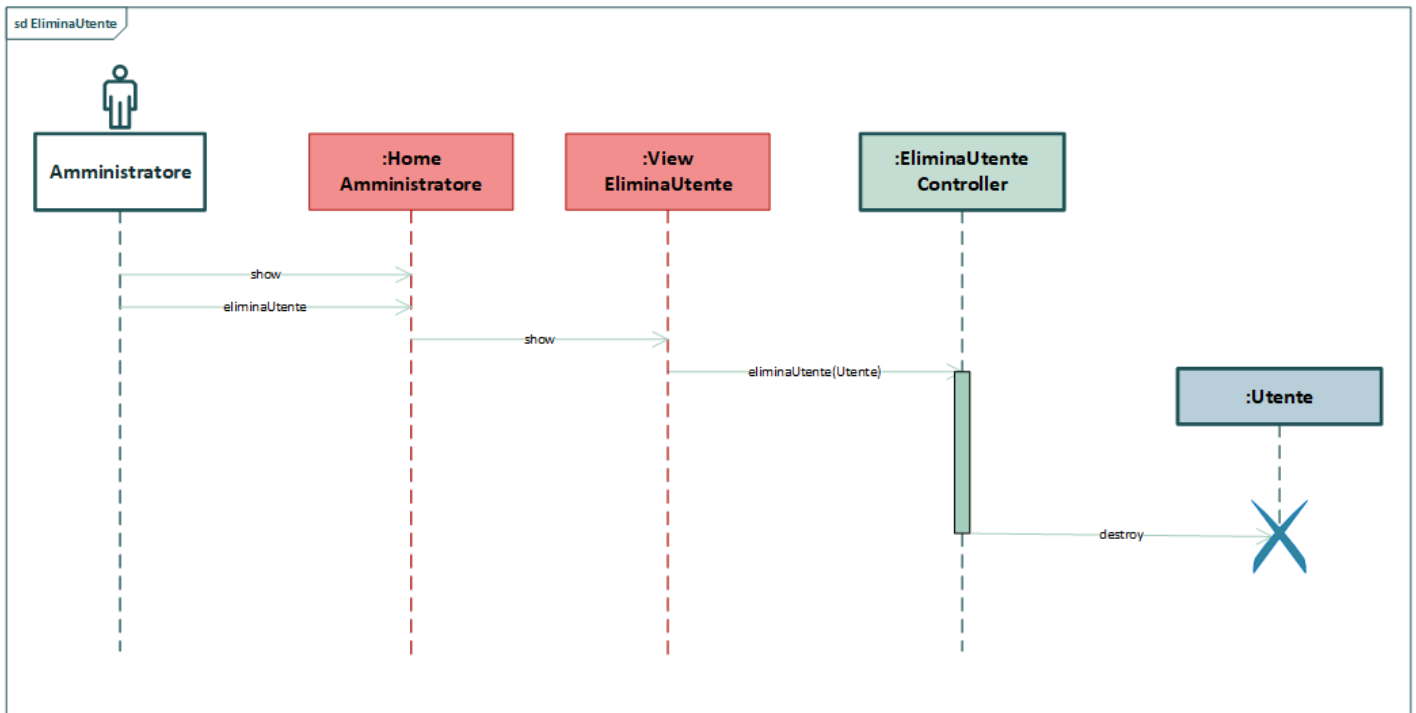
### Diagramma di Sequenza : RegistrazioneUtente



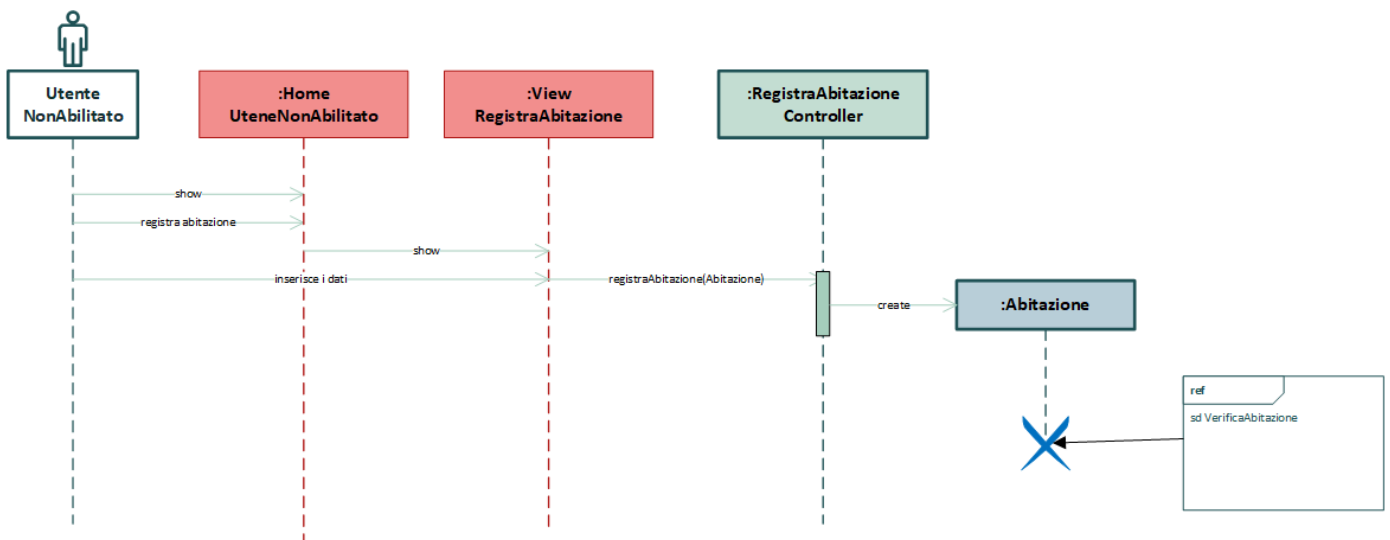
### Diagramma di Sequenza : Autenticazione



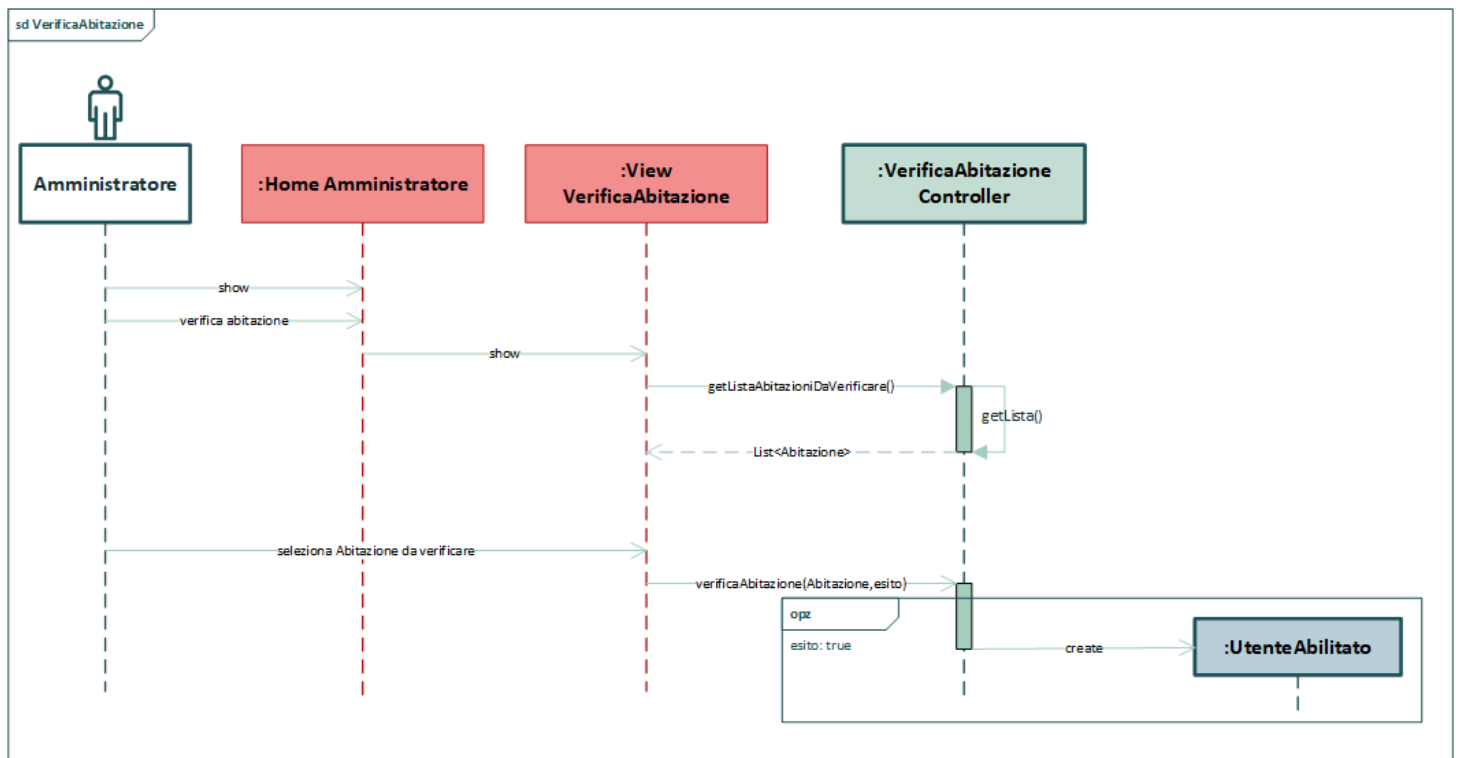
## Diagramma di Sequenza : EliminaUtente



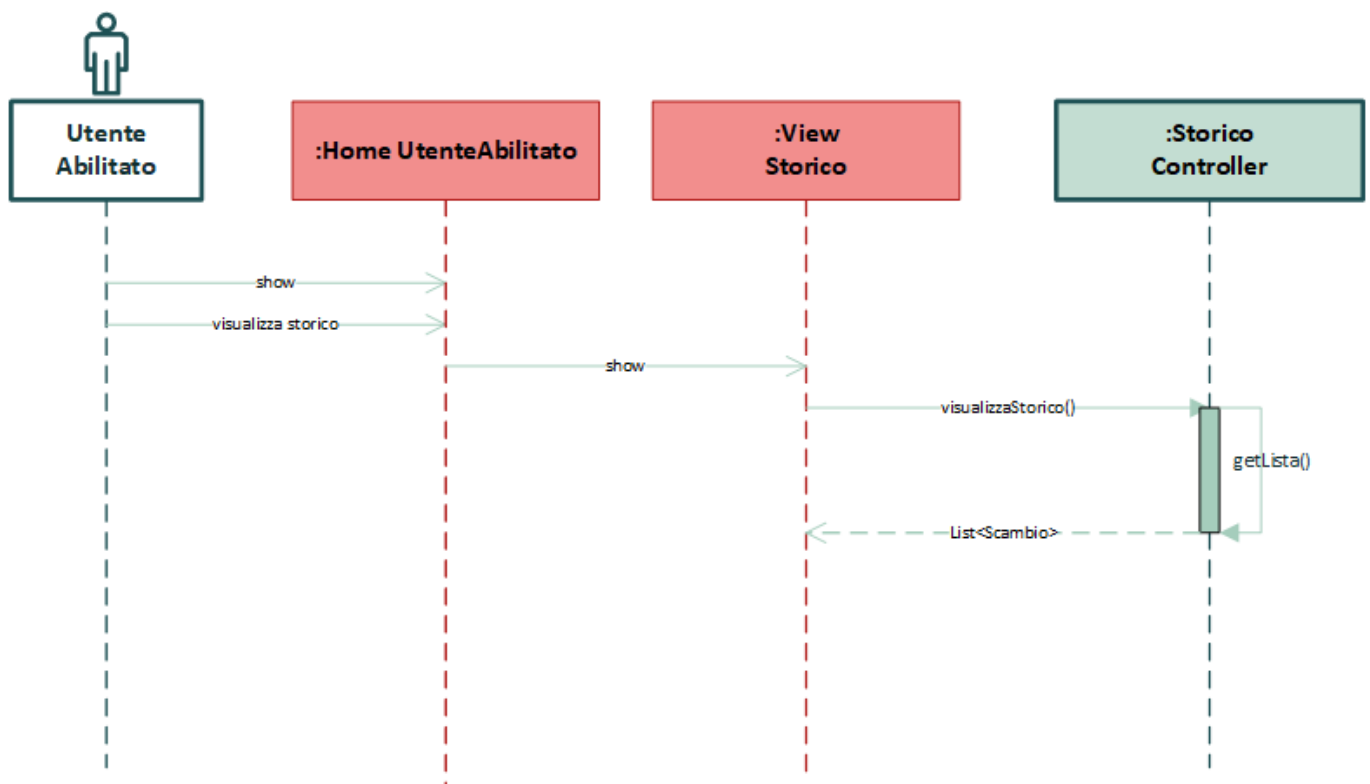
## Diagramma di Sequenza : RegistraAbitazione



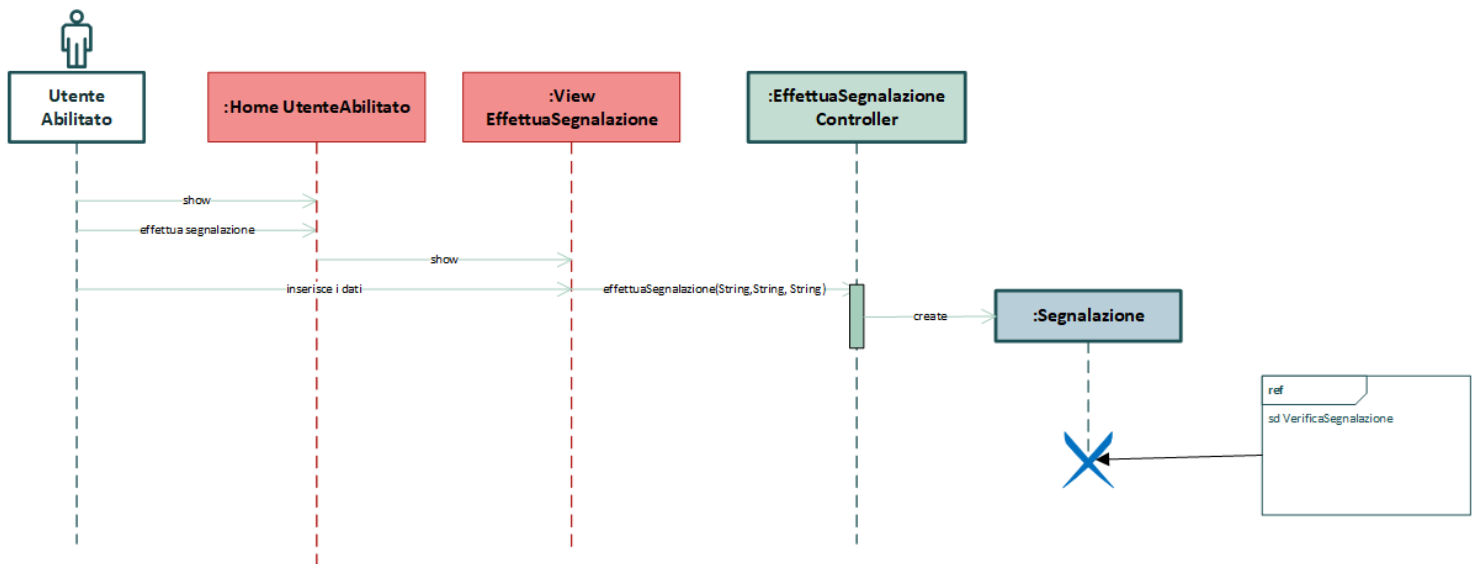
## Diagramma di Sequenza : VerificaAbitazione



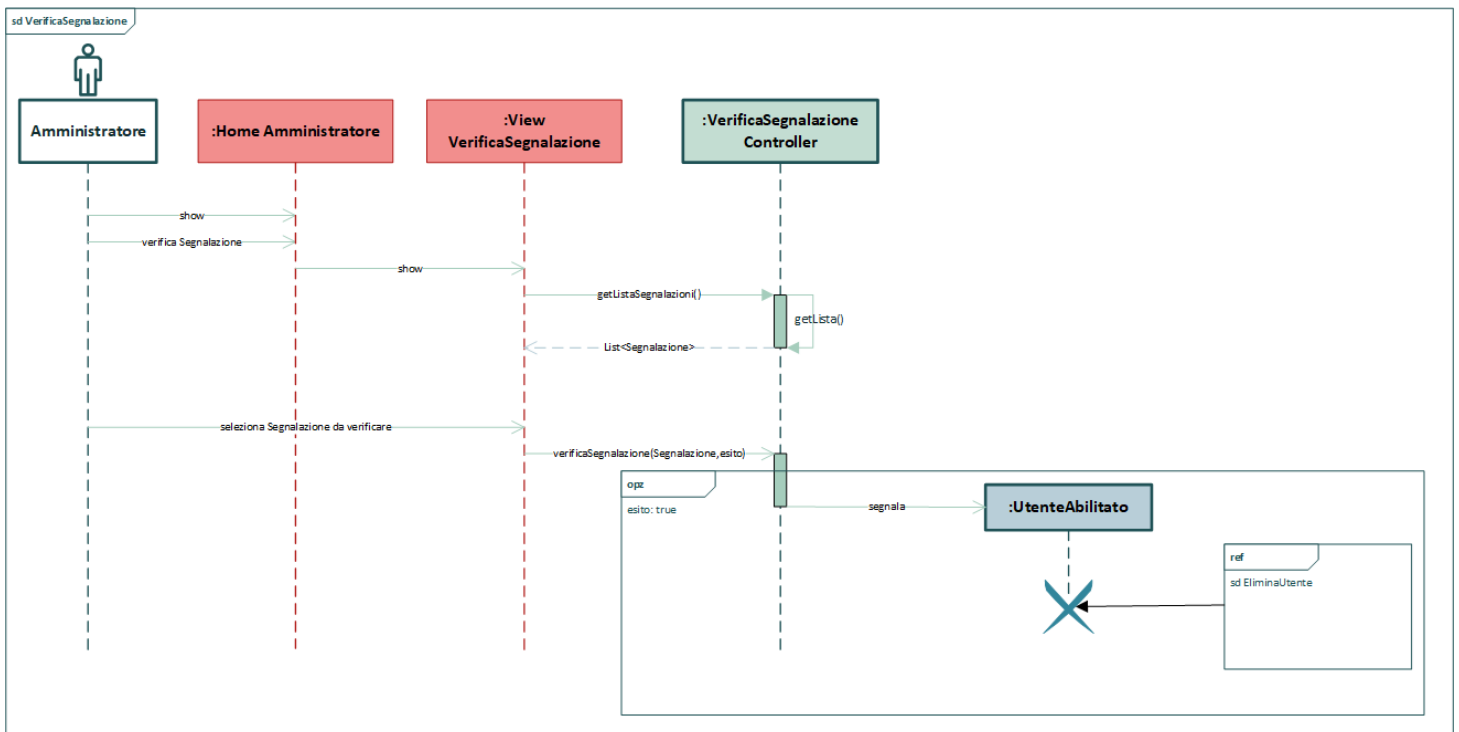
## Diagramma di Sequenza : Storico



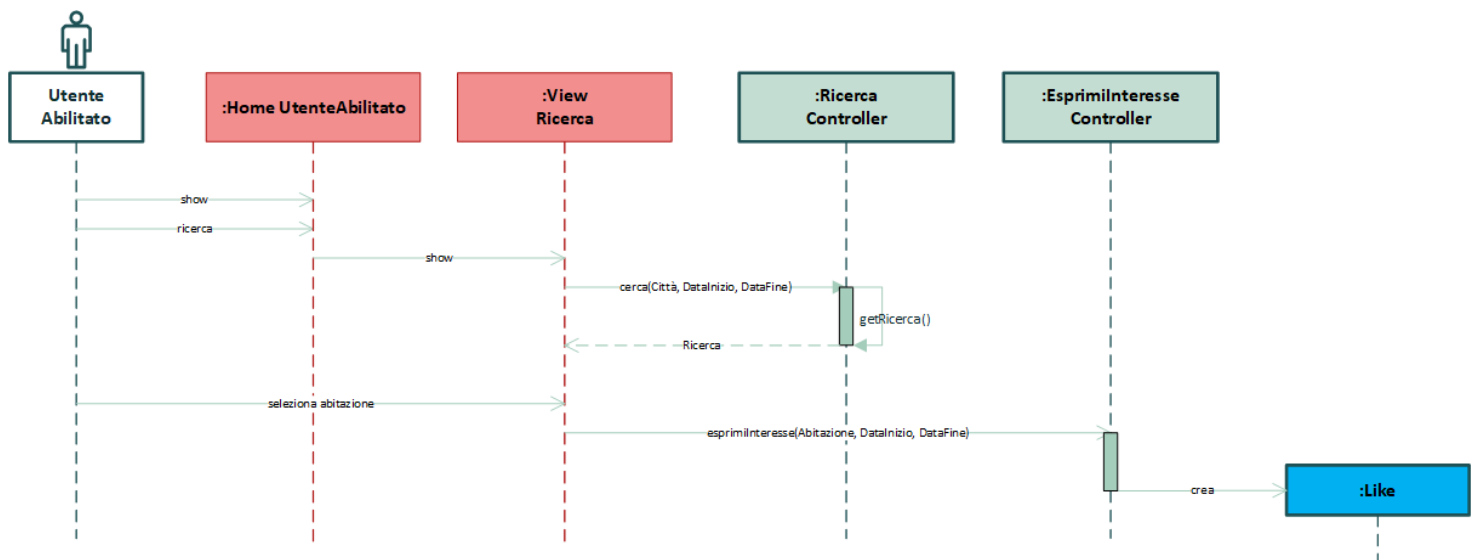
### Diagramma di Sequenza : EffettuaSegnalazione



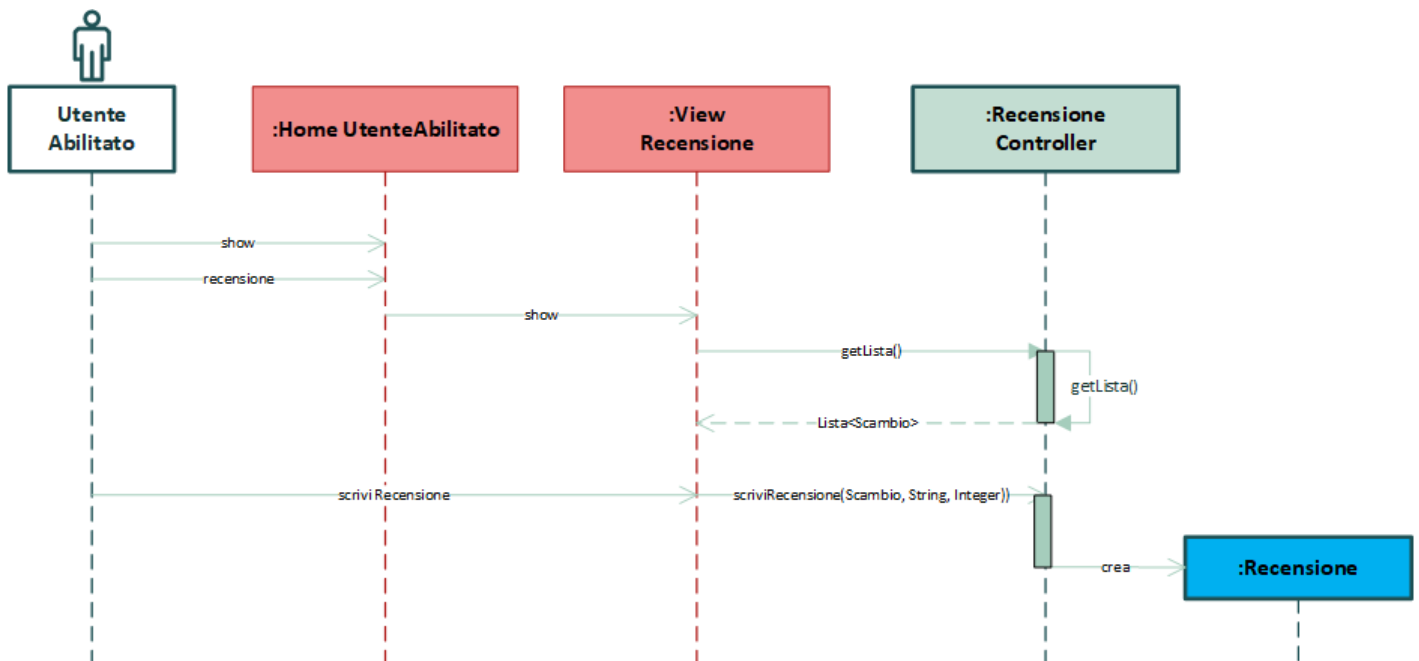
### Diagramma di Sequenza : VerificaSegnalazione



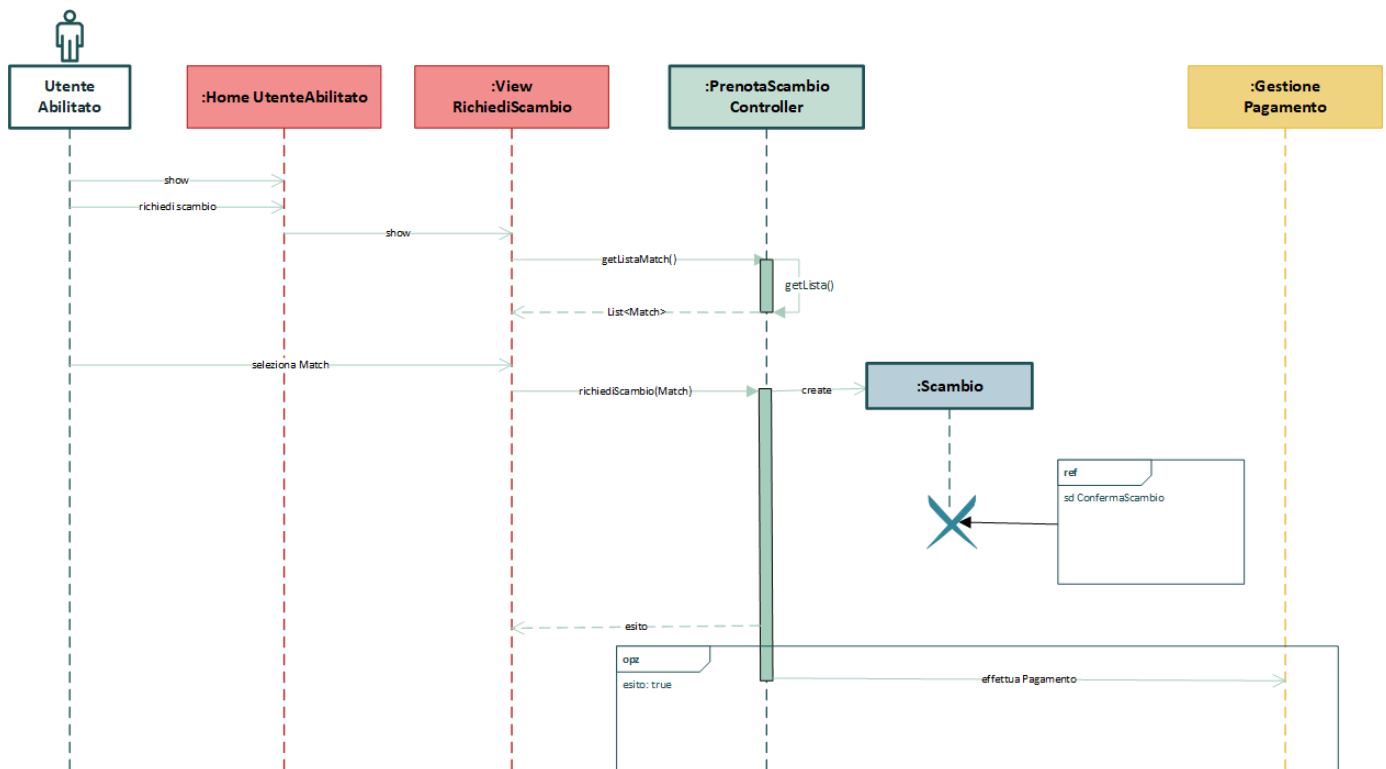
## Diagramma di Sequenza : Ricerca



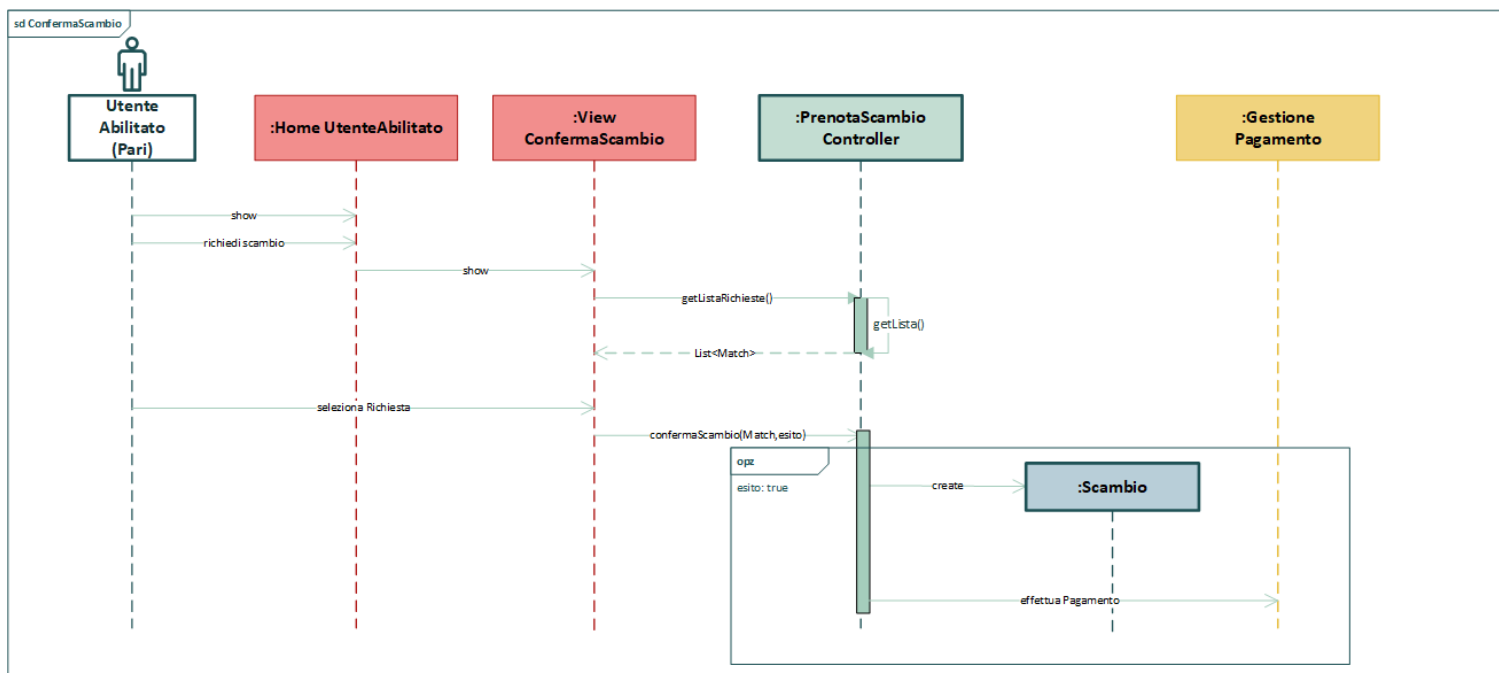
## Diagramma di Sequenza : Recensione



## Diagramma di Sequenza : PrenotaScambio



Lo *Scambio* che viene creato a seguito di una richiesta presenta come attributo “confermato : false”.



## Comportamento

Dopo un'attenta riflessione, abbiamo ritenuto che nessuna entità richieda un diagramma di stato.

## Piano di Lavoro

Package	Progetto	Sviluppo
Dominio	Noviello	Noviello
InterfacciaUtente	Di Domenico	Di Domenico
InterfacciaLogin	Di Battista	Di Battista
Interfaccia Amministratore	Di Battista	Di Battista
Autenticazione	Di Domenico,Di Battista,Noviello	Di Domenico, Noviello
RegistraAbitazione	Di Domenico,Di Battista,Noviello	Di Domenico
VerificaAbitazione	Di Domenico,Di Battista,Noviello	Noviello
Ricerca	Di Domenico,Di Battista,Noviello	Di Battista
EsprimiInteresse	Di Domenico,Di Battista,Noviello	Noviello, Di Domenico
Scambio	Di Domenico,Di Battista,Noviello	Di Battista
Storico	Di Domenico,Di Battista,Noviello	Di Battista
EffettuaSegnalazione	Di Domenico,Di Battista,Noviello	Noviello
VerificaSegnalazione	Di Domenico,Di Battista,Noviello	Noviello
EliminaUtente	Di Domenico,Di Battista,Noviello	Di Domenico
Log	Di Domenico,Di Battista,Noviello	Di Domenico

Dopo un'accurata stima dei tempi di sviluppo, i tempi di rilascio previsti sono:

- Progettazione: entro 14 giorni dalla data odierna
- Sviluppo dei vari moduli con annessi test unitari: entro 14 giorni dalla fine della progettazione
- Integrazione e testing: entro 7 giorni dalla fine della fase di sviluppo



## Sviluppi Futuri

Il committente ha richiesto che in futuro vengano inserite all'interno del *Sistema* le seguenti funzionalità:

- Registrazione di più Abitazioni per ogni Utente
- Possibilità di modifica delle Abitazioni

## Piano di Collaudo

Per garantire il corretto funzionamento del sistema sono necessari una gamma di test unitari e di integrazione che permettano la correttezza delle singole parti. Di seguito vengono riportati i casi più di interesse.

```
public class TestUtenteAbilitato {

    private Servizio servizio;
    private Abitazione abitazione;
    private UtenteAbilitato user;

    @Before
    public void setUp() {
        this.servizio = new Servizio();
        this.abitazione = new Abitazione();
        this.user = new UtenteAbilitato();
    }

    @Test
    public void testCostruttore() {
        // Servizio
        Assert.assertNull(servizio.getNome());
        Assert.assertNull(servizio.getDescrizione());
        Assert.assertEquals(servizio.getPrezzo(), -1.00, 0.00);
        // Abitazione
        Assert.assertNull(abitazione.getID());
        Assert.assertNull(abitazione.getVia());
        Assert.assertNull(abitazione.getCitta());
        Assert.assertNull(abitazione.getStato());
        Assert.assertNull(abitazione.getFoto());
        Assert.assertEquals(abitazione.getPrezzo(), -1.00, 0.00);
        Assert.assertEquals(abitazione.getServizi().isEmpty(), true);
        Assert.assertNull(abitazione.getDataInizioDisponibilita());
        Assert.assertNull(abitazione.getDataFineDisponibilita());
        Assert.assertEquals(abitazione.getPrenotazioni().isEmpty(), true);
        Assert.assertNull(abitazione.getUsername());
        // Utente
    }
}
```



```

        Assert.assertNull(user.getNome());
        Assert.assertNull(user.getCognome());
        Assert.assertNull(user.getEmail());
        Assert.assertNull(user.getDataDiNascita());
        Assert.assertNull(user.getDocumento());
        Assert.assertNull(user.getNumeroCellulare());
        Assert.assertNull(user.getPassword());
        Assert.assertNull(user.getAbitazione());
    }

    @Test
    public void testSetterMethods() {
        String[] Foto = {"path1", "path2", "path3"};
        LocalDate dataNascita = LocalDate.of(1963, 8, 3);
        LocalDate dataInizioDisponibilita = LocalDate.of(2021, 6, 1);
        LocalDate dataFineDisponibilita = LocalDate.of(2021, 6, 30);
        // Servizio
        servizio.setNome("Gita");
        servizio.setDescrizione("Passeggiata nel verde dei colli bolognesi");
        servizio.setPrezzo(10.5);

        Assert.assertEquals(servizio.getNome(), "Gita");
        Assert.assertEquals(servizio.getDescrizione(),
            "Passeggiata nel verde dei colli bolognesi");
        Assert.assertEquals(servizio.getPrezzo(), 10.50, 0.00);

        // Abitazione
        abitazione.setVia("Viale del Risorgimento 2");
        abitazione.setCitta("Bologna");
        abitazione.setStato("Italia");
        abitazione.setDataInizioDisponibilita(dataInizioDisponibilita);
        abitazione.setDataFineDisponibilita(dataFineDisponibilita);
        abitazione.setFoto(Foto);
        abitazione.setServizio(servizio);
        abitazione.setUsername("james.montana@unibo.it");

        Assert.assertEquals(abitazione.getVia(), "Viale del Risorgimento 2");
        Assert.assertEquals(abitazione.getCitta(), "Bologna");
        Assert.assertEquals(abitazione.getStato(), "Italia");
        Assert.assertArrayEquals(abitazione.getFoto(), Foto);
        Assert.assertEquals(user.getDataInizioDisponibilita(),
            dataInizioDisponibilita);
        Assert.assertEquals(user.getDataIFineDisponibilita(),
            dataFineDisponibilita);
        Assert.assertEquals(abitazione.getServizi().get(0), servizio);
        Assert.assertEquals(abitazione.getPrezzo(), 10.50, 0.00);
        Assert.assertEquals(abitazione.getUsername(),
            "james.montana@unibo.it");

        // Utente
        user.setNome("James");
    }

```



```

        user.setCognome("Montana");
        user.setEmail("james.montana@unibo.it");
        user.setDataDiNascita(data);
        user.setDocumento("CA00000AA");
        user.setNumeroCellulare("+39321123456");
        user.setPassword("password");
        user.setAbitazione(abitazione);

        Assert.assertEquals(user.getNome(), "James");
        Assert.assertEquals(user.getCognome(), "Montana");
        Assert.assertEquals(user.getEmail(), "james.montana@unibo.it");
        Assert.assertEquals(user.getDataDiNascita(), data);
        Assert.assertEquals(user.getDocumento(), "CA00000AA");
        Assert.assertEquals(user.getNumeroCellulare(), "+39321123456");
        Assert.assertEquals(user.getPassword(), "password");
        Assert.assertEquals(user.getAbitazione(), abitazione);
    }
}

```

# Progettazione

## Requisiti Non Funzionali

Dall'Analisi del Problema sono emersi i seguenti requisiti non funzionali:

- Integrità dei dati
- Sicurezza dei dati
- Controllo degli accessi
- Facilità d'utilizzo
- Rapidità di ricerca

L'Integrità e la Sicurezza dei dati sono requisiti fondamentali del *Sistema* in quanto, facendo largo uso del trattamento di dati sensibili degli *Utenti*, è necessario preservarli. A tal fine, si è scelto di utilizzare un protocollo **TLS** per garantire una comunicazione sicura.

Il Controllo degli accessi impedisce ad *Utenti* non autorizzati di manomettere il *Sistema*.

La Facilità d'utilizzo è importante perché permette di offrire un servizio anche ad *Utenti* con scarse conoscenze informatiche.

Essendo la *Ricerca* uno dei punti chiave del progetto, è necessario massimizzarne la rapidità visto il costante utilizzo.

# Scelta dell'Architettura

Dal punto di vista architettuale, l'architettura ritenuta più appropriata per il *Sistema* è quella Client/Server a 3 livelli.

Di seguito sono riportate le descrizioni dei vari livelli:

## L1-Client

Il primo livello è costituito dai seguenti Client:

- Un Client per l'*Utente*
- Un Client per l'*Amministratore*

I Client non hanno mai accesso diretto al Database.

## L2-Server

Il secondo livello è costituito dai seguenti Server:

- Un Server per le funzionalità dell'*Utente*
- Un Server per le funzionalità di gestione dell'*Amministratore*

## L3-Persistenza

Il terzo livello riguarda la gestione della persistenza ed è costituito da:

- Un DBMS per gestire i dati del Sistema, che verrà installato su un apposito server
- Un server per la memorizzazione dei Log

L'interfacciamento con il DBMS avverrà attraverso la metodologia "**Forza Bruta**", che permette di risolvere il conflitto tra il linguaggio ad oggetti utilizzato per l'implementazione e il formato della fonte dei dati.

# Pattern & Design Principle

Per offrire maggiore scalabilità, si è deciso di adottare il pattern MVC, che permette di suddividere il Sistema in tre parti:

## Model

Gestisce i dati e le entità del Sistema:

- *Utente*
- *Amministratore*
- *Abitazione*
- *Like*
- *Match*
- *Scambio*
- *Segnalazione*

## View

Visualizza i dati gestiti dal Model e le azioni del *Sistema* attraverso le viste definite nella “Tabella delle Maschere”.

## Controller

Manipola i dati ed ha il compito di eseguire le funzionalità richieste:

- *Registrazione Utente*
- *Registra Abitazione*
- *Esprimi Interesse*
- *Prenota Scambio*
- *Recensione*
- *Visualizza Storico*
- *Effettua Segnalazione*
- *Verifica Abitazione*
- *Verifica Segnalazione*
- *Elimina Utente*

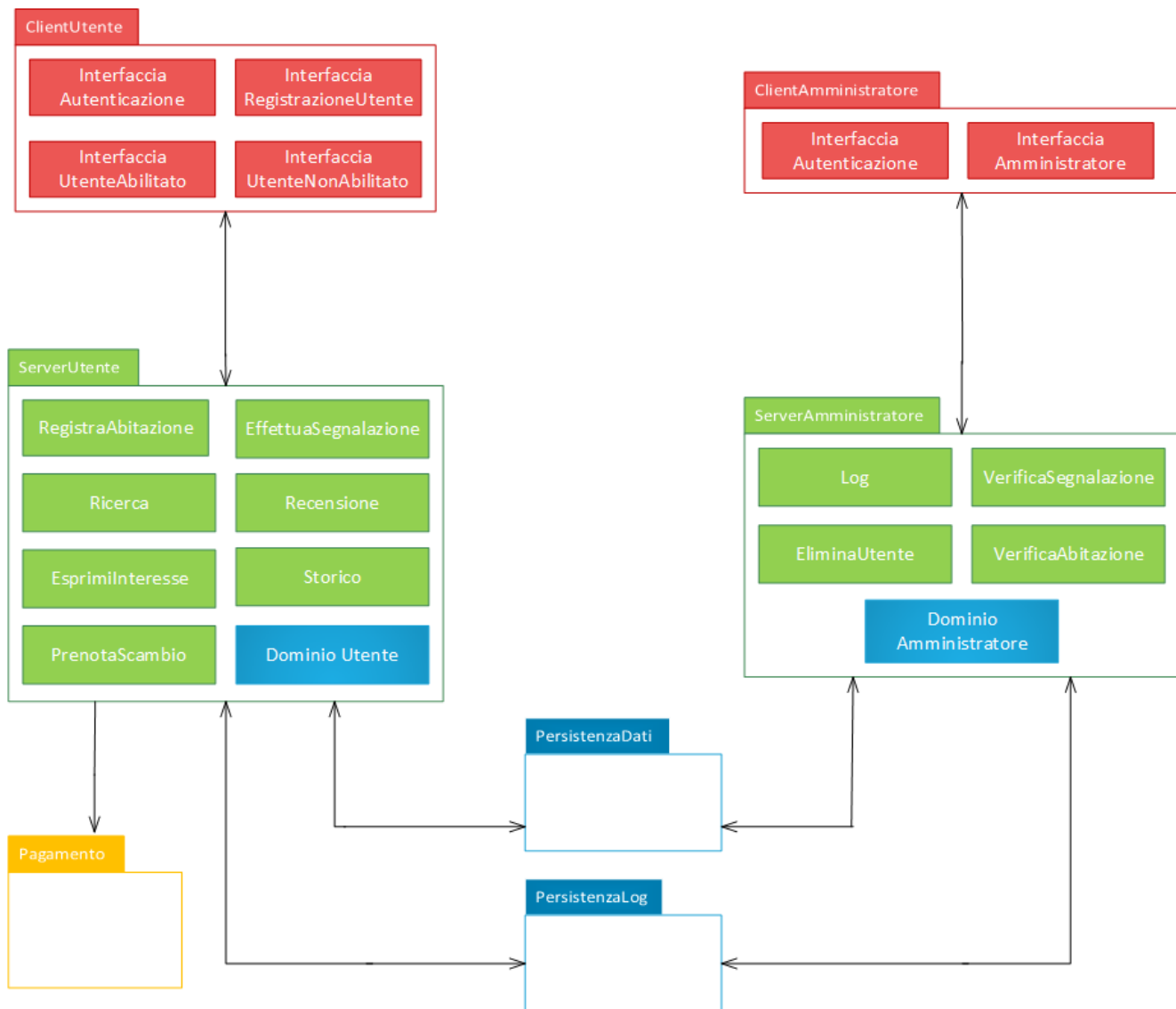


# Progettazione di Dettaglio

## Struttura

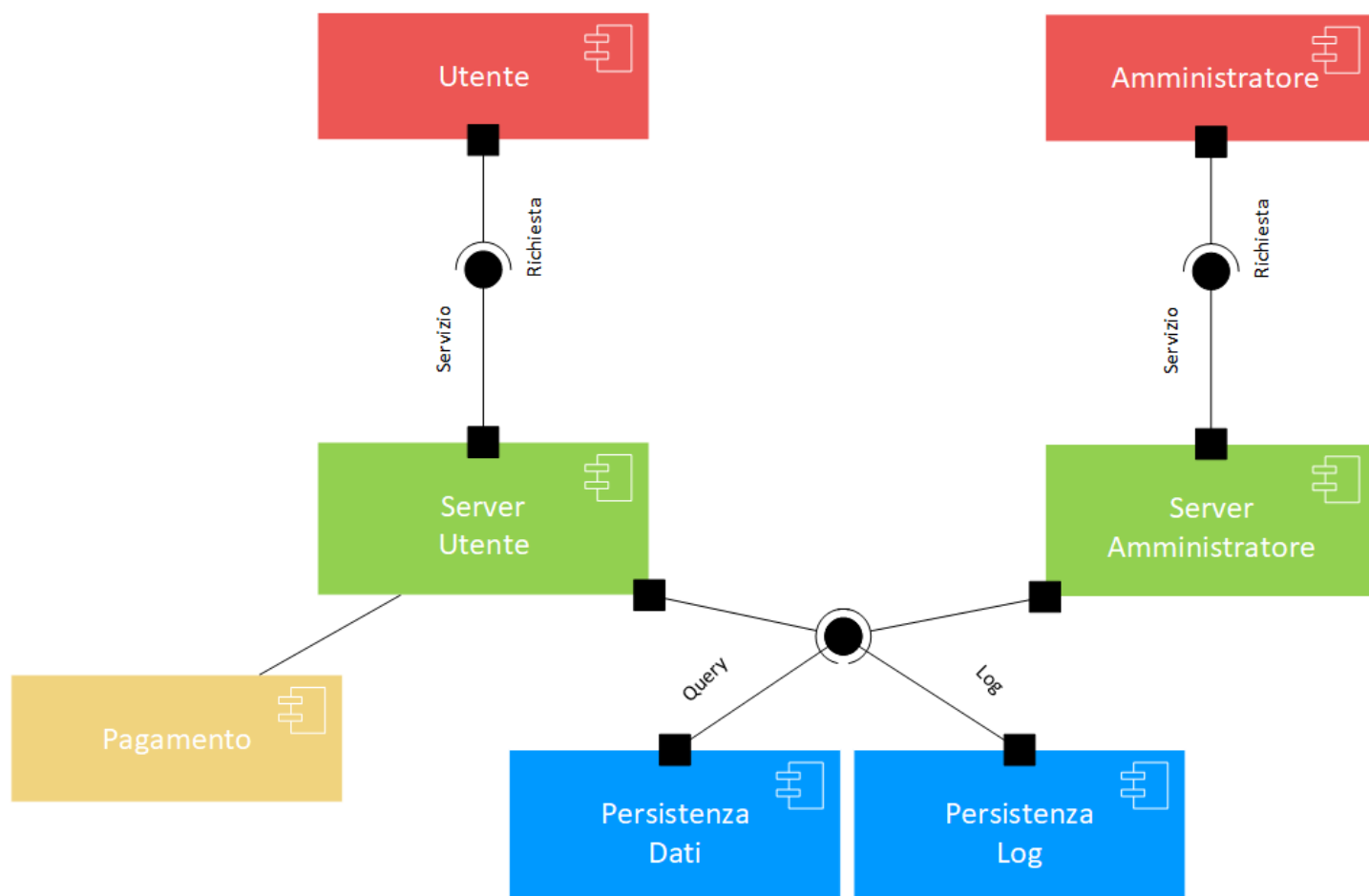
### Diagramma dei Package

Nella figura seguente è riportata l'Architettura del *Sistema* organizzata attraverso un diagramma dei package:



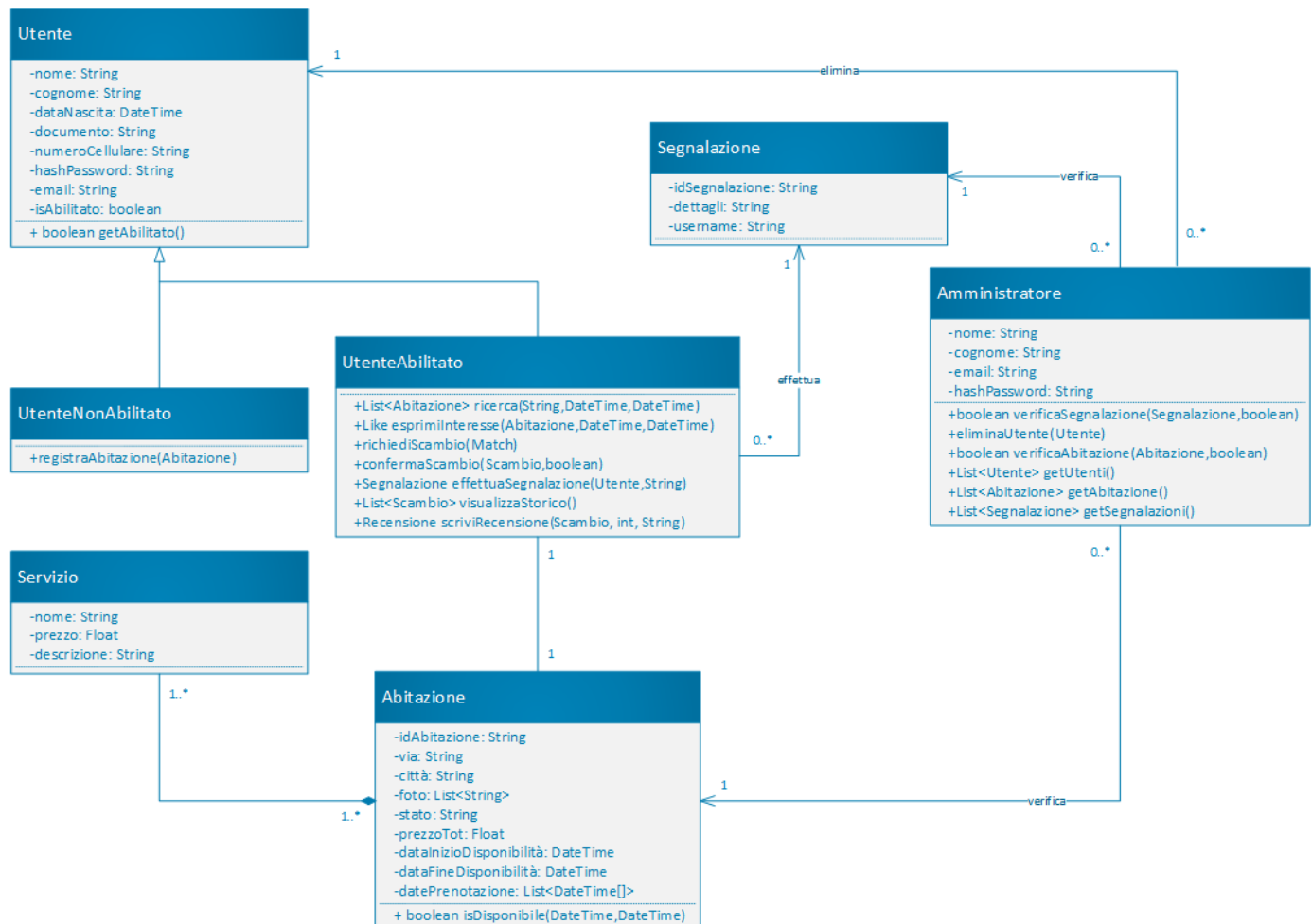
## Diagramma dei Componenti

Nella figura sottostante è riportata l'Architettura del *Sistema* organizzata attraverso un diagramma dei componenti:



## Diagramma di dettaglio: Dominio

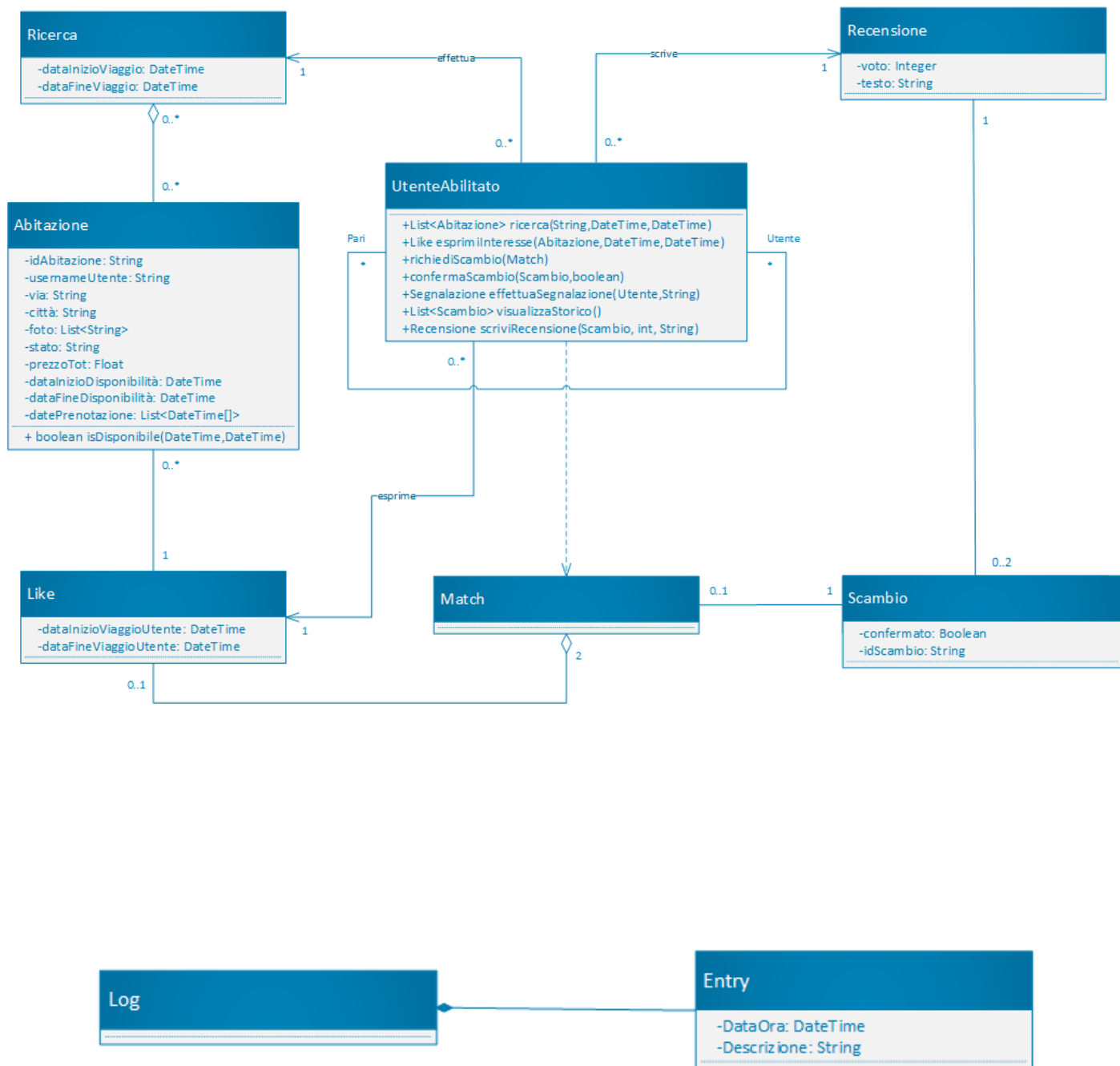
Nei diagrammi seguenti sono state apportate delle modifiche agli attributi e ai metodi. Per ogni figura sono specificate le modifiche più rilevanti.



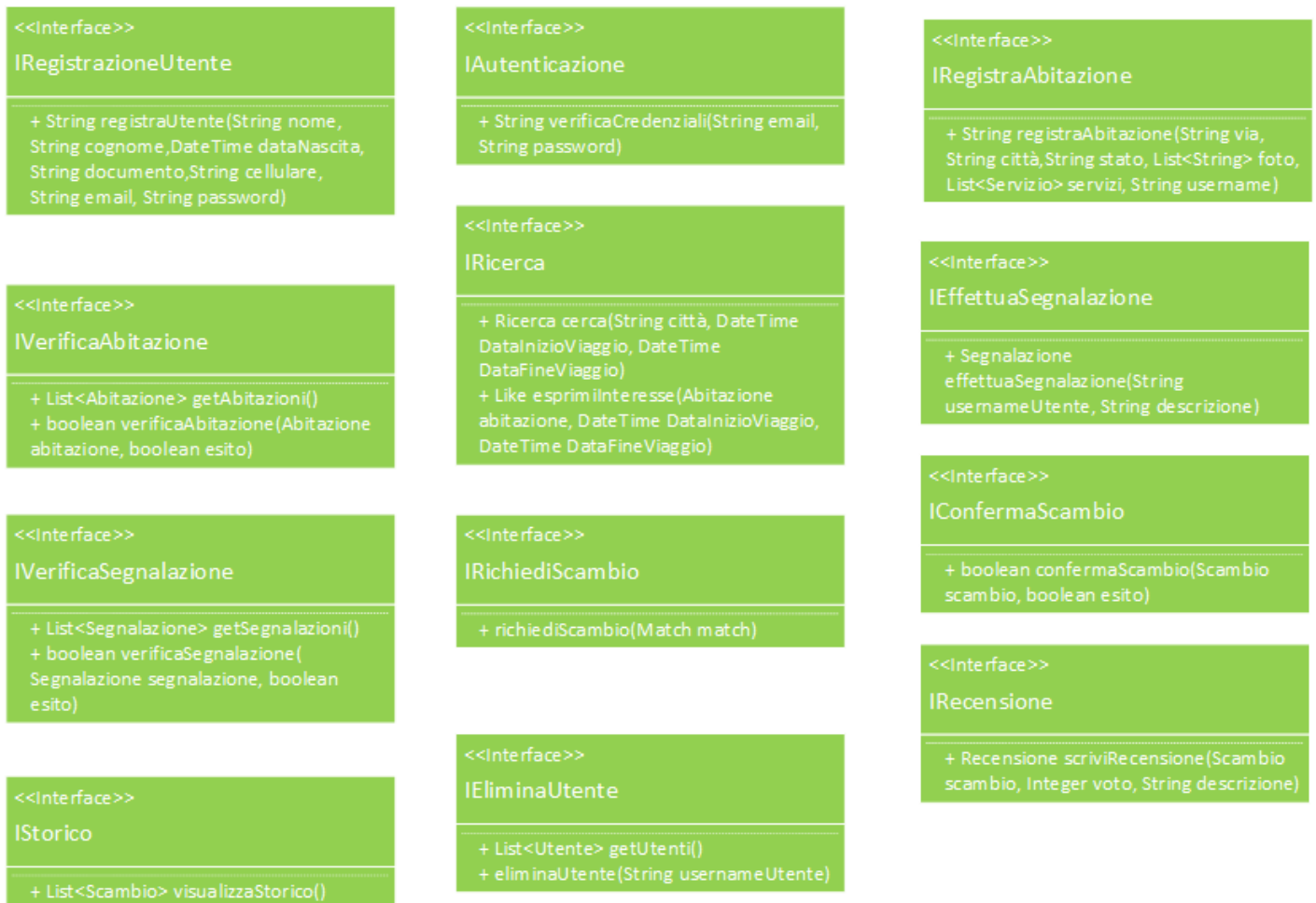
Ora l'*Utente* e l'*Amministratore* presentano l'attributo "*hashPassword*" che rappresenta l'hash delle rispettive *password* ottenuto a seguito della cifratura. All'*Utente* è stato aggiunto l'attributo "*isAbilitato*", che indica se ha registrato o meno un'*Abitazione*.

La *Segnalazione* presenta l'attributo "*username*", che indica a quale *Utente* è rivolta. Tuttavia, non è detto che l'*username* inserito appartenga realmente ad un *Utente* registrato nel *Sistema*, per questo motivo si è deciso di non collegare *Segnalazione* ad *Utente*.



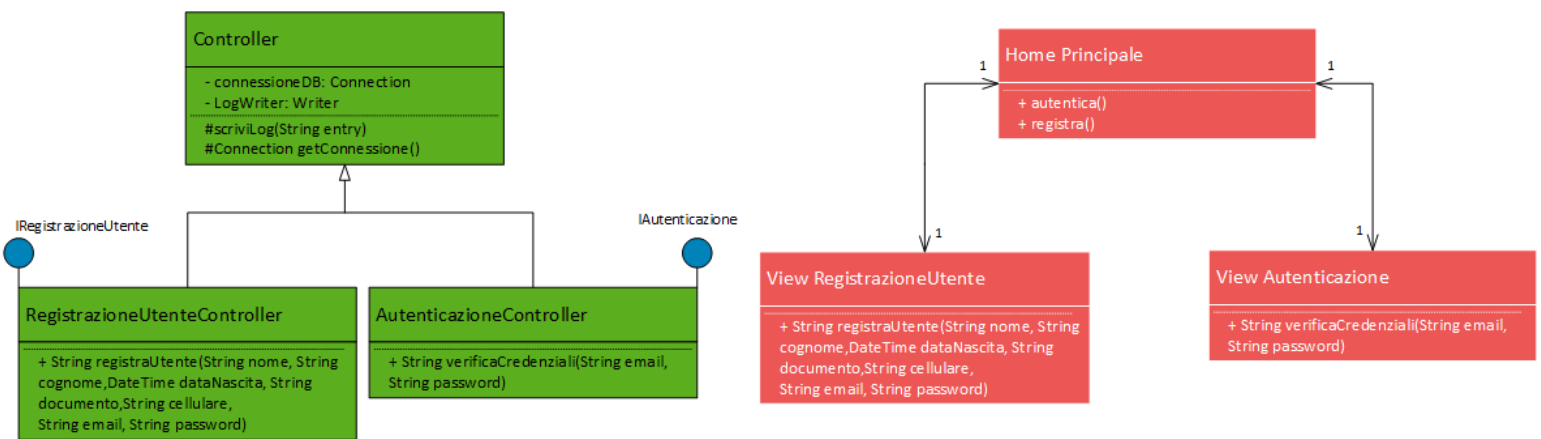


## Diagramma di dettaglio: Interfacce



In pieno rispetto del **Dependency Inversion Principle**, mediante l'utilizzo di *Interfacce* si evita di dipendere da una particolare implementazione.

## Diagramma di dettaglio: Registrazione-Autenticazione



Host2Host

LOGIN

Username  
 Nome  
 Cognome  
 30/05/2021   
 Documento  
 Cellulare  
 Password  
 Conferma Password

Host2Host

REGISTRAZIONE

LOGIN

---

 Username  
 Password



## Diagramma di dettaglio: Gestione Utente non Abilitato



Host2Host

LOGOUT

Registra la tua Abitazione

Stato

Città

Ex. Italy

Ex. Bologna

Indirizzo

Ex. Viale del Risorgimento, 2

Data di inizio disponibilità

Data di fine disponibilità

gg/mm/aaaa

gg/mm/aaaa

Foto

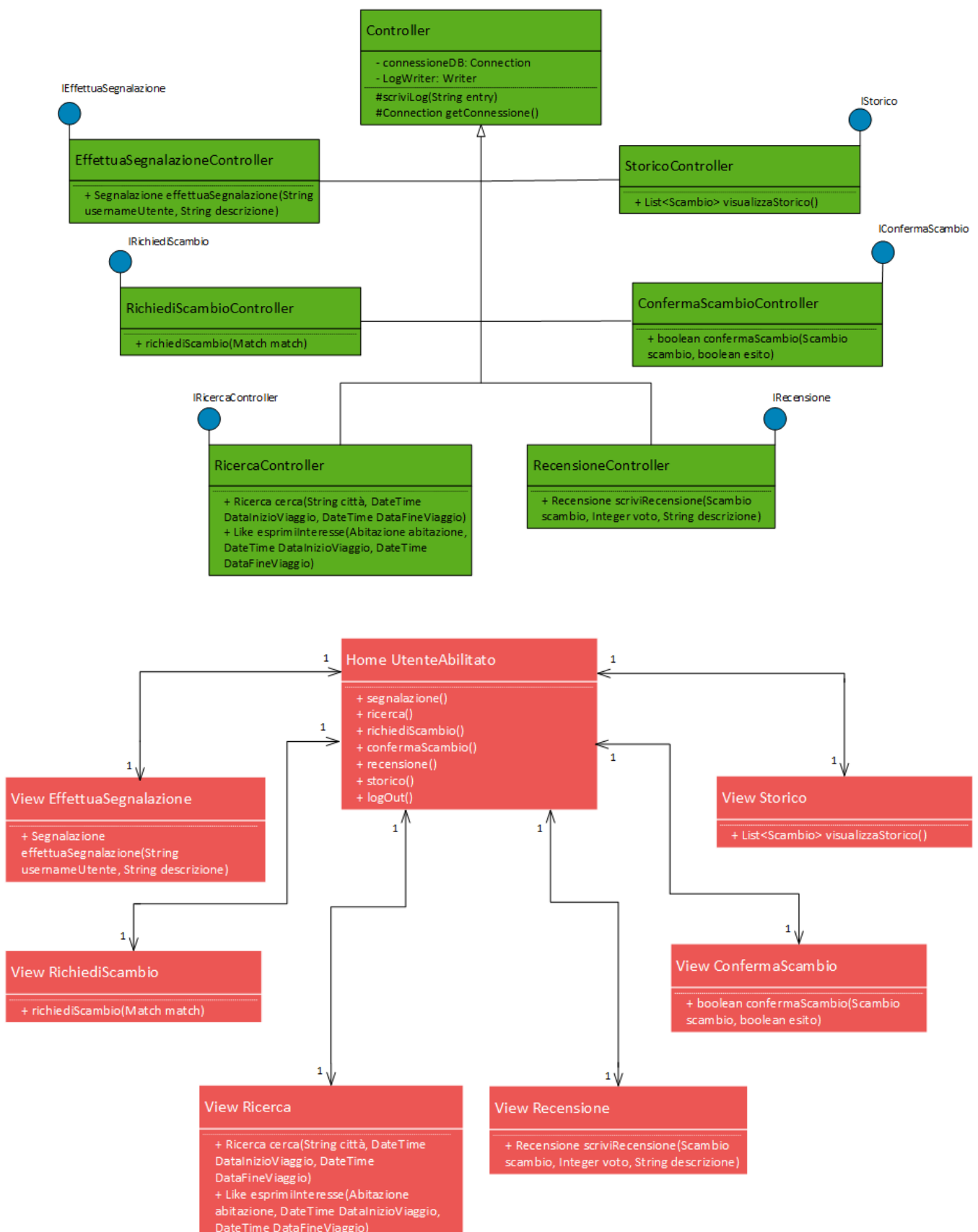
Servizi

UPLOAD FOTO

Aggiungi

Registra Abitazione




## Diagramma di dettaglio: Gestione Utente Abilitato





View  
Ricerca

**Host2Host**STORICOSCAMBIOLOGOUT

### CIAO CIRO

 City 30/05/2021 30/05/2021 Cerca

**Bologna, Italy** ★★★★★  
**Via San Mamolo, 14 (BO)**  
 Alessio Di Domenico, 22 anni  
*"Ciao, sono Alessio e sono una persona solare"*  
[LIKE](#)



**Servizi**


<b>Gita al parco</b>	Passeggiata nel verde dei colli bolognesi	<b>5€</b>
<b>Giro in vespa</b>	Giro in vespa sui colli bolognesi a mangiare tortellini	<b>10€</b>
<b>Visita al Museo</b>	Visita al museo di arte contemporanea al MAMbo	<b>25€</b>


View  
Richiedi  
Scambio

**Host2Host**STORICOSCAMBIOLOGOUT

### CIAO JACOPO

#### Lista dei Match

**Bologna, Italy** ★★★★★  
**Via San Mamolo, 14 (BO)**  
 Alessio Di Domenico, 22 anni  
*"Ciao, sono Alessio e sono una persona solare"*  
[Richiedi Scambio](#)



**Servizi**


<b>Gita al parco</b>	Passeggiata nel verde dei colli bolognesi	<b>5€</b>
<b>Giro in vespa</b>	Giro in vespa sui colli bolognesi a mangiare tortellini	<b>10€</b>
<b>Visita al Museo</b>	Visita al museo di arte contemporanea al MAMbo	<b>25€</b>


View  
Conferma  
Scambio

**Host2Host**STORICOSCAMBIOLOGOUT

### CIAO ALESSIO

#### Lista delle Richieste

**Napoli, Italy** ★★★★★  
**Via Calabritto, 10 (NA)**  
 Jacopo Di Battista, 22 anni  
*"Ciao sono Jacopo, il divertimento ci aspetta!"*  
[Conferma Scambio](#)

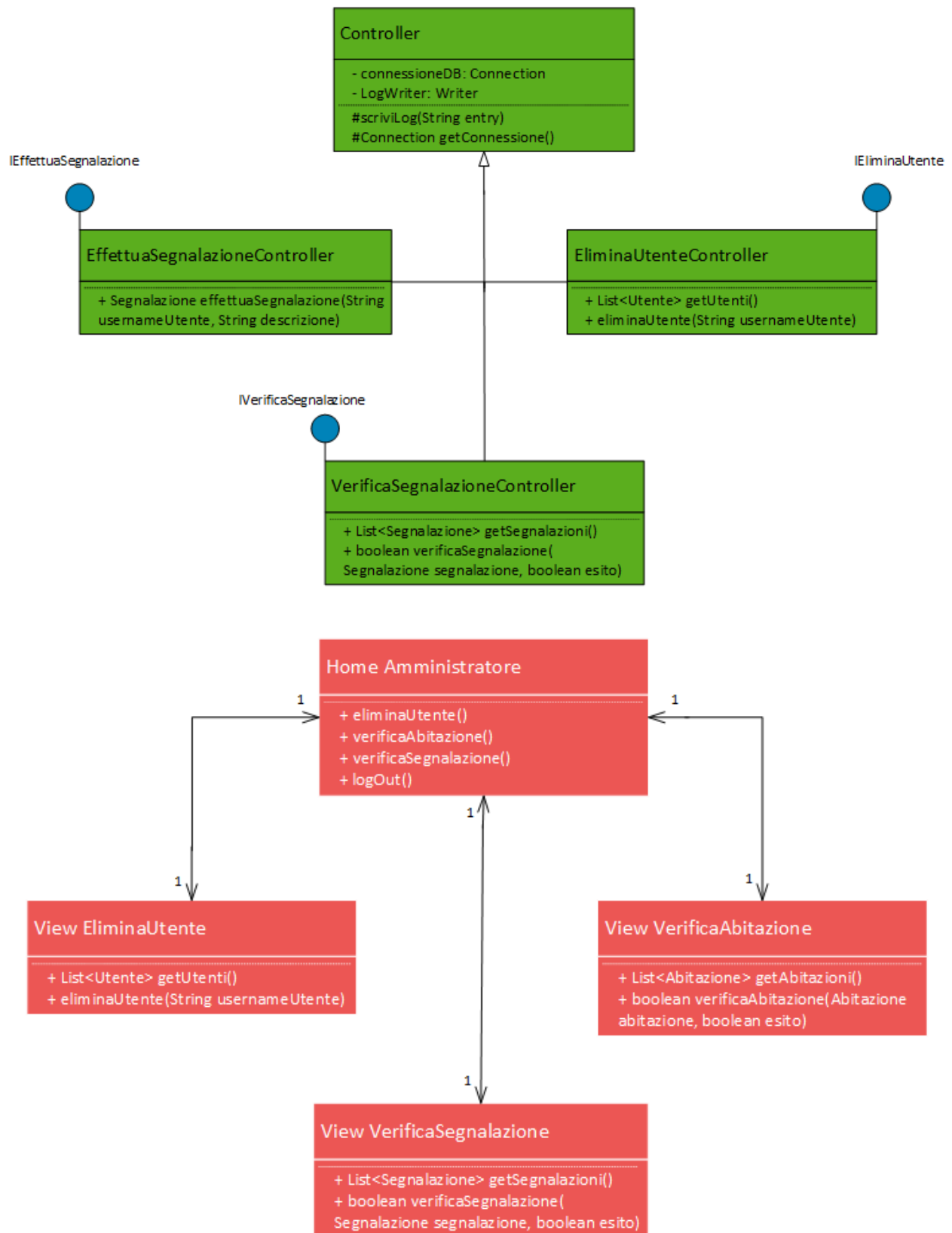


**Servizi**

<b>Gita in barca</b>	Giornata in barca sul golfo di Napoli	<b>50€</b>
<b>Grigliata di pesce</b>	Pranzo a base di pesce sul lungomare	<b>7€</b>
<b>Maschio Angiolino</b>	Visita al castello	<b>5€</b>



## Diagramma di dettaglio: Gestione Amministratore



View  
Elimina  
Utente

**Host2Host**SEGNALAZIONIABITAZIONILOGOUT

### CIAO ADMIN

Utenti	Elimina
james.montana@unibo.it	
marco.belinelli@virtus.it	
riccardo.ricci@unibo.it	

View  
Verifica  
Segnalazione

**Host2Host**ABITAZIONIUTENTILOGOUT

### CIAO ADMIN

Segnalazioni	Dettagli
ID:123 USER:marco.belinelli@virtus.it	
ID:345 USER:james.montana@unibo.it	

LOCATION  
Viale del Risorgimento,2

AROUND THE WEB

CONTACTS  
yuri.noviello@studio.unibo.it

View  
Verifica  
Abitazione

**Host2Host**SEGNALAZIONIUTENTILOGOUT

### CIAO ADMIN

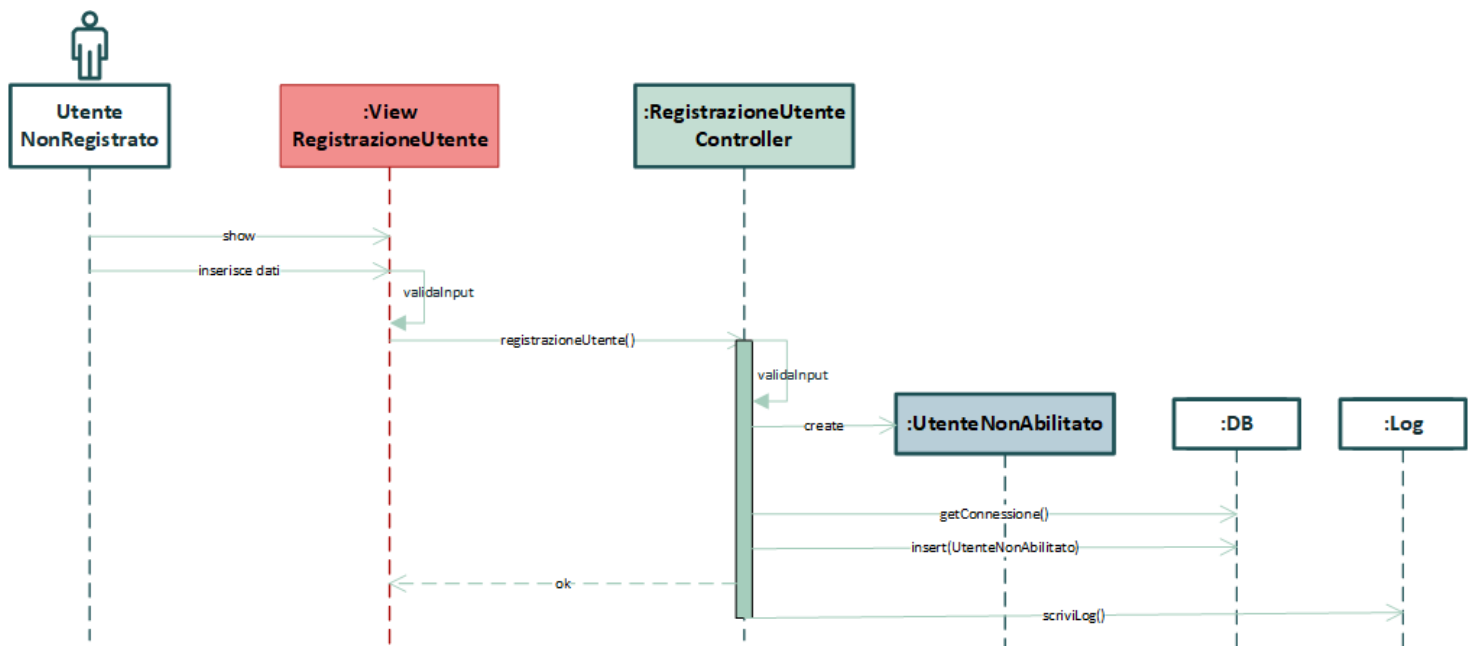
Abitazioni	Dettagli
Viale Risorgimento,2	
Via Vittorio Veneto,11	
Via San Mamolo,2	



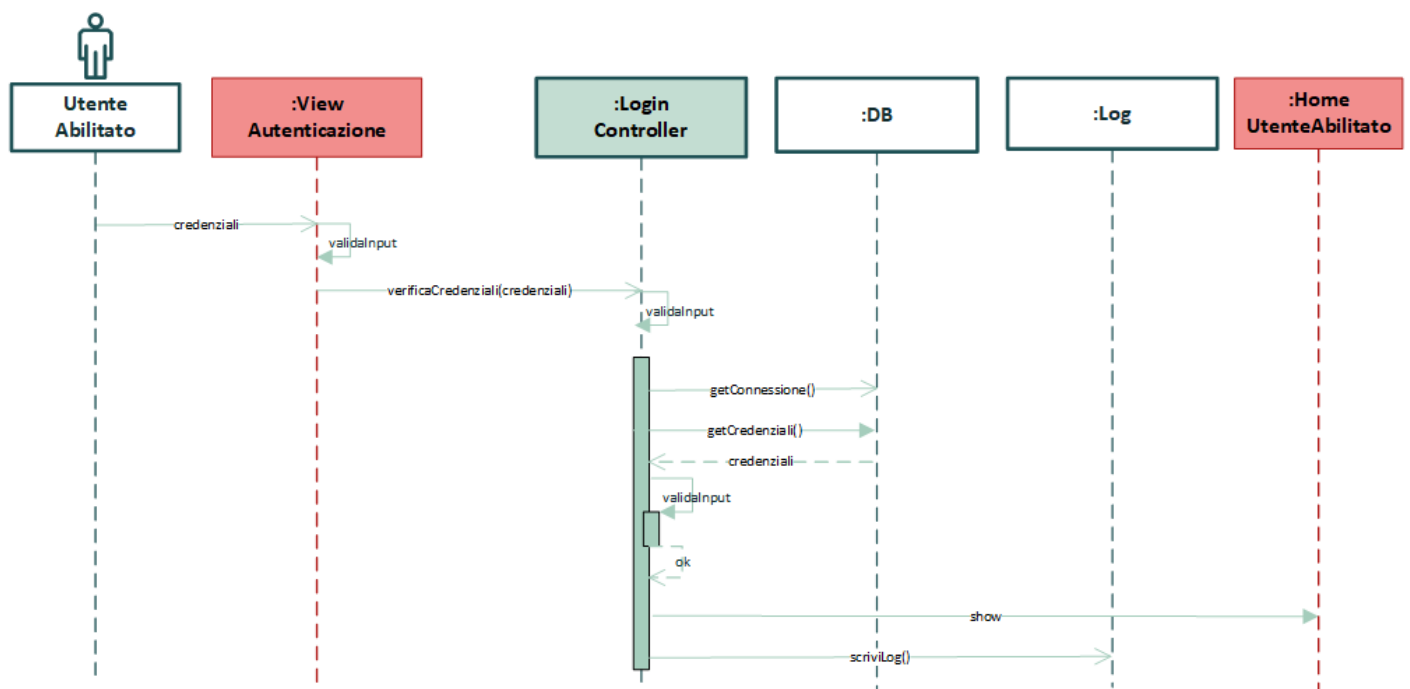


## Interazione

### Diagramma di sequenza: Autenticazione

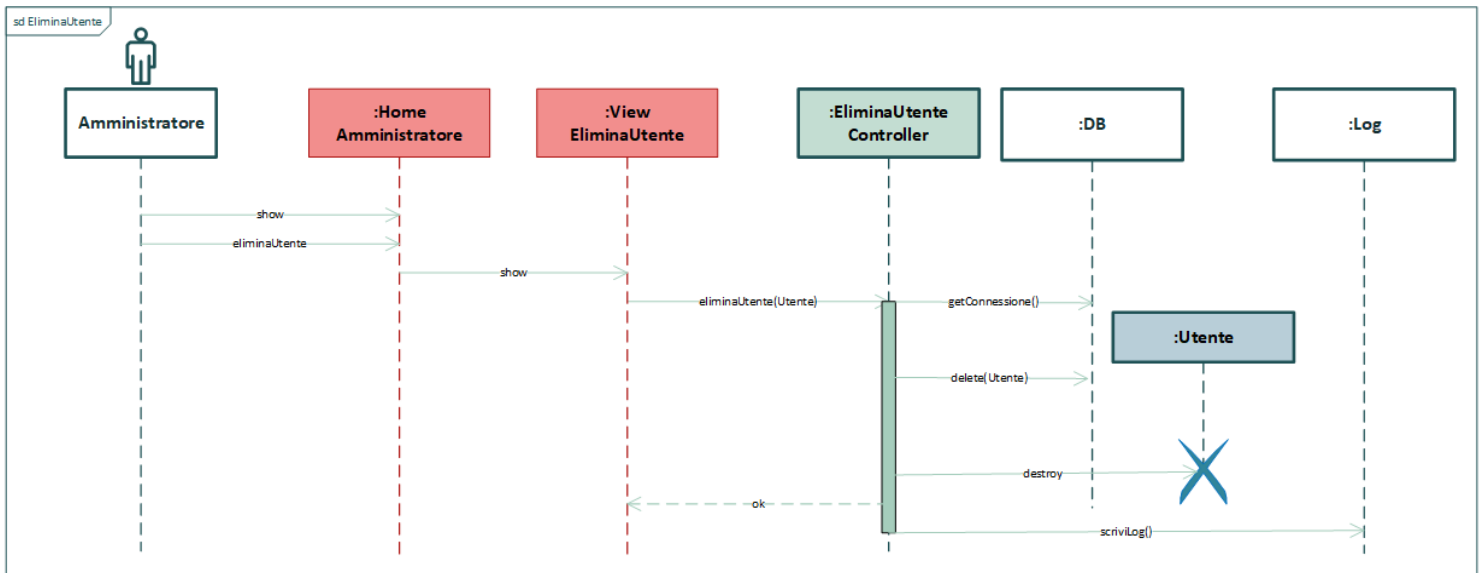


### Diagramma di sequenza: Autenticazione

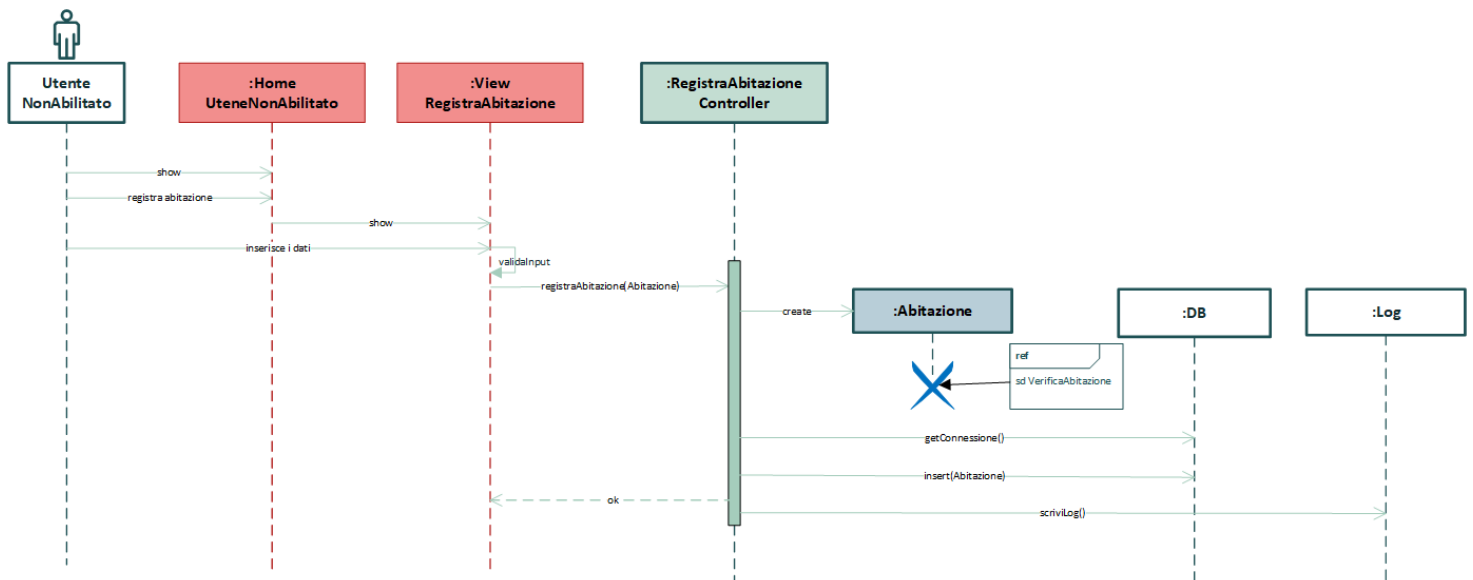


Sul **Database** non è memorizzata la *password* in chiaro ma solo il suo *hash*.

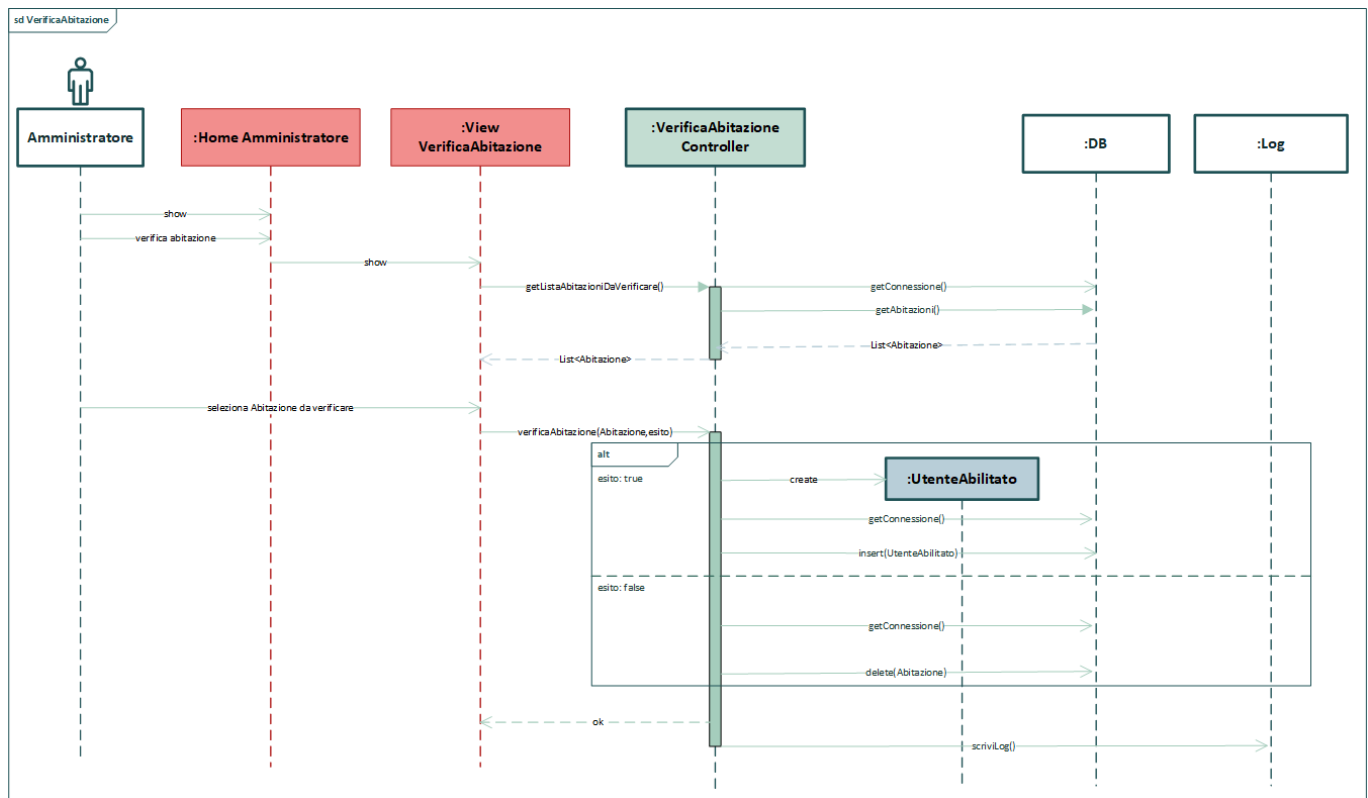
## Diagramma di sequenza: EliminaUtente



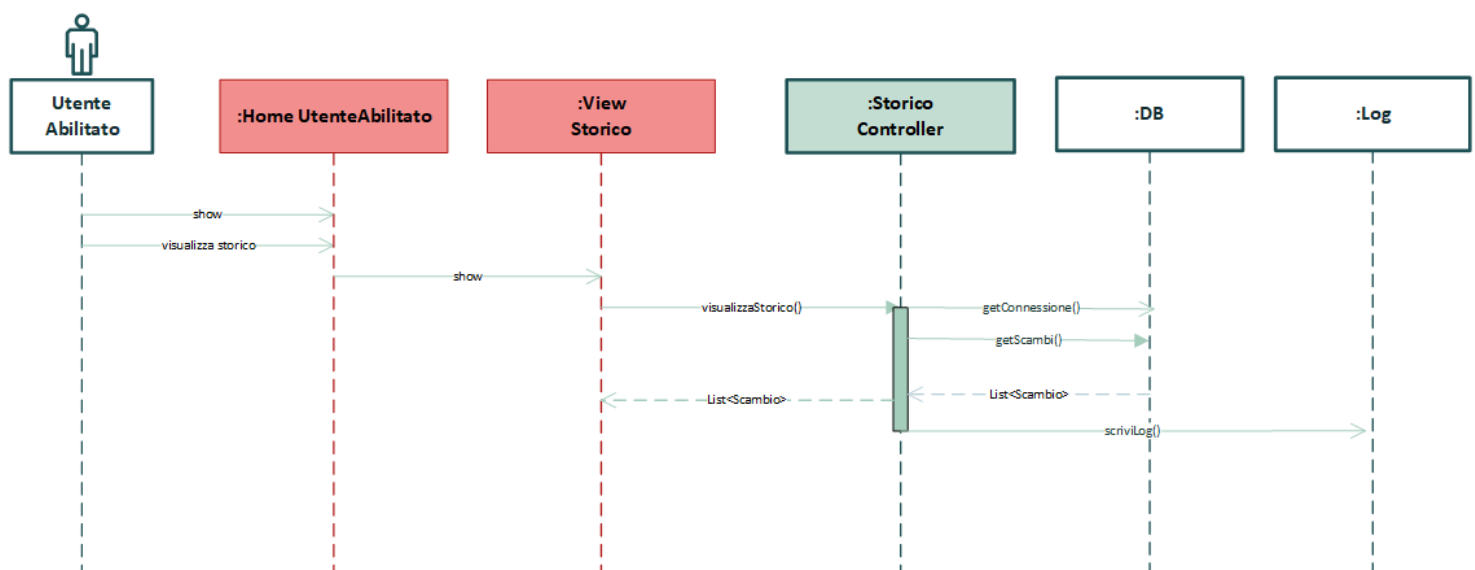
## Diagramma di sequenza: RegistraAbitazione



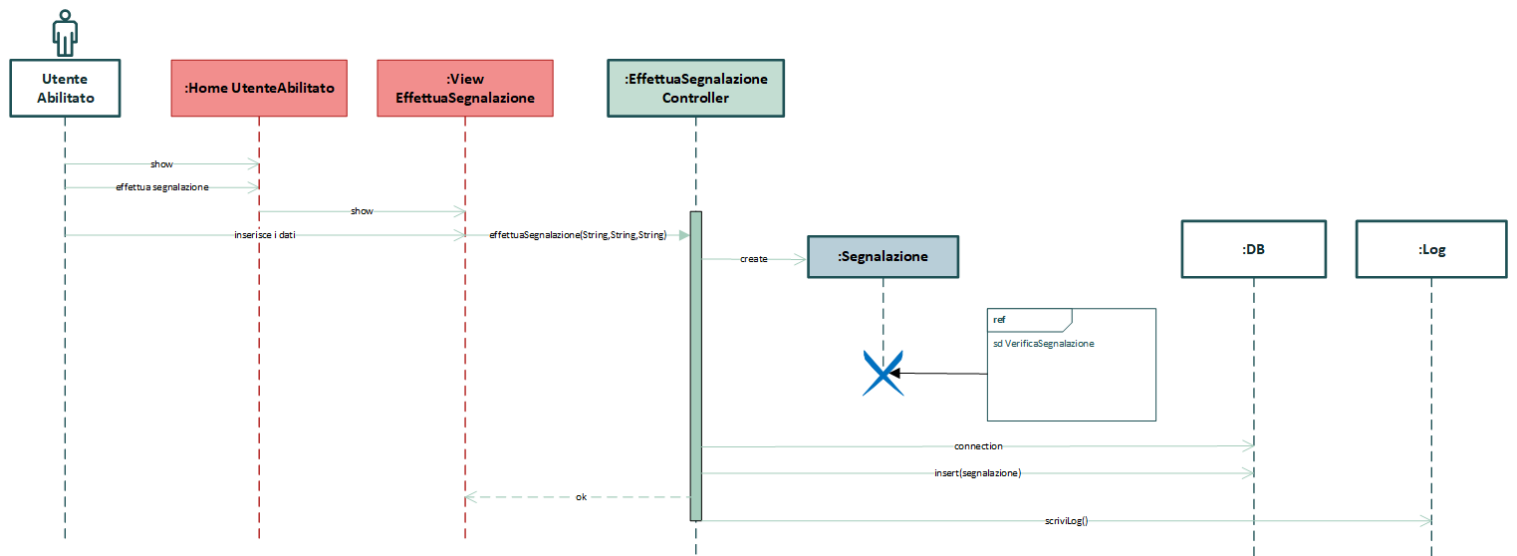
## Diagramma di sequenza: VerificaAbitazione



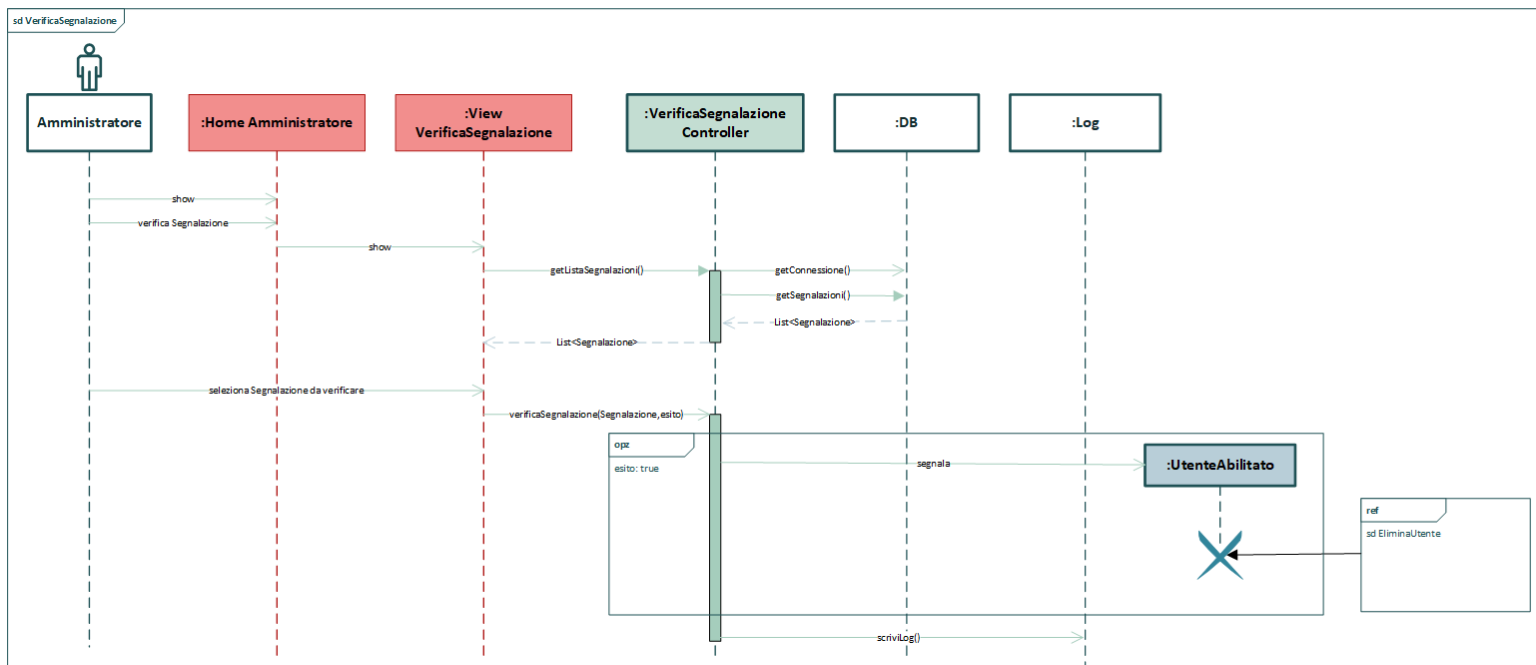
## Diagramma di sequenza: Storico



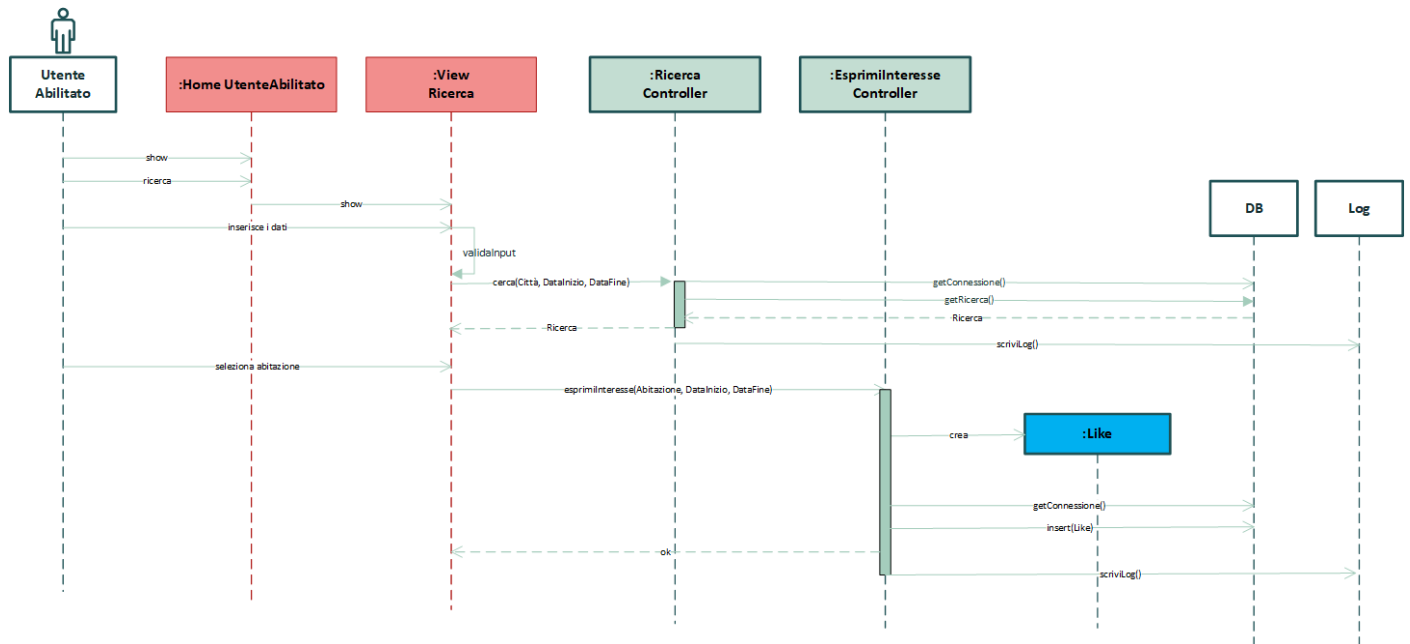
## Diagramma di sequenza: EffettuaSegnalazione



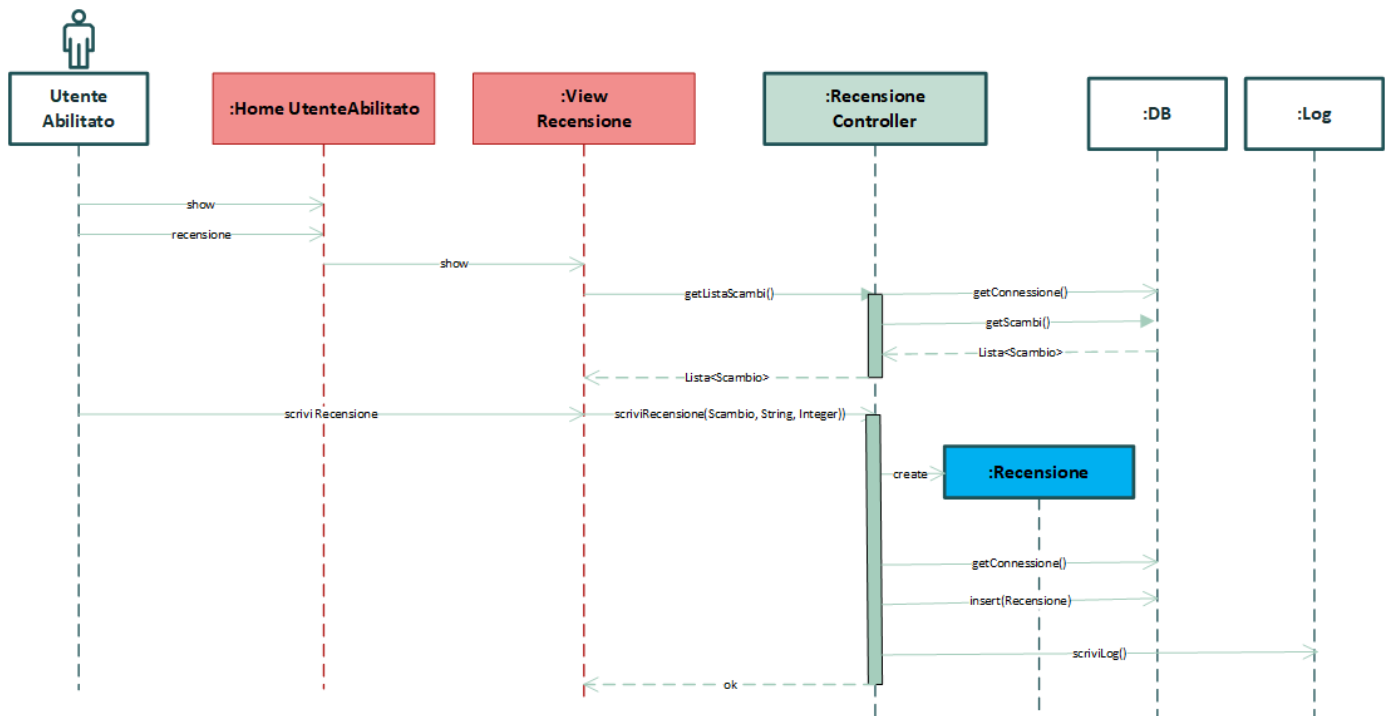
## Diagramma di sequenza: VerificaSegnalazione



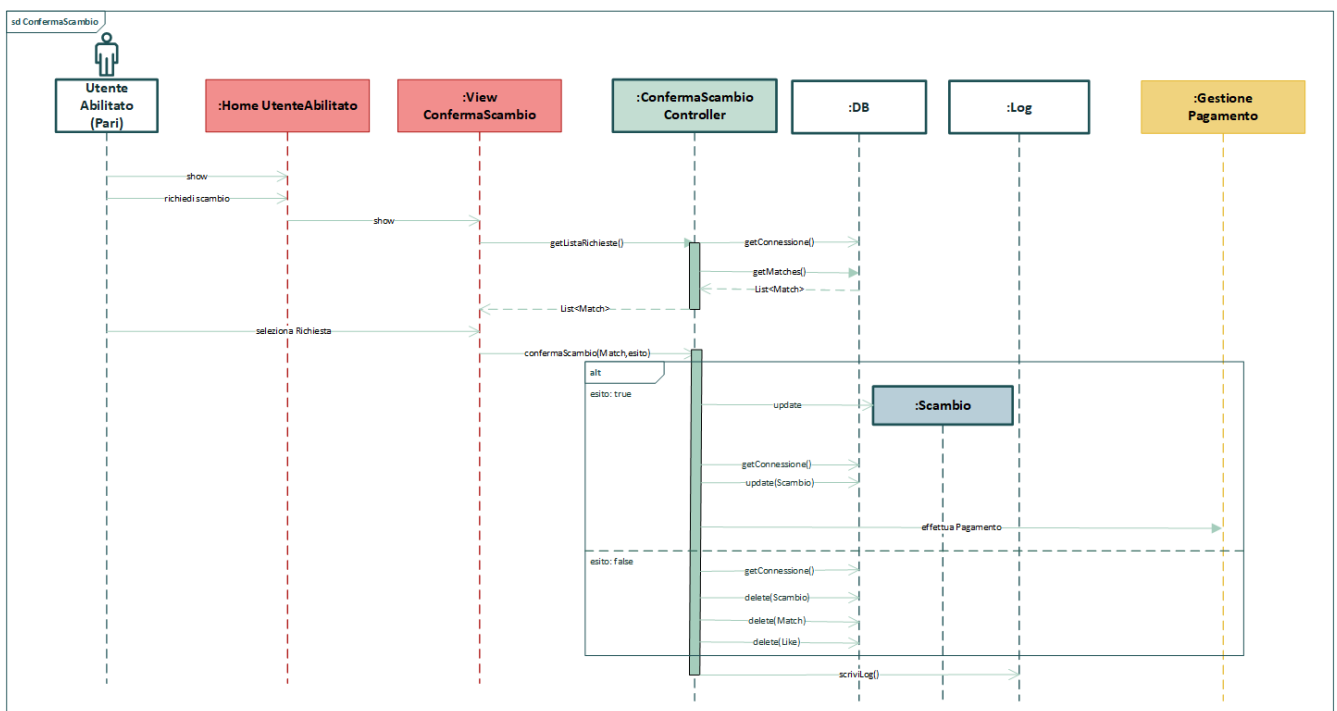
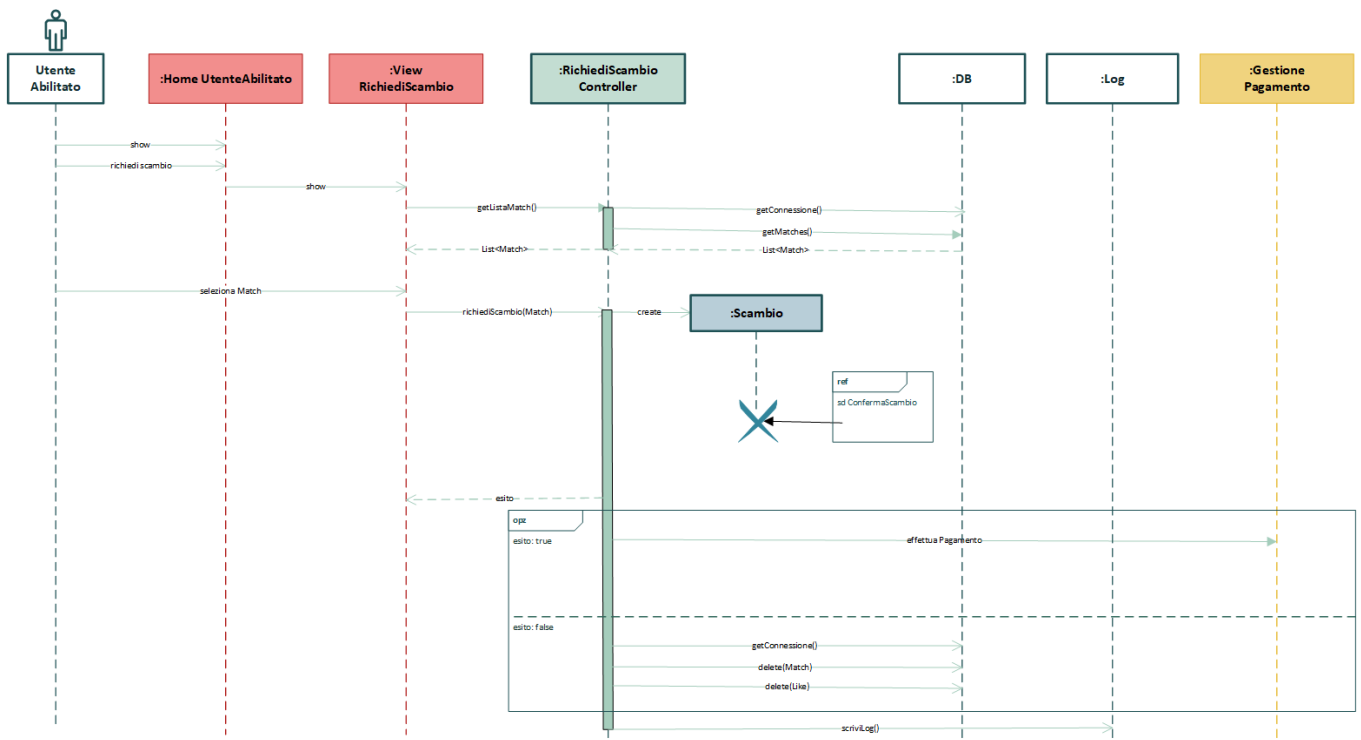
## Diagramma di sequenza: Ricerca



## Diagramma di sequenza: Recensione



## Diagramma di sequenza: PrenotaScambio

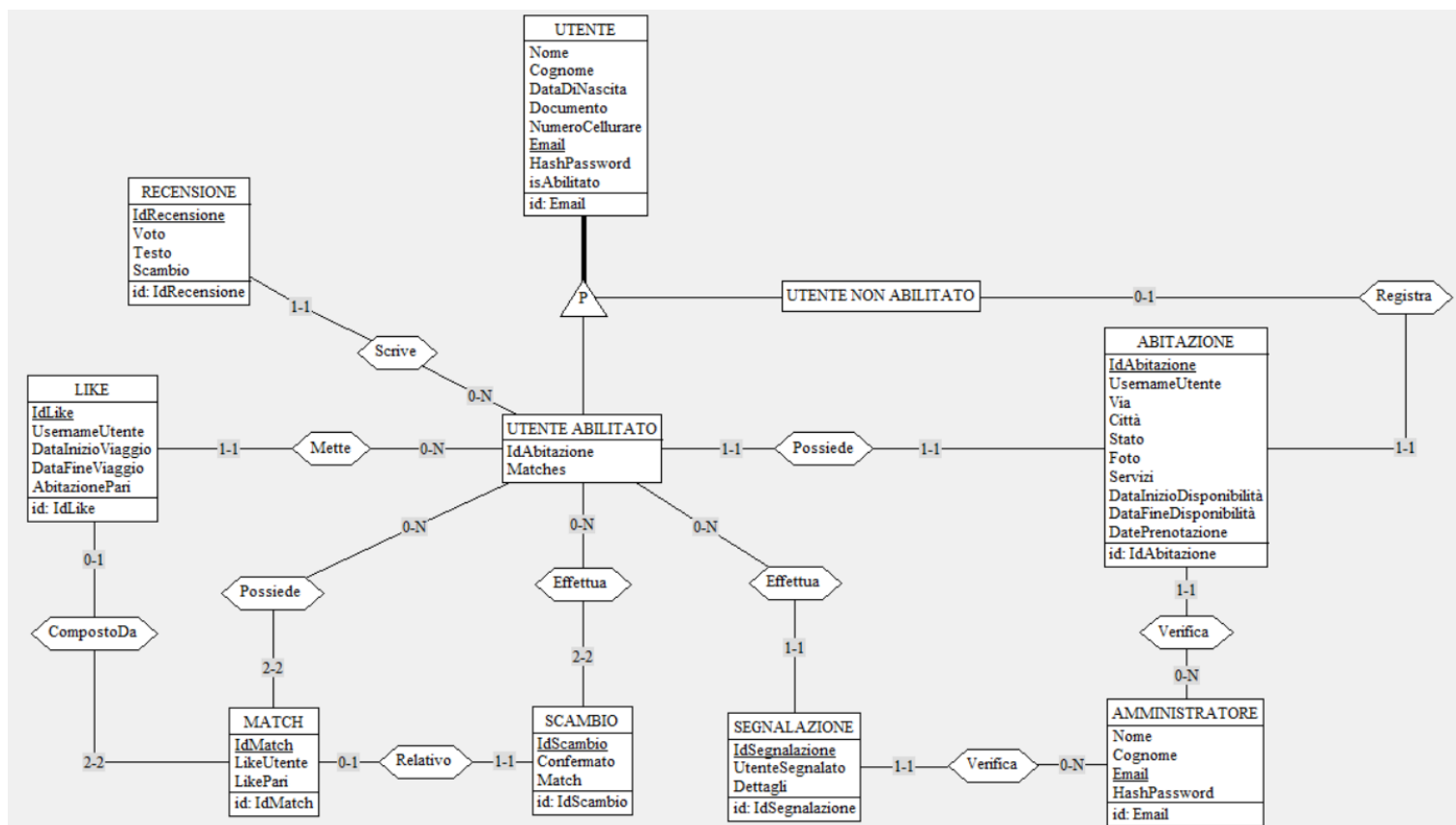


In caso di *Scambio* rifiutato, dal **Database** vengono eliminati i *Match* degli *Utenti* e il relativo *Scambio*

## Comportamento

Dopo un'attenta riflessione, abbiamo ritenuto che nessuna entità richieda un diagramma di stato.

## Progettazione della Persistenza



# Progettazione del Collaudo

Di seguito è riportato il test di funzionamento della *Ricerca* e della funzione *EsprimiInteresse* da parte di un *Utente*:

```
public class TestCollaudo {

    private RicercaController controllerRicerca;
    private EsprimiInteresseController controllerInteresse;
    private Abitazione a1,a2;
    private UtenteAbilitato user;

    @Before
    public void setUp(){
        controllerRicerca = new RicercaController();
        controllerInteresse = new EsprimiInteresseController();

        user = new UtenteAbilitato();
        LocalDate data = LocalDate.of(1986, 3, 25);

        user.setNome("Marco");
        user.setCognome("Belinelli");
        user.setEmail("marco.belinelli@virtus.it");
        user.setDataDiNascita(data);
        user.setDocumento("CA00000AA");
        user.setNumeroCellulare("+39321123456");
        user.setPassword("password");

        a1 = new Abitazione();
        a1.setVia("Viale del Risorgimento 2");
        a1.setCitta("Bologna");
        a1.setStato("Italia");
        a1.setUsername("james.montana@unibo.it");
        a1.setDataInizioDisponibilita(LocalDate.of(2021, 6, 1));
        a1.setDataFineDisponibilita(LocalDate.of(2021, 7, 15));
        controllerRicerca.addAbitazione(a1);

        a2 = new Abitazione();
        a2.setVia("Via Veneto 4");
        a2.setCitta("Bologna");
        a2.setStato("Italia");
        a2.setUsername("riccardo.ricci@unibo.it");
        a2.setDataInizioDisponibilita(LocalDate.of(2021, 7, 1));
        a2.setDataFineDisponibilita(LocalDate.of(2021, 7, 31));
        controllerRicerca.addAbitazione(a2);
    }
}
```





```

@Test
public void test() {
    LocalDate dataInizio = LocalDate.of(2021, 7, 1);
    LocalDate dataFine = LocalDate.of(2021, 7, 15);

    Ricerca ricerca = controllerRicerca.cerca("Bologna",
    dataInizio, dataFine);

    Assert.assertEquals(ricerca.getAnnunci().size(), 2);
    Assert.assertEquals(ricerca.getAnnunci().get(0), a1);
    Assert.assertEquals(ricerca.getAnnunci().get(1), a2);

    Like like = controllerInteresse.esprimiInteresse(
    user.getEmail(), ricerca.getAnnunci().get(0),
    dataInizio, dataFine);

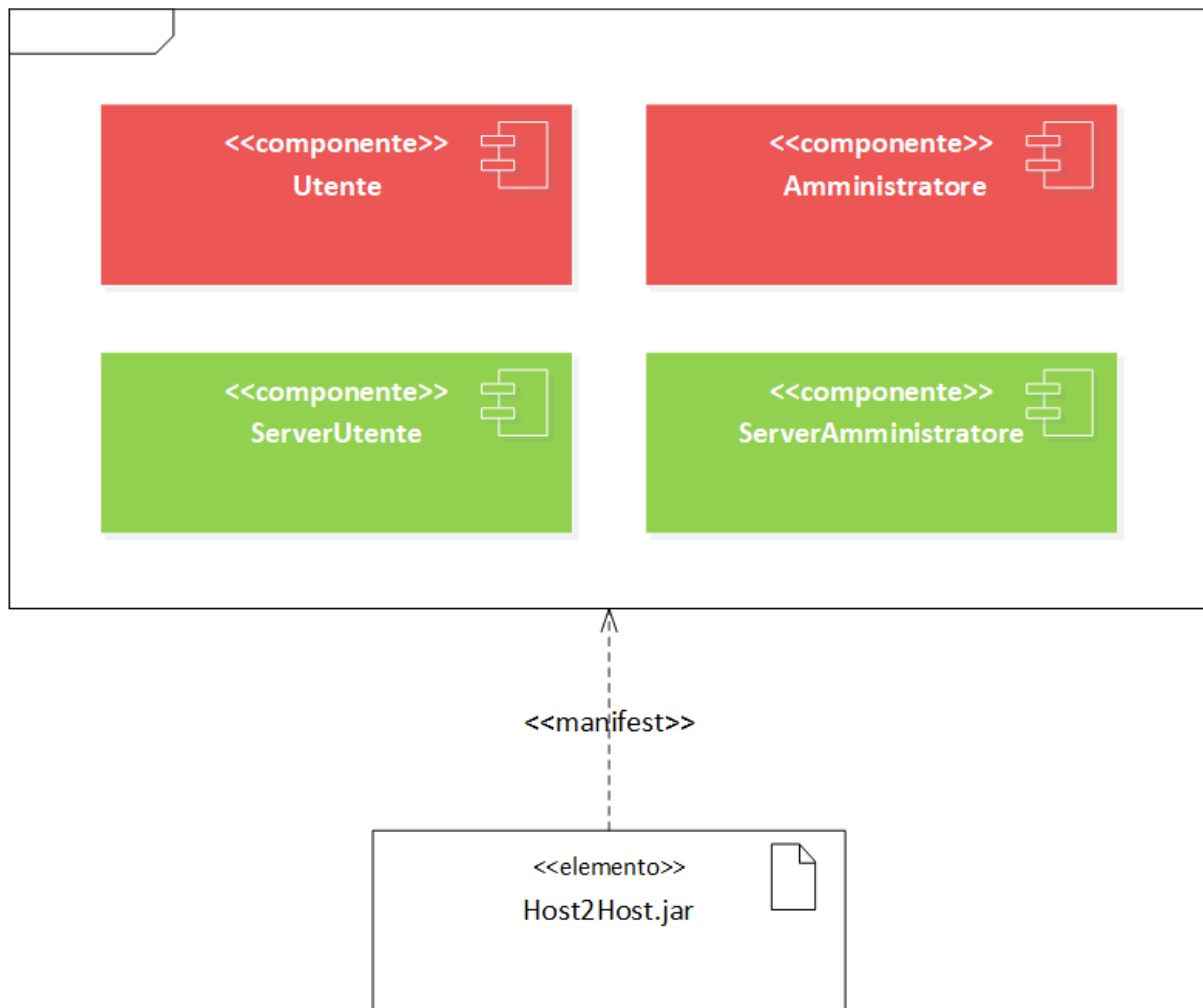
    Assert.assertEquals(like.getUsername(),
    "marco.belinelli@virtus.it");
    Assert.assertEquals(like.getAbitazione(), a1);
    Assert.assertEquals(like.getDataInizio(), dataInizio);
    Assert.assertEquals(like.getDataFine(), dataFine);
}
}

```



# Progettazione del Deployment

## Artefatti



## Deployment Type-Level

