

# **Flashy**

**il progetto delle FlashCard**

Alberto Catellani 0001020328  
Manuel Barbieri 0001021183  
Enrico Minguzzi 0001029905

# Sommario

<b>Sommario</b>	<b>2</b>
<b>ABSTRACT</b>	<b>4</b>
<b>Analisi dei Requisiti</b>	<b>4</b>
Requisiti del sistema	4
Tabella dei Requisiti	5
Vocabolario	6
Casi d'uso - Grafico	8
Casi d'uso - Scenari	8
<b>Analisi del Rischio</b>	<b>13</b>
Tabelle di Valutazione dei Beni	13
Tabelle delle Minacce / Controlli	13
Analisi della tecnologia dal punto di vista della sicurezza	14
Security: Use & Misuse Cases - Grafico	15
Security: Use & Misuse Cases - Scenari	15
Requisiti aggiuntivi emersi	17
<b>Analisi del Problema</b>	<b>17</b>
Analisi delle Funzionalità	17
Analisi dei Vincoli	21
Analisi delle Interazioni	21
Tabella Maschere	22
Tabella Sistemi Esterni	23
Analisi dei Ruoli e delle Responsabilità	23
Tabella dei Ruoli	23
Scomposizione del Problema	24
Tabella Scomposizione delle Funzionalità	24
Creazione del Modello di Dominio	26
Architettura logica: Struttura	28
Diagramma dei Package	28
Diagrammi delle Classi	28
Architettura logica: Interazione	31
Piano di Lavoro	33
Prototipo	34
Sviluppi futuri	34
Piano del Collaudo	35
<b>Progettazione</b>	<b>36</b>
Progettazione Architetturale	36
Requisiti non funzionali	36
Scelta dell'architettura	37
Architettura del Sistema: rappresentazione in package	39
Architettura dei componenti	40
Scelte tecnologiche	40

Progettazione di Dettaglio	41
Struttura: Dominio	41
Struttura: Interfacce	43
Struttura: Controller	45
View ed Interfacce	49
Interazione	58
Progettazione della Persistenza	63
Formato dei File di Log	63
Progettazione di Collaudo	64
<b>Deployment</b>	<b>65</b>
Artefatti	65
Deployment Type-Level	65

## ABSTRACT

*Si vuole realizzare una piattaforma di scopo educativo, che possa aiutare nel processo di memorizzazione ed autovalutazione di studenti e non solo, basandosi sulla tecnica delle Flashcards. Dopo aver inserito le proprie credenziali ed essersi autenticati, ogni utente ha la possibilità di interagire con le proprie collezioni di Flashcards, avviando una prova o creando una nuova collezione.*

*Nella prima casistica, si predisponde sia una modalità "esercitazione", in cui è possibile svelare le risposte delle varie carte, sia una modalità "simulazione" in cui non è possibile visionare le risposte. In entrambe è l'utente stesso a valutare la propria prestazione. Al termine della prova, verranno mostrati alcune statistiche come la percentuale di risposte corrette o il capitolo in cui si sono commessi il numero maggiore di errori.*

*La creazione di una collezione si basa sull'inserimento delle domande, delle risposte e l'eventuale aggiunta di un'etichetta per specificare l'argomento a cui la domanda si rivolge.*

# Analisi dei Requisiti

## Requisiti del sistema

- Gli utenti devono autenticarsi o registrarsi nel caso non possiedano un account.
- Gli utenti all'interno dell'applicazione scelgono di creare una nuovo mazzo di carte, popolarlo, effettuarvi un quiz o modificarlo. I mazzi prendono il nome di Collezioni.
- Per creare un nuovo mazzo, l'utente deve semplicemente inserire il nome di questo.
- Nel popolare una Collezione, l'utente, per ogni carta, inserisce la domanda, la risposta e può specificare un opzionale tag di "argomento".
- Ogni carta deve possedere una risposta. Non è possibile creare una carta con la sola domanda. Ergo, non possono esistere Collezioni con carte "incomplete".
- Non è possibile creare carte al di fuori di una Collezione. Le carte appartengono sempre e soltanto a una Collezione.

- Interagendo con una delle Collezioni, l'utente può modificare una o più carte della Collezione (cambiarne la domanda, la risposta o l'eventuale argomento). Inoltre, l'Utente può aggiungere e rimuovere carte.
- Interagendo con una delle Collezioni, sarà possibile avviare la modalità Esercitazione o la modalità Simulazione, usando le sole carte presenti in quella Collezione.
- Prima dell'avvio di un quiz (sia questo un'Esercitazione o una Simulazione), l'utente può specificare la lunghezza di questo. Tale scelta determina quante carte (scelte a caso) dalla Collezione saranno proposte nel quiz che sta per affrontare.
- Durante un quiz, l'utente è libero di visionare carte precedentemente proposte. Similmente, durante una Simulazione è possibile cambiare le valutazione che questi ha precedentemente espresso, fintanto che l'utente non termina la Simulazione.
- I dati relativi ad una Simulazione o una Esercitazione non sono salvati. Ergo, se una Simulazione o una Esercitazione vengono “impropriamente interrotte” non sarà possibile riprendere dal punto in cui l'utente era arrivato.
- È possibile, per le Collezioni in cui alcune carte sono appartenenti ad un argomento, specificare di voler sostenere un quiz usando le sole carte appartenenti a tale argomento.
- Durante una **Simulazione**, per ogni flashcard, l'utente visualizza la sola domanda sul “fronte” della carta, oltre che una serie di indicatori di autovalutazione (“positivo”, “negativo”). Non è possibile, in questa modalità, visualizzare le risposte delle carte. L'utente esprime un'autovalutazione per ognuna delle carte proposte. Solo quando tutte le carte avranno ricevuto una valutazione sarà possibile concludere la Simulazione.  
Una volta terminata la Simulazione, vengono presentate alcune statistiche, basate sulle autovalutazioni espresse dall'utente.
- Durante un'**Esercitazione**, per ogni Flashcard, l'utente visualizza la sola domanda sul “fronte” della carta. In qualunque momento, l'utente può interagire con la carta per rivelarne la risposta.  
L'utente non esprime alcuna valutazione durante una Esercitazione.  
Nessuna informazione aggiuntiva è presentata al termine dell'Esercitazione, che può essere conclusa in qualunque momento.

## Tabella dei Requisiti

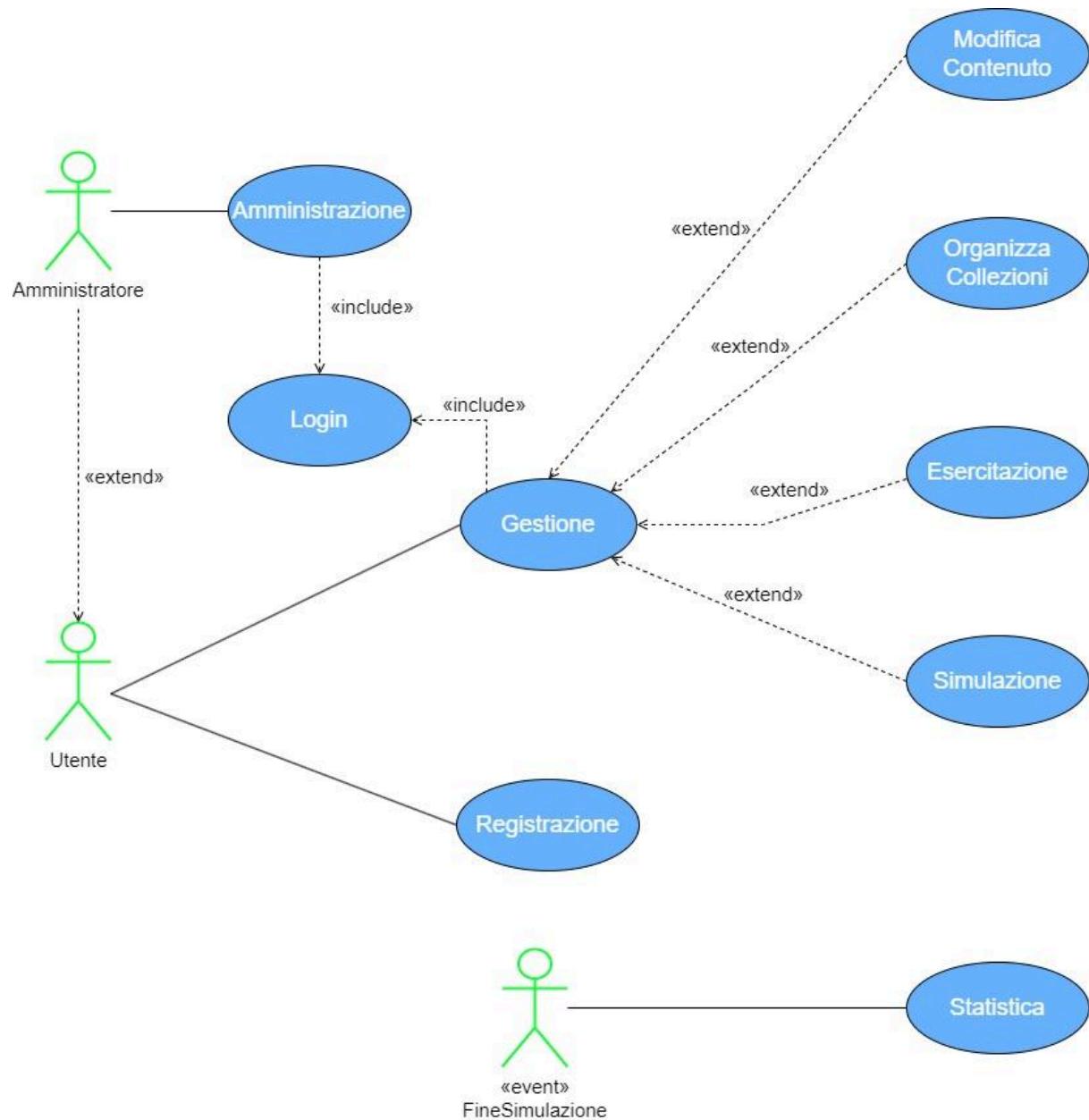
ID	Requisito	Tipo
R1F	Autenticazione Utente con username e password	Funzionale
R2F	Registrazione per utenti che non dispongono di credenziali	Funzionale
R3F	Il Sistema permette all'Utente di interagire solo con Collezioni di FlashCard	Funzionale
R4F	L'Utente crea una Collezione specificando il nome. Successivamente, la potrà popolare con Carte	Funzionale

<b>R5F</b>	E' possibile alterare il contenuto di una Collezione esistente	Funzionale
<b>R6F</b>	La Creazione delle carte avviene specificando domanda, risposta ed eventuale argomento	Funzionale
<b>R7F</b>	Le modalità Esercitazione e Simulazione sono riferite alle sole carte appartenenti alla Collezione selezionata	Funzionale
<b>R8F</b>	L'Utente può scegliere di effettuare un quiz comprendente le sole carte appartenenti ad uno specifico argomento	Funzionale
<b>R9F</b>	Prima dell'inizio di un Esercitazione o di una Simulazione è possibile scegliere la lunghezza di questa	Funzionale
<b>R10F</b>	Durante una Simulazione, l'utente esprime una valutazione per ogni carta	Funzionale
<b>R11F</b>	Durante un'Esercitazione o una Simulazione, l'utente può navigare tra le varie carte che compongono il quiz	Funzionale
<b>R12F</b>	I progressi di una Simulazione o un'Esercitazione sono persi in caso di interruzioni	Funzionale
<b>R13F</b>	Nella sola modalità Esercitazione, le risposte sul retro delle carte sono visualizzabili.	Funzionale
<b>R14F</b>	Un Utente può terminare una Esercitazione in qualunque momento	Funzionale
<b>R15F</b>	Alla conclusione di una Simulazione, sono presentate statistiche all'utente riguardo la sua prestazione	Funzionale
<b>R16F</b>	L'Utente può concludere una Simulazione solo dopo aver espresso una valutazione per ognuna delle carte proposte	Funzionale
<b>R17F</b>	L'Amministratore di sistema ha accesso ad alcune funzioni di controllo	Funzionale
<b>R1NF</b>	Appoggio ad una struttura di memorizzazione dati. Il database in questione non trattiene la password scelta dall'utente, bensì un hash derivato da questa	Non Funzionale
<b>R2NF</b>	Si predispongono credenziali di accesso complesse per l'amministratore. Queste saranno protette con una maggiore attenzione rispetto alle credenziali di un utente	Non Funzionale
<b>R3NF</b>	Il Sistema ammette solo l'esistenza di carte aventi <u>sia</u> una domanda <u>sia</u> una risposta	Non Funzionale
<b>R4NF</b>	Velocità di ricerca dati e caricamento delle funzionalità	Non Funzionale

## Vocabolario

Voce	Definizione	Sinonimi
<u>Utente</u>	Colui che usufruisce dei servizi offerti dall'applicazione	
<u>Amministratore</u>	Il supervisore dell'applicazione, in grado di svolgere alcune funzioni di controllo	Admin
<u>FlashCard</u>	Insieme di informazioni inserite dall'utente stesso per consolidare una nozione. La FlashCard è composta da una Domanda e una Risposta, più un eventuale Argomento	Carta
<u>Domanda</u>	Simbolico "Fronte" della FlashCard	Quesito
<u>Risposta</u>	Simbolico "Retro" della FlashCard	
<u>Argomento</u>	Attributo aggiuntivo che una o più FlashCard della stessa Collezione possono condividere	Capitolo, Etichetta
<u>Username</u>	Pseudonimo scelto dall'Utente al momento della registrazione	Nome utente
<u>Password</u>	Codice alfanumerico	Parola d'accesso
<u>Collezione</u>	Insieme di FlashCard di dimensione variabile	Mazzo
<u>Quiz</u>	Termine generico per indicare una delle due funzionalità in cui l'Utente fa uso delle FlashCard di una Collezione per valutare la propria preparazione sul contenuto di questa	Simulazione, Esercitazione
<u>Esercitazione</u>	Modalità in cui è possibile svelare le risposte delle varie carte. Può essere terminata in qualunque momento	
<u>Simulazione</u>	Modalità in cui non è possibile svelare le risposte delle varie carte. Dopo aver espresso una valutazione per ogni carta, sono presentate alcune statistiche riguardo l'esito conseguito dall'Utente.	Prova
<u>Valutazione</u>	Giudizio autonomo espresso dall'Utente stesso circa la sua preparazione riguardo la Domanda riportata su una FlashCard	
<u>Statistica</u>	Raccolta di dati e successiva analisi delle Valutazioni espresse dall'Utente durante una Simulazione	

## Casi d'uso - Grafico



## Casi d'uso - Scenari

Titolo	Amministrazione
Descrizione	L'Amministratore ha il compito di gestire il Sistema
Attori	Amministratore
Relazioni	Login
Precondizioni	Possesso delle credenziali di Amministratore
Postcondizioni	

<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Amministratore inserisce le credenziali adeguate e viene identificato</li> <li>2. Nella schermata che si presenta, questi può eseguire funzioni di controllo sul Sistema</li> </ol>
<b>Scenari Alternativi</b>	
<b>Requisiti non funzionali</b>	Si predispongono credenziali di accesso complesse per l'amministratore. Queste saranno protette con una maggiore attenzione rispetto alle credenziali di un utente
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Login</b>
<b>Descrizione</b>	L'Utente si identifica tramite credenziali
<b>Attori</b>	Utente, Amministratore
<b>Relazioni</b>	Amministrazione, Gestione
<b>Precondizioni</b>	
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente inserisce le credenziali di accesso al Sistema</li> <li>2. Il Sistema, riconosce le credenziali e presenta la maschera adeguata</li> </ol>
<b>Scenari Alternativi</b>	<p><i>Scenario A - Credenziali non riconosciute</i>  Il Sistema non riconosce i dati immessi, fornendo all'utente la possibilità di tentare di nuovo</p>
<b>Requisiti non funzionali</b>	Appoggio ad una struttura di memorizzazione dati. Il database in questione non trattiene la password scelta dall'utente, bensì un hash derivato da questa.  Velocità di ricerca dati e caricamento delle funzionalità
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Registrazione</b>
<b>Descrizione</b>	Permette agli Utenti di registrarsi alla piattaforma
<b>Attori</b>	Utente
<b>Relazioni</b>	
<b>Precondizioni</b>	
<b>Postcondizioni</b>	L'Utente è ora memorizzato nel Sistema

<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente inserisce le nuove credenziali di accesso al Sistema</li> <li>2. Il Sistema registra internamente l'Utente</li> </ol>
<b>Scenari Alternativi</b>	<p><i>Scenario A: Credenziali già presenti</i>  Le credenziali inserite dall'Utente appartengono a un Utente già registrato nel Sistema. Quest'ultimo propone nuovamente la schermata di Registrazione.</p>
<b>Requisiti non funzionali</b>	Appoggio ad una struttura di memorizzazione dati. Il database in questione non trattiene la password scelta dall'utente, bensì un hash derivato da questa.
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Gestione</b>
<b>Descrizione</b>	L'Utente interagisce con l'applicazione
<b>Attori</b>	Utente
<b>Relazioni</b>	Login, Organizza Collezioni, Simulazione, Esercitazione, Modifica Contenuto
<b>Precondizioni</b>	L'identificazione dell'Utente è avvenuta con successo
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente usufruisce di una delle funzionalità del Sistema, interagendovi</li> </ol>
<b>Scenari Alternativi</b>	
<b>Requisiti non funzionali</b>	
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Modifica Contenuto</b>
<b>Descrizione</b>	L'Utente sceglie una Collezione e ne modifica il contenuto
<b>Attori</b>	Utente
<b>Relazioni</b>	Gestione
<b>Precondizioni</b>	L'identificazione dell'Utente è avvenuta con successo
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente sceglie la Collezione da modificare</li> <li>2. Il Sistema presenta le varie Carte della Collezione, se presenti.</li> <li>3. L'Utente modifica liberamente le Carte, ne aggiunge di nuove oppure elimina alcune di queste</li> </ol>

<b>Scenari Alternativi</b>	
<b>Requisiti non funzionali</b>	
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Organizza Collezioni</b>
<b>Descrizione</b>	L'Utente crea una Collezione di FlashCard
<b>Attori</b>	Utente
<b>Relazioni</b>	Gestione
<b>Precondizioni</b>	L'identificazione dell'Utente è avvenuta con successo
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente crea o rimuove una Collezione</li> <li>2. La Collezione è ora memorizzata ed appare nella schermata dell'Utente</li> </ol>
<b>Scenari Alternativi</b>	
<b>Requisiti non funzionali</b>	
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Simulazione</b>
<b>Descrizione</b>	L'Utente prende parte ad una Simulazione
<b>Attori</b>	Utente
<b>Relazioni</b>	Gestione
<b>Precondizioni</b>	L'identificazione dell'Utente è avvenuta con successo
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente sceglie la Collezione su cui desidera esercitarsi</li> <li>2. L'Utente avvia una Simulazione e sceglie la lunghezza di questa</li> <li>3. L'Utente sceglie se includere le sole carte appartenenti a un Argomento oppure carte scelte a caso</li> <li>4. Il Sistema mostra le varie carte all'Utente</li> <li>5. L'Utente esprime un giudizio per ogni carta presentatagli</li> <li>6. Dopo che tutte le carte sono state valutate, la Simulazione termina</li> </ol>
<b>Scenari</b>	

<b>Alternativi</b>	
<b>Requisiti non funzionali</b>	Velocità di ricerca dati e caricamento delle funzionalità
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Esercitazione</b>
<b>Descrizione</b>	L'Utente prende parte ad un'Esercitazione
<b>Attori</b>	Utente
<b>Relazioni</b>	Gestione
<b>Precondizioni</b>	L'identificazione dell'Utente è avvenuta con successo
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. L'Utente sceglie la Collezione</li> <li>2. L'Utente avvia un'Esercitazione e sceglie la lunghezza di questa</li> <li>3. L'Utente sceglie se includere le sole Carte appartenenti a un determinato Argomento</li> <li>4. Il Sistema mostra le varie Carte all'Utente</li> <li>5. L'Utente può decidere di visualizzare o meno la risposta di una specifica Carta</li> <li>6. L'Utente termina l'Esercitazione quando desidera</li> </ol>
<b>Scenari Alternativi</b>	
<b>Requisiti non funzionali</b>	Velocità di ricerca dati e caricamento delle funzionalità
<b>Punti aperti</b>	

<b>Titolo</b>	<b>Statistica</b>
<b>Descrizione</b>	Viene mostrata la Statistica della Simulazione appena conclusa
<b>Attori</b>	Evento "Fine Simulazione"
<b>Relazioni</b>	
<b>Precondizioni</b>	L'identificazione dell'Utente è avvenuta con successo
<b>Postcondizioni</b>	
<b>Scenario Principale</b>	<ol style="list-style-type: none"> <li>1. Alla fine della Simulazione, il Sistema mostra all'Utente la relativa Statistica</li> <li>2. L'Utente visualizza le informazioni presentate e successivamente ritorna alla maschera di Gestione</li> </ol>

<b>Scenari Alternativi</b>	
<b>Requisiti non funzionali</b>	
<b>Punti aperti</b>	

## Analisi del Rischio

### Tabelle di Valutazione dei Beni

Bene	Valore	Esposizione
<u>Sistema Informativo</u>	Alto	Alta - Costi ripristino sistema - Perdita di immagine
<u>Credenziali degli utenti</u>	Medio	Media - Perdita d'immagine in caso di divulgazione di informazioni personali
<u>Credenziali amministratore</u>	Alto - L'admin ha la possibilità di rimuovere gli utenti - Porterebbe alla potenziale perdita di tutte le credenziali utenti	Alta - Perdita di immagine - Perdita finanziaria - Necessità di accertamento dei danni e valutazione del comportamento da adottare
<u>Informazioni relative alle Collezioni</u>	Medio-Alto	Media-Alta - Impossibilità nel fornire i servizi - Perdita di immagine

### Tabelle delle Minacce / Controlli

Minaccia	Probabilità	Controllo	Fattibilità
<u>Furto delle credenziali degli utenti</u>	Media Le credenziali degli utenti sono criptate debolmente	La base di dati su cui l'applicazione si appoggia mantiene solo le informazioni strettamente necessarie.  Di queste, i dati più sensibili non sono	<b>Medio costo</b> Crittografia di media complessità

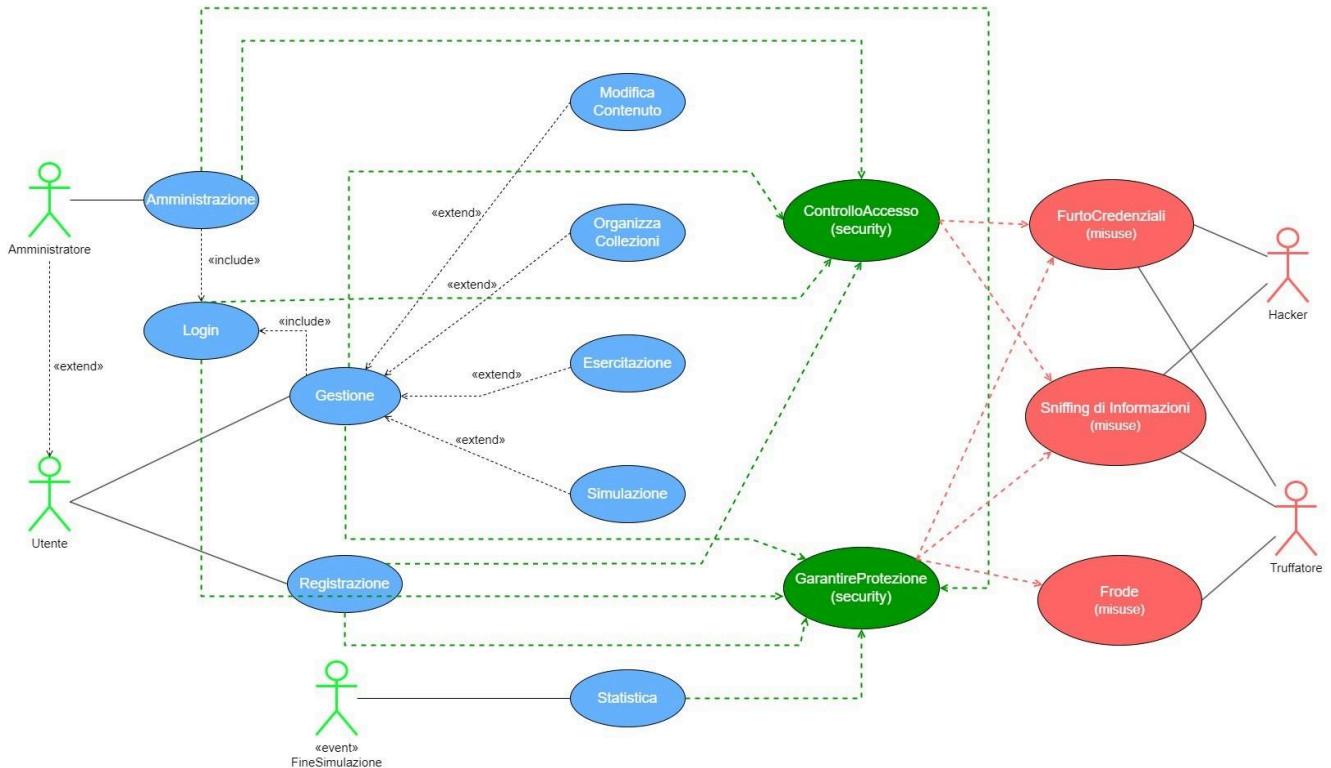
		mantenuti “in chiaro”	
<u>Furto credenziali amministratore</u>	<b>Bassa</b>  Ottenere le credenziali di amministratore è un’operazione tutt’altro che semplice	Predisposizione di misure aggiuntive per la protezione delle suddette credenziali	<b>Alto costo</b>  Crittografia più complessa richiede costi maggiori  Garantire la reperibilità di almeno un admin durante le diverse fasce orarie
<u>DoS</u>	<b>Bassa</b>	Progettazione adeguata tramite controllo e limitazione dei tentativi di accesso	<b>Basso costo</b>  Un attacco DoS bene eseguito non è prevenibile  Considerando le possibili perdite, non vale la pena stimare fondi per misure di prevenzione.
<u>Intercettazione delle comunicazioni</u>	<b>Alta</b>  Il sistema è distribuito e basato su interazioni con modello Cliente-Servitore.	Utilizzo di sistemi di comunicazione sicuri, eventualmente basati su crittografia	<b>Costo variabile in base alla cifratura scelta</b>
		Log di tutte le operazioni	<b>Costo implementativo quasi inesistente</b>

## Analisi della tecnologia dal punto di vista della sicurezza

Tecnologia	Vulnerabilità
<u>Autenticazione username/password</u>	<ul style="list-style-type: none"> <li>• Password banali</li> <li>• Utente condivide volontariamente o inconsciamente le sue credenziali</li> </ul>

<p><u>Cifratura della comunicazione</u></p>	<p>Le vulnerabilità dipendono dal tipo di cifratura:</p> <ul style="list-style-type: none"> <li>• Cifratura Simmetrica:           <ul style="list-style-type: none"> <li>a. Tempo di vita della chiave</li> <li>b. Lunghezza della chiave</li> <li>c. Metodo di memorizzazione della chiave</li> </ul> </li> </ul> <p>Per “tempo di vita della chiave” si intende ogni quanto viene generata una nuova chiave. Come è intuitivo, più dati sono cifrati dalla stessa chiave, più è facile venire a capo della cifratura.</p> <ul style="list-style-type: none"> <li>• Cifratura Asimmetrica           <ul style="list-style-type: none"> <li>a. Metodo di memorizzazione chiave privata</li> </ul> </li> </ul>
<p><u>Architettura Client/Server</u></p>	<ul style="list-style-type: none"> <li>• Attacco Man in the Middle</li> <li>• Intercettazione delle comunicazioni</li> <li>• Attacco DoS</li> </ul>

## Security: Use & Misuse Cases - Grafico



## Security: Use & Misuse Cases - Scenari

Titolo	Controllo Accesso	
<b>Descrizione</b>	Gli accessi devono essere controllati	
<b>Misuse case</b>	Furto di credenziali, Sniffing	
<b>Precondizioni</b>	L'attaccante possiede i mezzi per ottenere le credenziali di accesso al Sistema	
<b>Postcondizioni</b>	Il Sistema impedisce temporaneamente l'accesso e notifica un possibile tentativo di autenticazione non autorizzato	
<b>Scenario principale</b>	<u>Sistema</u>	<u>Attaccante</u>
		L'attaccante tenta di accedere al sistema tramite attacchi dizionario
<b>Scenario di un attacco avvenuto con successo</b>	Dopo alcuni accessi falliti, l'Amministratore viene notificato dell'attività sospetta dell'Utente	
	<u>Sistema</u>	<u>Attaccante</u>
		L'attaccante riesce a penetrare nel sistema
	Il sistema inserisce l'avvenuta autenticazione nei log	

Titolo	Garantire Protezione	
<b>Descrizione</b>	I dati del sistema devono essere protetti	
<b>Misuse case</b>	Sniffing, Man In The Middle, Frode	
<b>Precondizioni</b>	L'attaccante è in grado di intercettare le comunicazioni	
<b>Postcondizioni</b>	Il Sistema impedisce all'attaccante di trarre profitto dalle informazioni ottenute	
<b>Scenario principale</b>	<u>Sistema</u>	<u>Attaccante</u>
	Il sistema protegge i dati in circolo con sistemi di cifratura	
		Le informazioni che l'attaccante ha intercettato non sono utilizzabili

Scenario di un attacco avvenuto con successo	<u>Sistema</u>	<u>Attaccante</u>
		L'attaccante intercetta le comunicazioni, riuscendo a decifrare le informazioni
	Il sistema tiene traccia di ogni comunicazione e dell'esito di questa nel log di sistema	

## Requisiti aggiuntivi emersi

Dall'*Analisi del Rischio* si evincono le seguenti necessità:

1. Predisporre un file di log per tracciare tutte le azioni potenzialmente critiche che avvengono nel Sistema, visualizzabile unicamente da un Amministratore
2. Pattiuire una politica di controllo degli accessi
3. Stabilire un sistema di protezione per i dati in circolo

Vengono quindi introdotti due nuovi requisiti funzionali:

R18F	Lettura Log	Funzionale
R19F	Scrittura Log	Funzionale

# Analisi del Problema

## Analisi delle Funzionalità

Funzionalità	Tipo	Grado complessità	Requisiti collegati
<u>Amministrazione</u>  Composta da: - Visualizzazione Utenti - Rimozione Utenti - Visualizzazione Log	Gestione dati, interazione esterno	Complessa	R17F, R18F
<u>Login</u>	Gestione dati, interazione esterno	Semplice	R1F, R1NF, R2NF
<u>Registrazione</u>	Gestione dati, interazione esterno	Semplice	R2F, R1NF
<u>Gestione</u>  Composta da: - Modifica Contenuto	Gestione dati, Interazione esterno	Complesso	R3F, R5F, R6F, R7F, R8F, R9F, R10F, R11F, R12F, R13F, R14F, R15F,

- Organizza Collezioni - Esercitazione - Simulazione			R16F R3NF
<u>Statistica</u>	Interazione esterno	Semplice	R14F
<u>Log</u>	Gestione dati	Semplice	R18F, R19F

Di seguito presentiamo le *Tabelle di Informazioni/Flusso* per ognuna delle funzionalità sopra individuate. Ai fini di leggibilità, si riporta una tabella informazioni/flusso distinta per ognuna delle sotto-funzionalità in metodi complessi Gestione ed Amministrazione sono stati scomposti.

#### Visualizzazione Utenti: Tabella Informazioni/Flusso

Informazione	Tipo	Livello protezione / Privacy	Input / Output	Vincoli
Nome Utente	Semplice	Medio	Output	Non più di 128 caratteri
Data Ultimo Accesso	Semplice	Non protetto	Output	

#### Rimozione Utenti: Tabella Informazioni/Flusso

Informazione	Tipo	Livello protezione / Privacy	Input / Output	Vincoli
Username	Semplice	Non protetto	Input	L'Utente deve figurare tra quelli memorizzati nel Sistema

#### Visualizzazione Log: Tabella Informazioni/Flusso

Informazione	Tipo	Livello protezione / Privacy	Input / Output	Vincoli
Timestamp	Semplice	Bassa	Output	
Descrizione	Semplice	Bassa	Output	Non più di 256 caratteri

#### Login: Tabella Informazioni/Flusso

Informazione	Tipo	Livello	Input / Output	Vincoli

		<b>protezione / Privacy</b>		
Username	Semplice	Alto	Input	Non più di 128 caratteri
Password	Semplice	Alto	Input	Non più di 128 caratteri

Registrazione: Tabella Informazioni/Flusso

<b>Informazione</b>	<b>Tipo</b>	<b>Livello protezione / Privacy</b>	<b>Input / Output</b>	<b>Vincoli</b>
Username	Semplice	Alto	Input	Non più di 128 caratteri
Password	Semplice	Alto	Input	Non più di 128 caratteri

Organizza Collezioni: Tabella Informazioni/Flusso

<b>Informazione</b>	<b>Tipo</b>	<b>Livello protezione / Privacy</b>	<b>Input / Output</b>	<b>Vincoli</b>
Nome Collezione	Semplice	Non protetto	Input	Non più di 128 caratteri

Modifica Contenuto: Tabella Informazioni/Flusso

<b>Informazione</b>	<b>Tipo</b>	<b>Livello protezione / Privacy</b>	<b>Input / Output</b>	<b>Vincoli</b>
Domanda (fronte)	Semplice	Non protetto	Input	Non più di 256 caratteri
Risposta (retro)	Semplice	Non protetto	Input	Non più di 256 caratteri
Argomento	Semplice	Non protetto	Output/Input Opzionale	Non più di 256 caratteri
Flashcard	Complesso	Non protetto	Output	
Collezione	Complesso	Non protetto	Output	

Esercitazione: Tabella Informazioni/Flusso

<b>Informazione</b>	<b>Tipo</b>	<b>Livello</b>	<b>Input / Output</b>	<b>Vincoli</b>
---------------------	-------------	----------------	-----------------------	----------------

		<b>protezione / Privacy</b>		
Collezione	Complesso	Non protetto	Input	Collezione Esistente
Lunghezza	Semplice	Non protetto	Input	Non superiore al numero di carte della Collezione
Argomento Scelto	Semplice	Non protetto	Input Opzionale	Scelto tra gli argomenti che le carte della Collezione posseggono

Simulazione: Tabella Informazioni/Flusso

<b>Informazione</b>	<b>Tipo</b>	<b>Livello protezione / Privacy</b>	<b>Input / Output</b>	<b>Vincoli</b>
Collezione	Complesso	Non protetto	Input	Collezione Esistente
Lunghezza	Semplice	Non protetto	Input	Non superiore al numero di carte della Collezione
Argomento Scelto	Semplice	Non protetto	Input Opzionale	Scelto tra gli argomenti che le carte della Collezione posseggono
Valutazione espressa dall'utente	Semplice	Non protetto	Input	Enumerativo [ "positivo" / "negativo" ]

Statistica: Tabella Informazioni/Flusso

<b>Informazione</b>	<b>Tipo</b>	<b>Livello protezione / Privacy</b>	<b>Input / Output</b>	<b>Vincoli</b>
Tempo Impiegato	Semplice	Non protetto	Input	
Numero Carte Valutate Positivamente	Semplice	Non protetto	Input	

Numero Carte Negativamente	Semplice	Non protetto	Input	
Esito Complessivo	Semplice	Non protetto	Output	
Capitolo Peggior	Semplice	Non protetto	Output	

Log : Tabella Informazioni/Flusso

Informazione	Tipo	Livello Protezione / privacy	Input/Output	Vincoli
Timestamp	Semplice	Medio	Input	
Tipo Operazione	Semplice	Alto	Input	Valori ammessi: - Login - Registrazione - Creazione - Modifica
Descrizione	Semplice	Alto	Input	Non più di 256 caratteri

## Analisi dei Vincoli

Requisito	Categorie	Impatto	Funzionalità
Appoggio ad una struttura di memorizzazione dati. Il database in questione non trattiene la password scelta dall'utente, bensì un hash derivato da questa	Sicurezza	Peggior tempo di risposta. Maggior privacy dei dati	Login, Registrazione
Si predispongono credenziali di accesso per l'Amministratore complesse e protette con una maggiore attenzione rispetto alle credenziali di un utente	Sicurezza	Peggior tempo di risposta. Protezione credenziali più efficace	Amministrazione, Login
Velocità di ricerca dati e caricamento delle funzionalità	Tempo di Risposta	Migliorare l'esperienza dell'utente	Login, Simulazione, Esercitazione

# Analisi delle Interazioni

Tabella Maschere

Maschera	Informazioni	Funzionalità
<u>Home Amministrazione</u>	Messaggio benvenuto Elenco delle funzionalità di Controllo	Amministrazione
<u>View Elenco Utenti</u>	Elenco Utenti: Username e Data Ultimo Accesso	Amministrazione
<u>View Log</u>	Elenco delle righe del file Log: Timestamp, azione, Username	Amministrazione
<u>View Login</u>	Username, Password	Login
<u>View Registrazione</u>	Username, Password	Registrazione
<u>Home Utente</u>	Messaggio di benvenuto, Elenco delle Collezioni dell'Utente	Gestione
<u>View Crea Collezione</u>	Nome nuova Collezione	Organizza Collezioni
<u>View Gestione Collezione</u>	Nome Collezione Informazioni delle FlashCards presenti	Organizza Collezioni, Modifica Contenuto
<u>View Aggiungi Carta</u>	Informazioni della FlashCard aggiunta	Modifica Contenuto
<u>View Altera Carta</u>	Informazioni (modificate e non) della FlashCard	Modifica Contenuto
<u>View Preliminare Quiz</u>	Lunghezza del Quiz scelta dall'Utente Eventuale Argomento scelto	Esercitazione, Simulazione
<u>View Esercitazione</u>	Informazioni della FlashCard estratta	Esercitazione
<u>View Simulazione</u>	Informazioni della FlashCard estratta (Auto)Valutazione espressa dall'Utente	Simulazione

<u>View Statistiche</u>	Tempo impiegato Percentuale Valutazioni Positive, Percentuale Valutazioni Negative, Capitolo con la valutazione peggiore	Statistiche
-------------------------	---	-------------

## Tabella Sistemi Esterni

Il Sistema ideato non fa uso di terzi, rispecchiando la visione di semplicità alla base del progetto.

## Analisi dei Ruoli e delle Responsabilità

### Tabella dei Ruoli

Ruolo	Responsabilità	Maschere	Riservatezza	Numerosità
<u>Amministratore</u>	Gestione degli Utenti  Rimozione di Utenti inattivi o sospetti  Controllo dei Log	View Login  Home Admin, View Elenco Utenti  Home Log, View Log	Massimo grado di Riservatezza	Considerando l'alternanza, dai 2 ai 4 Amministratori
<u>Utente</u>	Usufruisce dei servizi del Sistema: gestisce le proprie Collezioni, interagendovi come desidera	View Login, View Registrazione, Home Gestione  View Gestione Collezione, View Aggiungi Carta, View Altera Carta  View Preliminare Quiz, View Esercitazione, View Simulazione, View Statistiche	Grado di riservatezza medio	Numero massimo limitato dalle risorse del sistema

## Amministratore: Tabella Ruolo-Informazioni

Informazione	Tipo di Accesso
Username	Lettura
Data dall'Ultimo Accesso	Lettura
Entry Log File	Lettura
Utente da Rimuovere	Lettura/Scrittura

## Utente: Tabella Ruolo-Informazioni

Informazione	Tipo di Accesso
Username	Lettura
Collezione	Lettura/Scrittura
Domanda	Lettura/Scrittura
Risposta	Lettura/Scrittura
Argomento	Lettura/Scrittura
FlashCard	Lettura/Scrittura
Statistica	Lettura

## Scomposizione del Problema

### Tabella Scomposizione delle Funzionalità

Informazione	Scomposizione
<u>Amministrazione</u>	Visualizzazione Utenti Rimozione Utenti Visualizzazione Log

Informazione	Scomposizione
<u>Gestione</u>	Modifica Contenuto Organizza Collezioni Esercitazione Simulazione

### Amministrazione: Tabella Sotto-Funzionalità

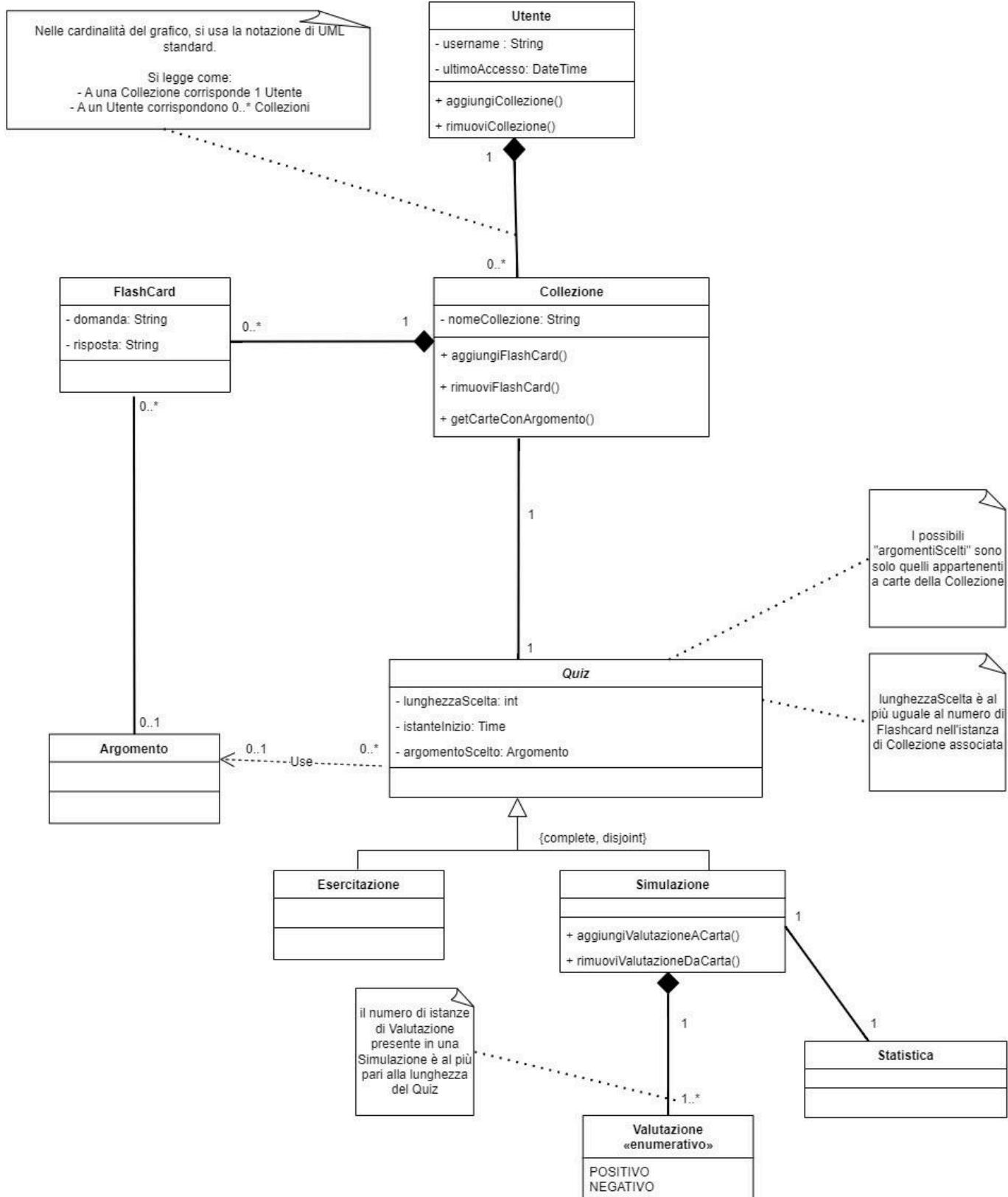
Sotto-funzionalità	Sotto-funzionalità	Legame	Informazioni
Visualizzazione Utenti	Rimozione Utenti	Si possono rimuovere solo gli Utenti elencati	Username

### Gestione: Tabella Sotto-Funzionalità

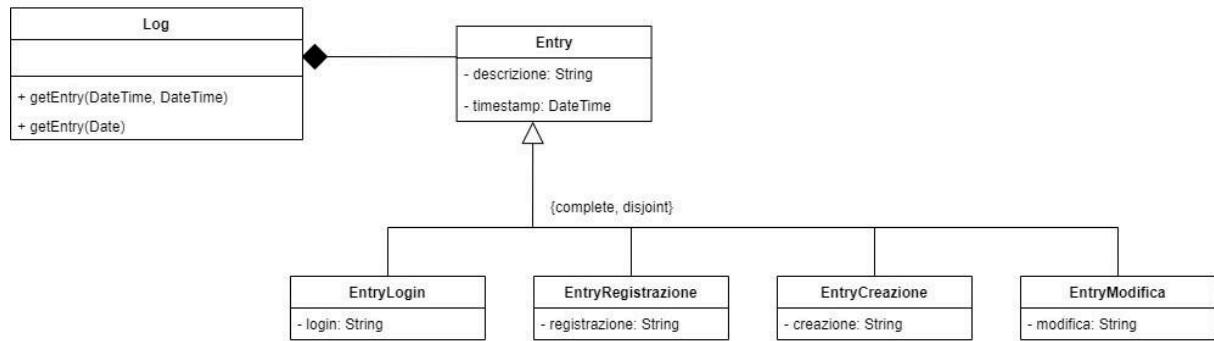
Sotto-funzionalità	Sotto-funzionalità	Legame	Informazioni
Organizza Collezioni	Modifica Contenuto	Non si può modificare una Collezione non esistente	Collezione
Organizza Collezioni, Modifica Contenuto	Esercitazione, Simulazione	Non si possono avviare Esercitazioni o Simulazioni su Collezioni non esistenti o vuote	Collezione

# Creazione del Modello di Dominio

## Modello del Dominio - Core

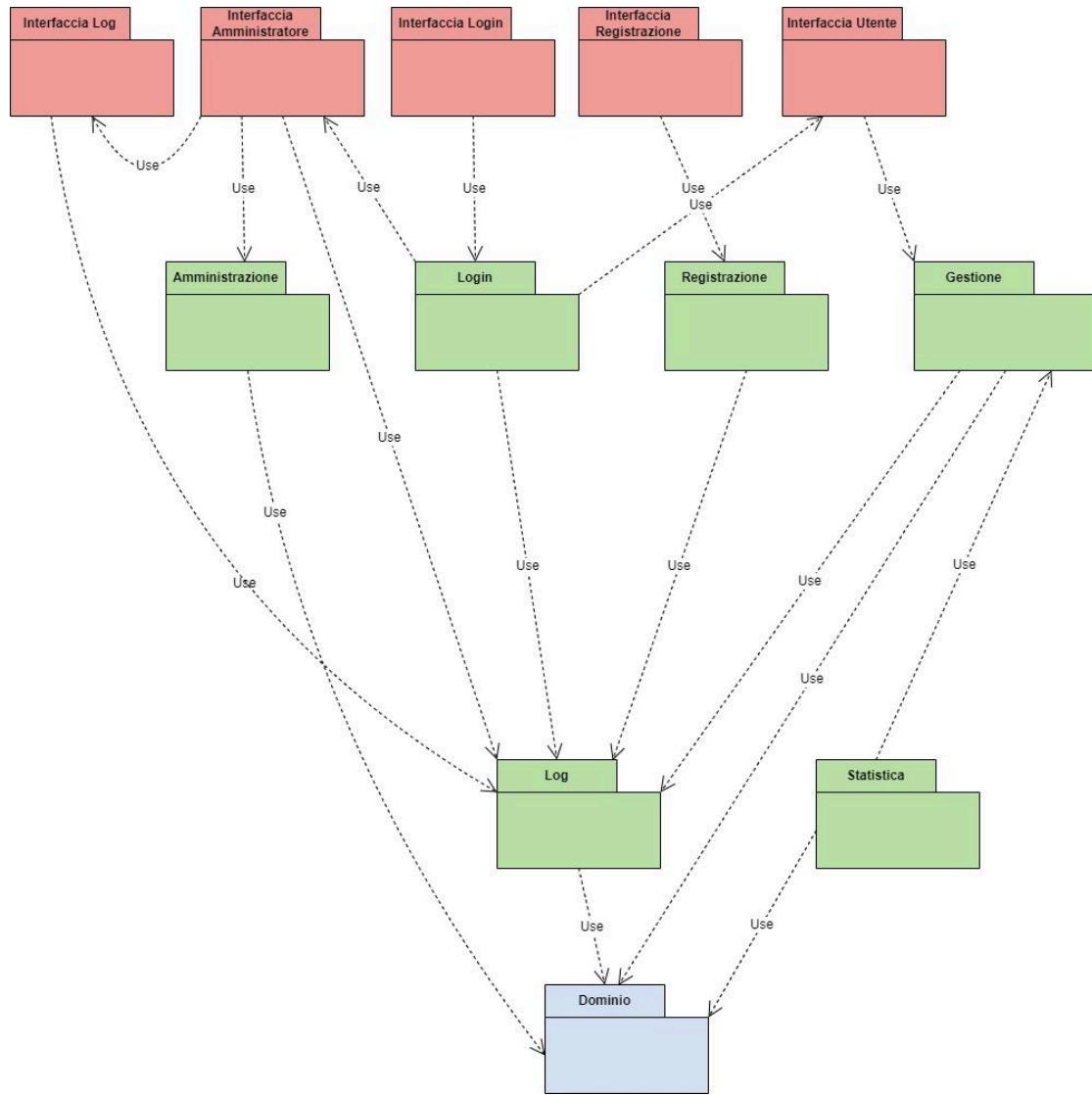


## Modello del Dominio - Log



# Architettura logica: Struttura

## Diagramma dei Package



## Diagrammi delle Classi

### Diagramma delle Classi - Dominio

Il diagramma delle Classi riguardante il dominio non è mostrato, in quanto già realizzato nel **Modello di Dominio**

Diagramma delle Classi - Amministrazione

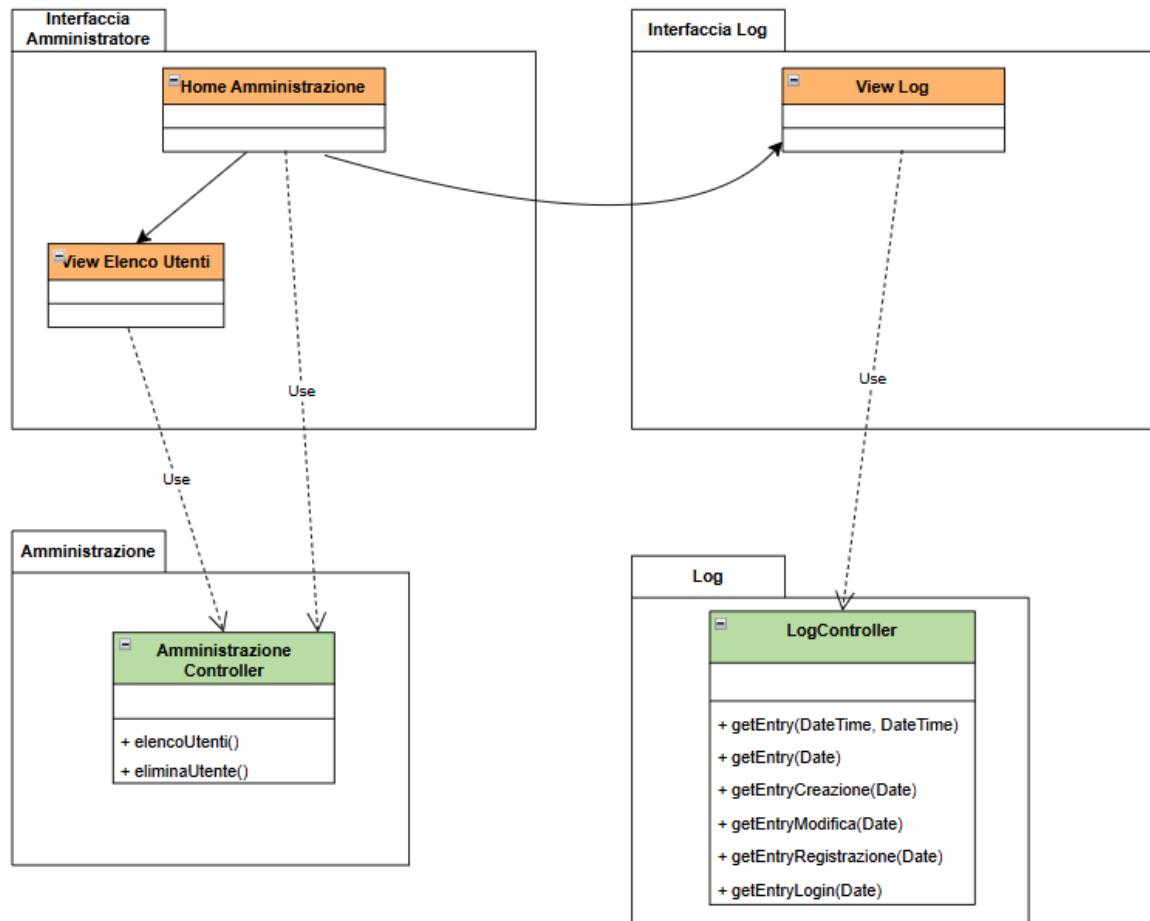
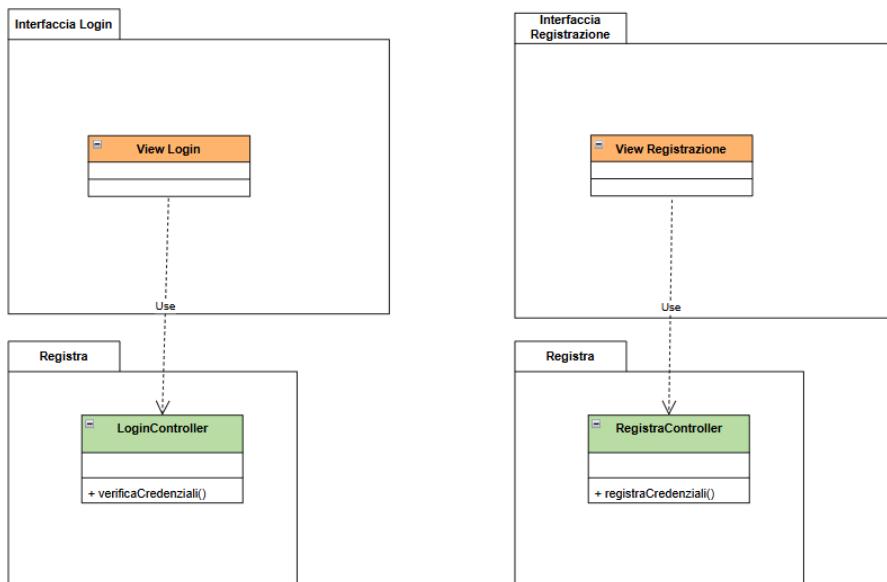
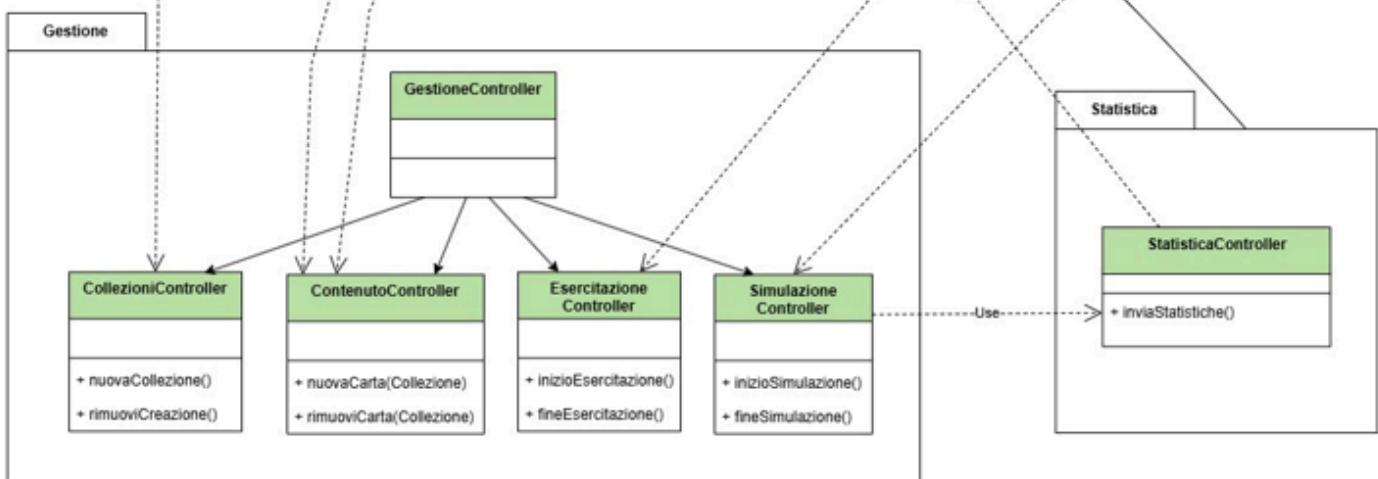
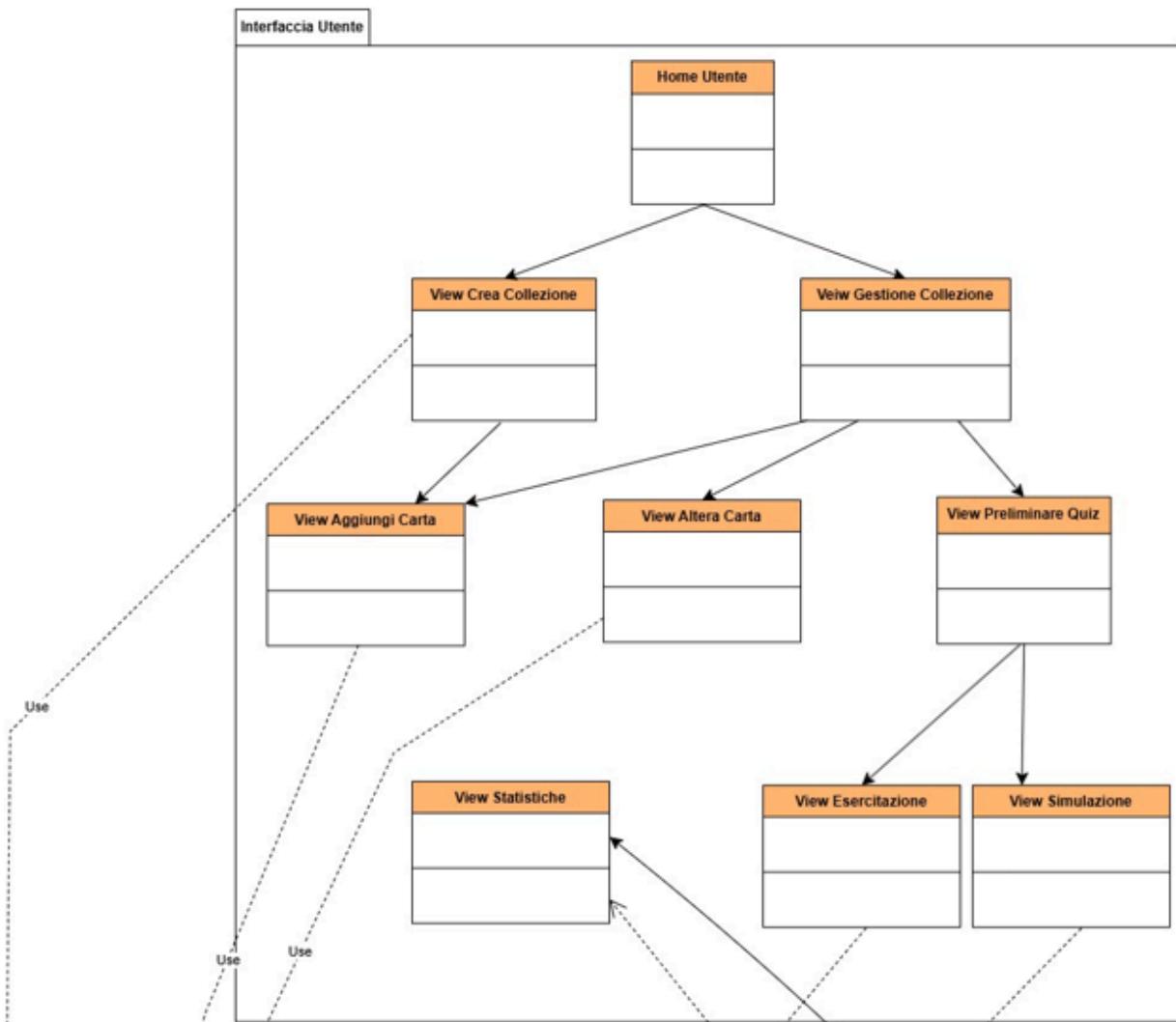


Diagramma delle Classi - Login e Registrazione

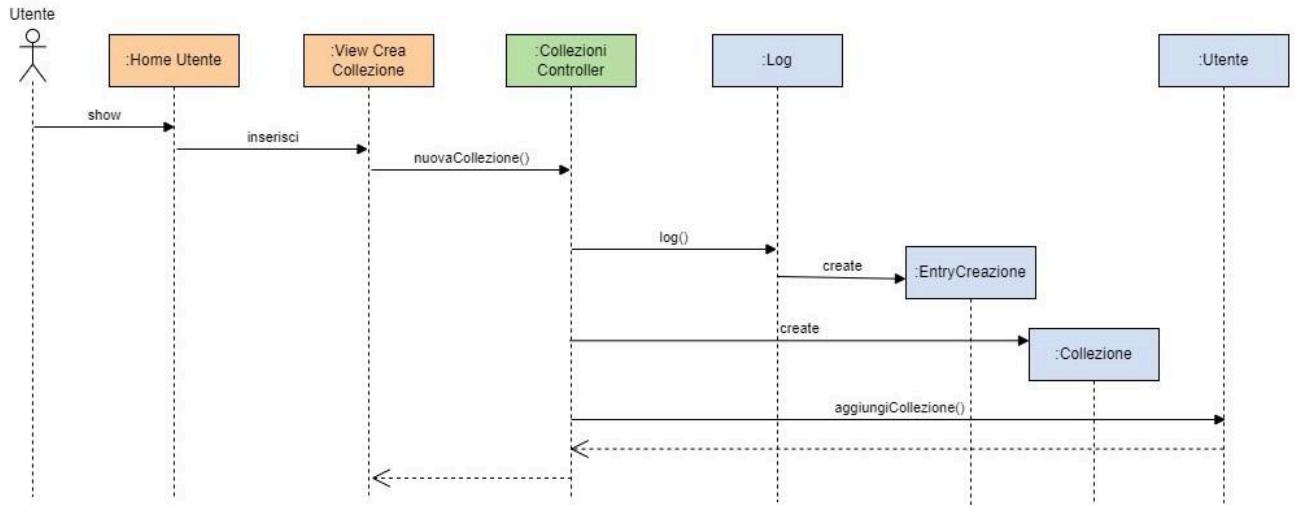


## Diagramma delle Classi - Interfaccia Utente

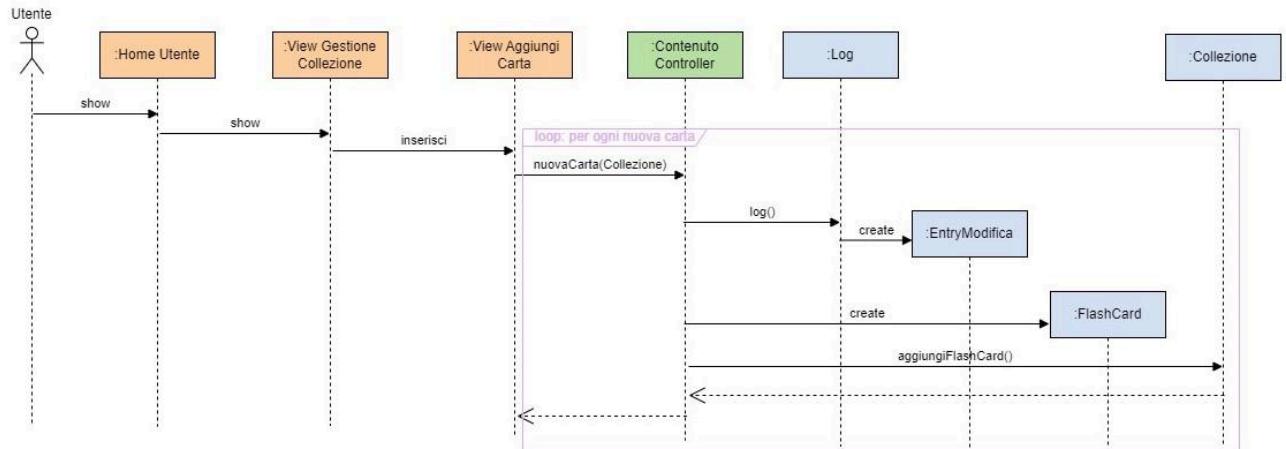


# Architettura logica: Interazione

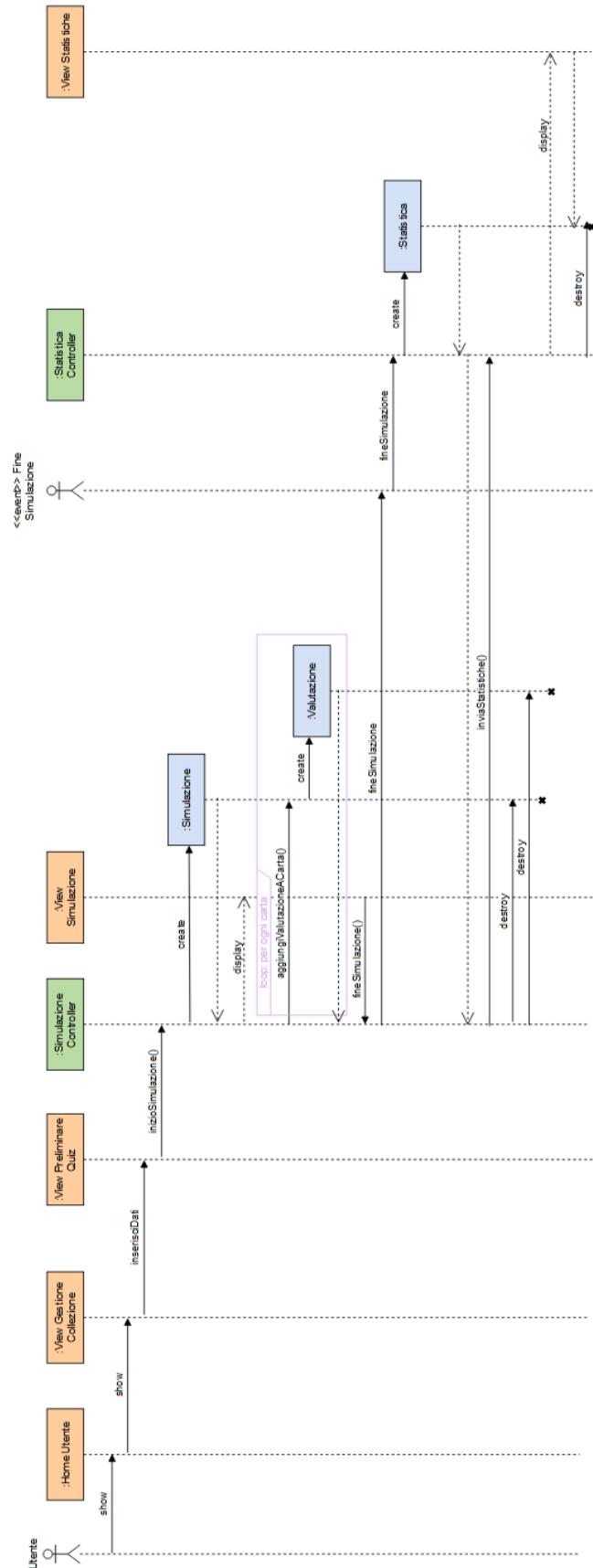
## Diagramma di Sequenza: Creazione Collezione



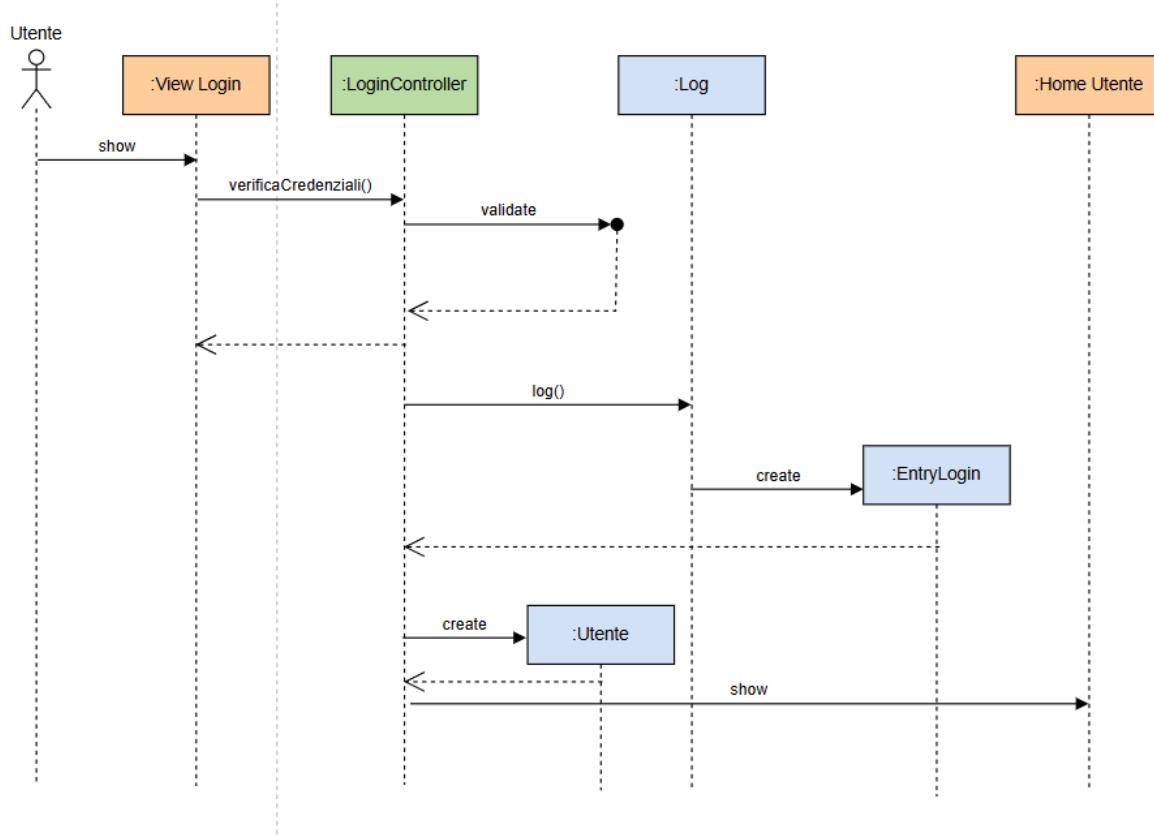
## Diagramma di Sequenza: Aggiunta Carta



## Diagramma di Sequenza: Simulazione



## Diagramma di Sequenza: Login



Non si ritiene necessario inserire diagrammi di stato a seguito della mancanza di algoritmi complessi al punto da richiederli.

## Piano di Lavoro

Il progetto e lo sviluppo del sistema sono assegnati ai vari membri del team, come indicato nella tabella sottostante.

Package	Progetto	Sviluppo
<u>Dominio</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Interfaccia Log</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Interfaccia Amministratore</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Interfaccia Login</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Interfaccia Registrazione</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Interfaccia Utente</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Gestione</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi

<u>Login</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Registrazione</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Selezione Modalità</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Scrittura Log</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi
<u>Statistica</u>	Catellani, Barbieri, Minguzzi	Catellani, Barbieri, Minguzzi

I tempi di rilascio previsti sono i seguenti :

- Progettazione entro 3 settimane dalla data odierna
- Sviluppo delle singole parti con collaudo unitario entro 2 settimane dalla fine della progettazione
- Integrazione e testing entro una settimana dalla fine dello sviluppo

## Prototipo

Si presuppone che, nel prototipo da realizzare, non tutte le funzionalità del Sistema saranno implementate. In particolare:

- Non verranno stabiliti canali di comunicazione sicura con tecnologia HTTPS
- Il Sistema si baserà su un DB fittizio, organizzato come una struttura dati complessa conservata in locale
- Non verranno realizzati i sistemi di gestione dei log

## Sviluppi futuri

Attualmente, non sono stati definiti futuri sviluppi per l'applicazione. Di seguito, sono riportate alcune ipotesi di venture estensioni:

### 1 - Prevedere l'aggiunta di un servizio a pagamento

Predisporre la possibilità agli utenti, per una cifra mensile irrisoria, di iscriversi all'abbonamento Premium. Utenti Premium dispongono di alcuni vantaggi: numero illimitato di Collezioni e la possibilità di salvare i propri progressi di un quiz per riprendere in un secondo momento. All'atto pratico, questo richiederebbe una revisione del **Modello del Dominio** e della **Tabella delle Funzionalità**, oltre che un considerevole cambio di approccio dal punto di vista della Security. Inoltre, occorrerebbe individuare un servizio di pagamento affidabile (Paypal o simile) a cui affidarsi.

### 2 - Porting a mobile

La semplicità del Sistema lo predisponde per un porting verso dispositivi mobili. Con alcuni cambiamenti che sfruttano la browser cache per mantenere l'utente autenticato e la memoria fisica del dispositivo per memorizzare le Collezioni, sarebbe persino possibile rendere l'applicazione quasi completamente fruibile in assenza di connessione alla rete.

## Piano del Collaudo

Per garantire il corretto funzionamento del sistema sono necessari una gamma di test unitari. In questo caso, sono presenti i test della classi FlashCard e Utente:

```
public class FlashCardTest {  
  
    private FlashCard flashCard;  
  
    @Before  
    public void setUp() {  
        flashCard = new FlashCard(1,"Domanda", "Risposta");  
    }  
  
    @Test  
    public void testCostruttore() {  
        Assert.assertEquals("", flashCard.getArgomento());  
    }  
  
    @Test  
    public void testGetDomanda() {  
        Assert.assertEquals("Domanda", flashCard.getDomanda());  
    }  
  
    @Test  
    public void testGetRisposta() {  
        Assert.assertEquals("Risposta", flashCard.getRisposta());  
    }  
  
    @Test  
    public void testSetDomanda() {  
        flashCard.setDomanda("Nuova domanda");  
        Assert.assertEquals("Nuova domanda", flashCard.getDomanda());  
    }  
  
    @Test  
    public void testSetRisposta() {  
        flashCard.setRisposta("Nuova risposta");  
        Assert.assertEquals("Nuova risposta", flashCard.getRisposta());  
    }  
  
    @Test  
    public void testImpostaArgomento() {  
        flashCard.impostaArgomento("Argomento");  
        Assert.assertEquals("Argomento", flashCard.getArgomento());  
    }  
}
```

```

public class UtenteTest {
    private Utente utente;

    @Before
    public void setUp() {
        utente = new Utente("Utente1");
    }

    @Test
    public void testCostruttore() {
        Assert.assertEquals("Utente1", utente.getName());
        Assert.assertEquals(0, utente.getCollezioni().size());
    }

    @Test
    public void testGetName() {
        Assert.assertEquals("Utente1", utente.getName());
    }

    @Test
    public void testSetName() {
        utente.setName("Utente2");
        Assert.assertEquals("Utente2", utente.getName());
    }

    @Test
    public void testAddCollezione() {
        utente.addCollezione(new Collezione(1, "Collezione1"));
        Assert.assertEquals(1, utente.getCollezioni().size());
    }

    @Test
    public void testRemoveCollezione() {
        utente.addCollezione(new Collezione(1, "Collezione1"));
        utente.removeCollezione(1);
        Assert.assertEquals(0, utente.getCollezioni().size());
    }

    @Test
    public void testGetCollezione() {
        utente.addCollezione(new Collezione(1, "Collezione1"));
        Assert.assertEquals("Collezione1", utente.getCollezione(1).getName());
    }
}

```

# Progettazione

## Progettazione Architetturale

### Requisiti non funzionali

Dalla **Analisi dei Vincoli** sono emersi requisiti non funzionali che impongono le seguenti limitazioni sul sistema:

- Usabilità
- Tempo di risposta
- Sicurezza (del Sistema e dei Dati)

Il vincolo di Usabilità interessa la quasi totalità dei servizi: il Sistema si basa su funzionalità semplici e maschere intuitive fruite da svariati utenti. Nonostante vi sia attrito tra il requisito di Usabilità e quello di Sicurezza nella specifica situazione in cui gli utenti siano costretti a re-inserire le proprie credenziali a seguito dell'interruzione di una sessione, si tratta anche dell'unico scenario in cui questo conflitto è stato riscontrato. Conseguentemente, si è scelto di non considerare questi vincoli come contrastanti.

In quanto nel Sistema non sono presenti algoritmi o computazioni complesse che possano essere rese più efficienti, rispettare il vincolo di Tempo di Risposta ricade sulla scelta della tecnologia adeguata. In particolare, le funzionalità considerabili “più lente” riguardano la comunicazione tra il lato dell'applicazione dedicato alla sessione utente e la struttura in cui sono conservate le Collezioni di quest'ultimo (Model). La velocità di responso dell'Applicazione si può dire “inversamente proporzionale” a quanto i dati di questa siano sicuri. Inserire meccanismi di protezione rispecchia il vincolo di Sicurezza, ma conduce ad un'applicazione con tempi di risposta più longevi.

Diviene dunque necessaria un'architettura che si basi su un compromesso in cui la maggior attenzione è rivolta alla soddisfazione del vincolo di Sicurezza del Sistema, di Usabilità e Tempo di Risposta, eventualmente a scapito del vincolo sulla Sicurezza dei Dati.

Considerando i requisiti relativi alla sicurezza delle comunicazioni emersi nell'analisi del rischio, si è scelto di utilizzare comunicazioni cifrate per mezzo del protocollo HTTPS a livello applicativo.

## Scelta dell'architettura

La scelta architettonica ritenuta più idonea ricade sul modello Cliente/Servitore. Il Sistema verrà quindi strutturato come segue:

### **Layer 1-Client:**

- unico client per l'utente

### **Layer 2-Server:**

- unico server di gestione dell'applicazione

### **Layer 3-Persistenza:**

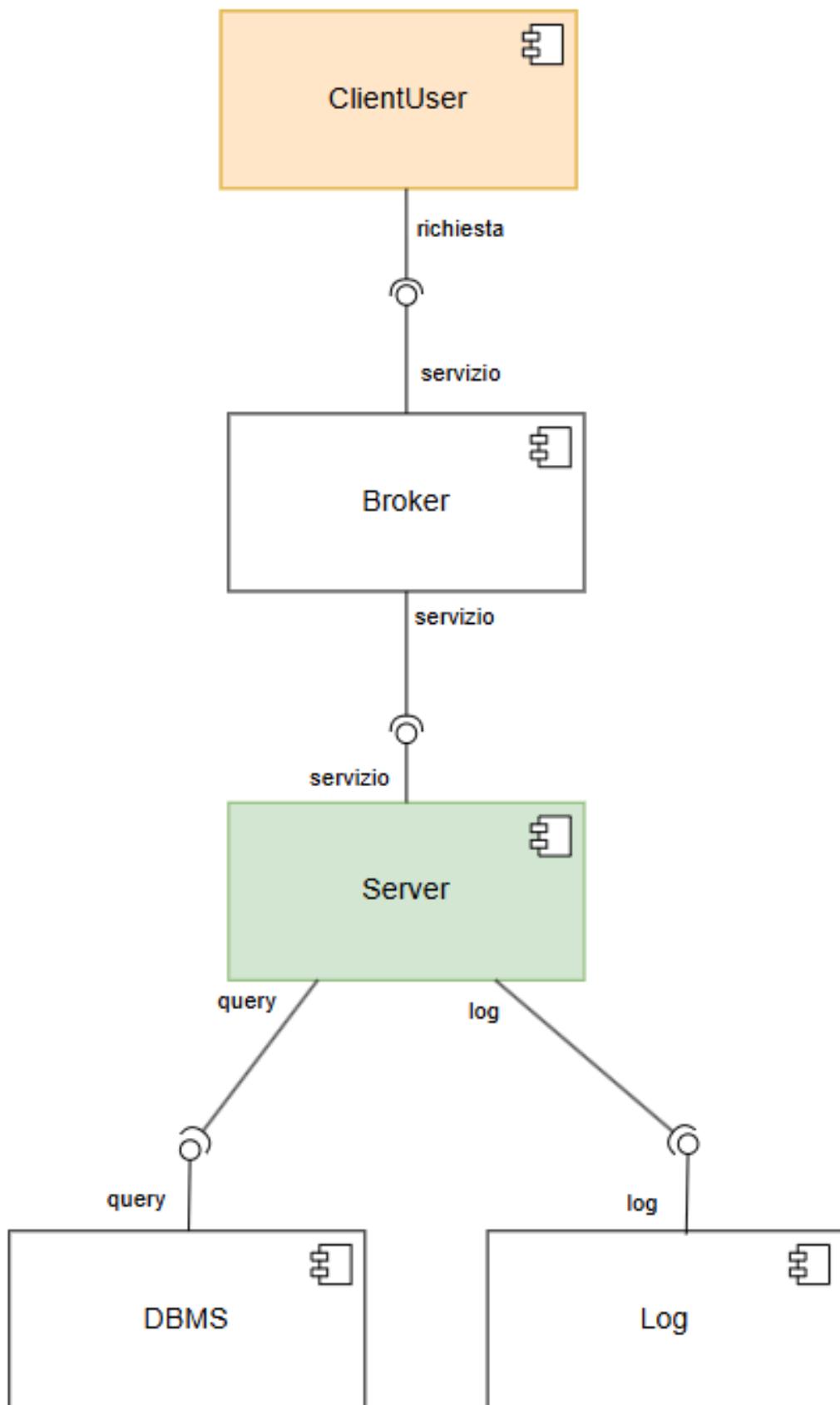
- file di testo per il salvataggio dei log
- sistema DBMS dedicato alla memorizzazione dei dati del sistema

L'interfacciamento con il DBMS sarà utilizzato applicando il ***pattern DAO***: saranno le classi DAO ad interfacciarsi direttamente con il DB tramite opportuni metodi. A seguito della delega della logica applicativa a queste ultime, le classi del Dominio gestiranno unicamente la logica applicativa.

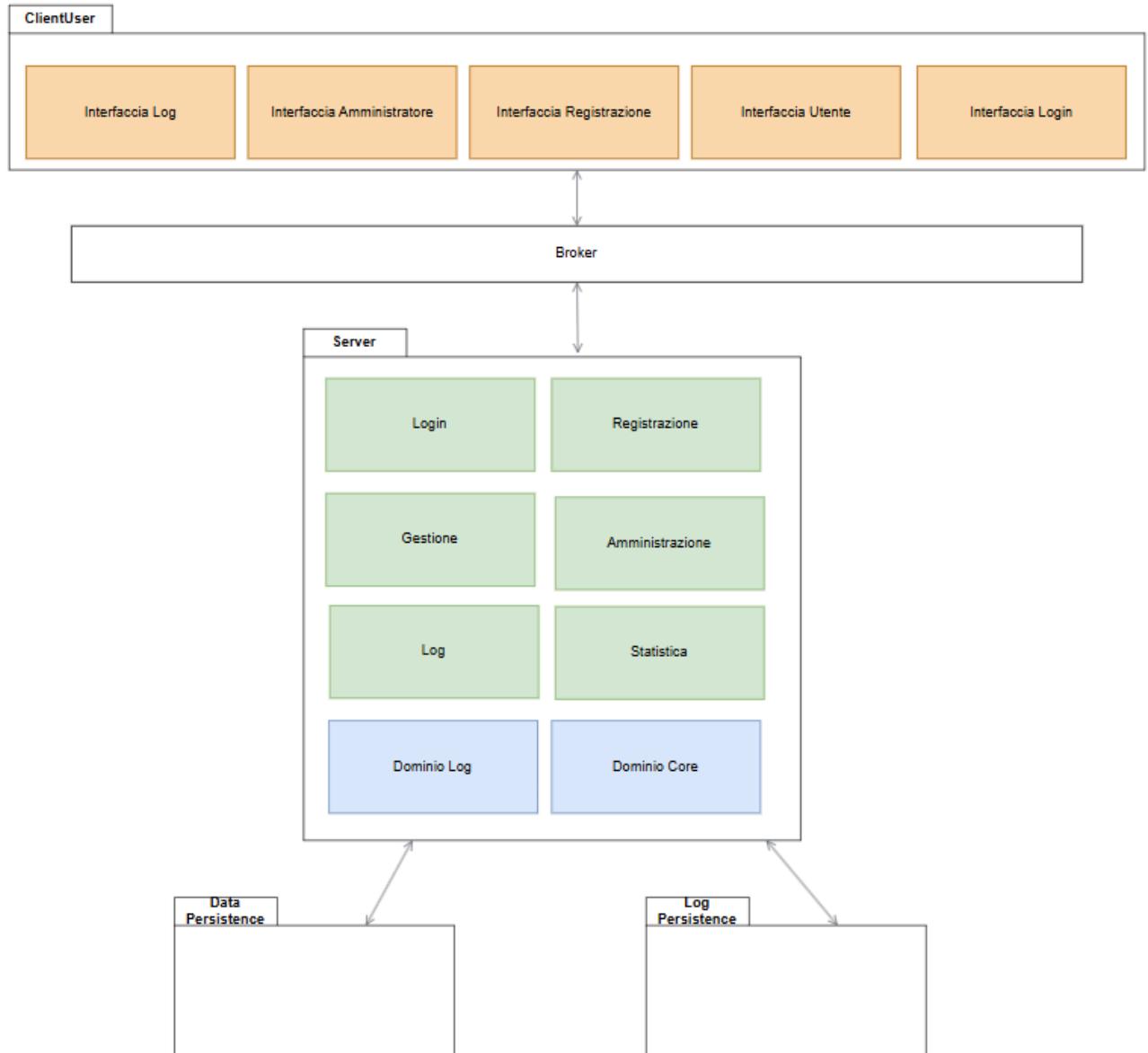
Verrà utilizzato il pattern ***Abstract Factory***, che consentirà all'applicazione di creare svariate tipologie di "DAOFactories", capaci di istanziare diversi oggetti DAO. Così facendo, gli oggetti con cui il Sistema interagirà saranno "ad hoc" per la tecnologia che si vuole utilizzare per il Database.

Infine, rifacendosi al pattern ***Proxy***, verrà creata la classe *CollezioneProxy*. Questa viene introdotta per migliorare i tempi di risposta del sistema, permettendo di non caricare le informazioni di tutte le Carte di tutte le Collezioni fino a quando non sia necessario, alleggerendo il carico computazionale.

## Architettura del Sistema: rappresentazione in package



## Architettura dei componenti



## Scelte tecnologiche

Il Sistema farà uso del web-server **Apache Tomcat**. Questo funge da Broker, fornendo gli strumenti per la gestione della sessione e del tempo di vita delle connessioni. Conseguentemente, non verrà successivamente mostrata la struttura interna del Broker.

# Progettazione di Dettaglio

## Struttura: Dominio

Giunti alla Progettazione, si è presentato un problema nella implementazione logica della classe Argomento ed i legami che questa ha con le classi FlashCard e Collezione.

In quanto l'Utente può aggiungere ad una Carta uno qualunque degli argomenti che questi ha creato, nel momento in cui questo accade, si devono recuperare e presentare all'Utente tutti gli Argomenti creati.

Attribuire alle Carte la possibilità di richiamare tutti gli Argomenti esistenti creerebbe attrito con l'idea iniziale di Carta come entità con una visione limitata, esistente unicamente come membro di una Collezione. Il massimo risultato ottenibile senza infrangere questa idea vedrebbe ogni Carta in grado di richiamare tutti gli Argomenti delle altre Carte della Collezione a cui appartiene.

Questa soluzione è, però, incompleta: l'Utente potrebbe voler aggiungere alla Carta un Argomento esistente ma finora non presente in quella Collezione, che quindi non verrebbe recuperato con questo metodo.

Si sono dunque individuate due alternative:

1. Inserire l'elenco degli Argomenti nella classe di Dominio Collezione. Ovvero:
  - a. Quando l'utente desidera attribuire un Argomento ad una Carta, questi lo inserisce prima tra gli argomenti contenuti nella Collezione (se non è già presente) e solo poi modifica la Carta.
  - b. Gli argomenti presentati all'Utente nel momento della modifica sono solo quelli che l'Utente ha aggiunto alla Collezione.
2. Inserire l'elenco degli Argomenti nella classe di Dominio Utente. Ovvero:
  - a. Quando l'Utente desidera attribuire un Argomento ad una Carta, questi modifica direttamente la Carta.
  - b. Gli argomenti presentati all'Utente nel momento della modifica sono tutti quelli che questi ha creato.

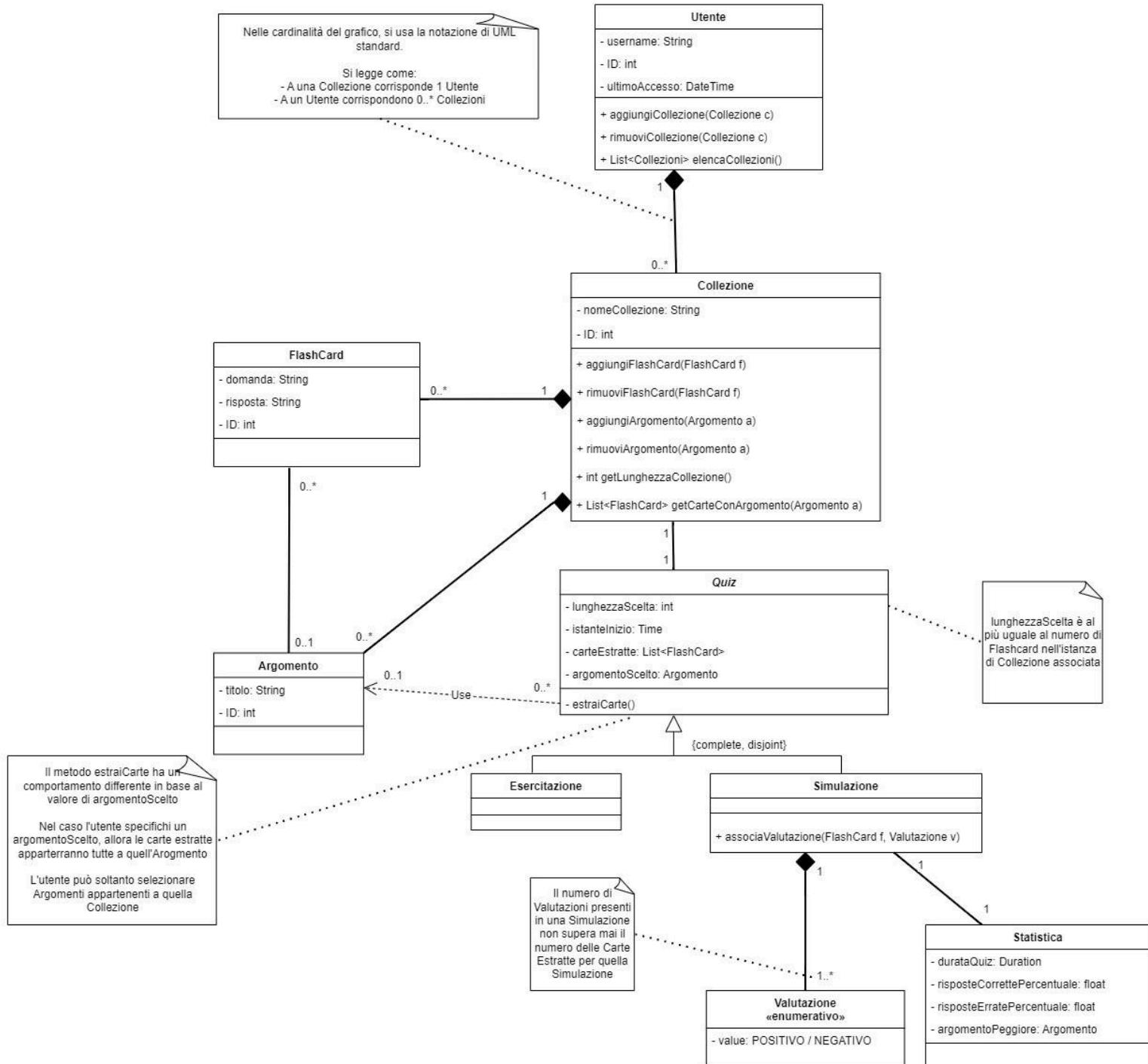
La seconda strada appare immediatamente la realizzazione più semplice. Legare gli Argomenti alla classe Utente introduce, però, un vincolo: **non possono esistere due Argomenti con lo stesso nome**.

Se supponiamo che l'Utente utilizzi nomi distinti come Argomenti, ad esempio dividendo le carte della Collezione “*Storia*” in “*Illuminismo*”, “*Rivoluzione Francese*” e “*Impero di Napoleone*” allora la soluzione 2. è perfettamente funzionale.

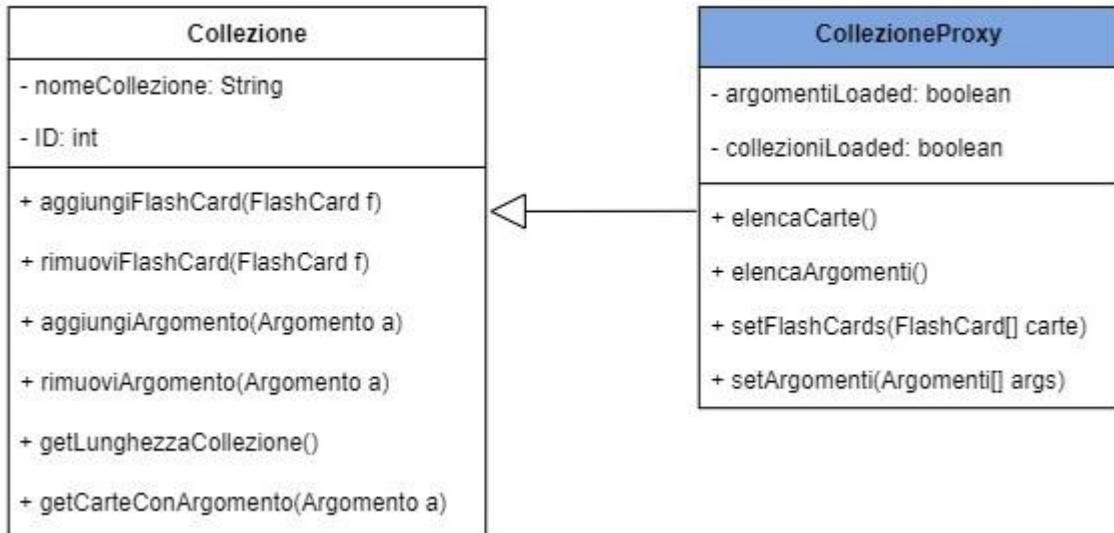
Di contro, niente impedisce all'Utente di dividere la Collezione presa ad esempio negli Argomenti “*Capitolo 1*”, “*Capitolo 2*” e “*Capitolo 3*”. Se così fosse, l'Utente non potrebbe creare l'Argomento “*Capitolo 1*” per la Collezione di “*Geografia*”, in quanto è già esistente.

Non potendo prevedere come l'utenza si approcci alla funzionalità degli Argomenti, si è deciso di intraprendere la prima strada per non rischiare di penalizzare parte dell'Utenza.

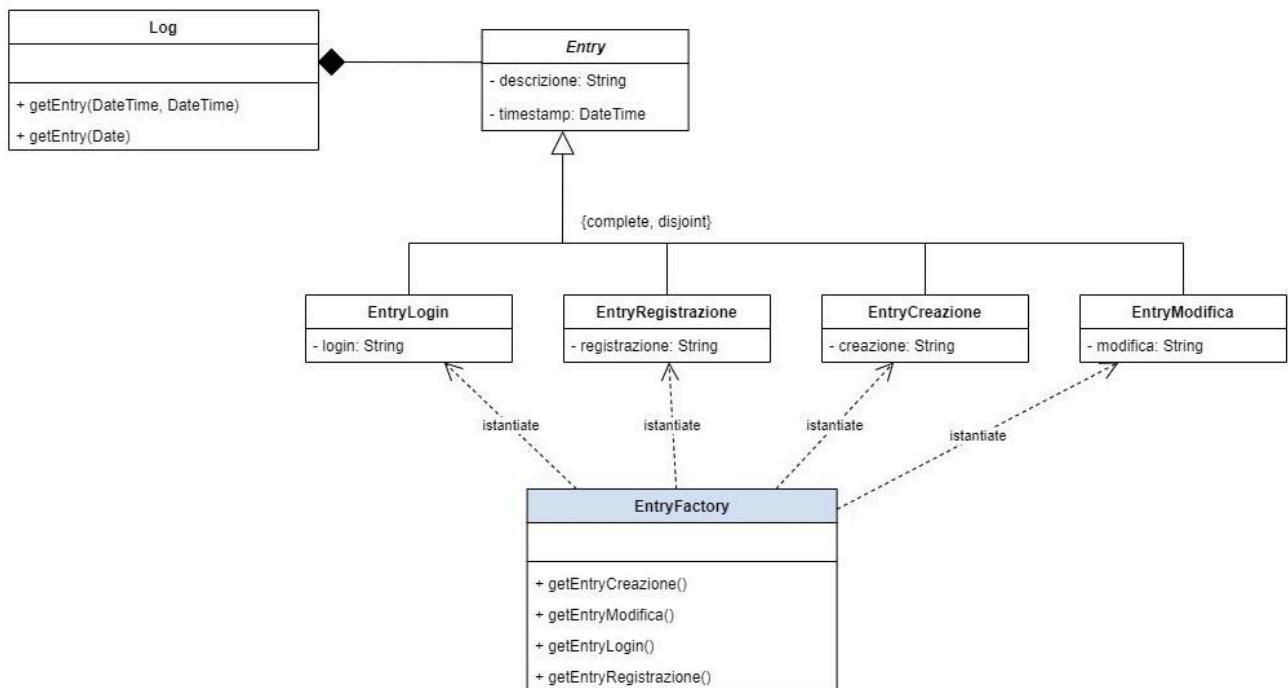
Di seguito viene quindi presentato il Modello di Dominio aggiornato con i metodi e gli attributi ritenuti necessari. Collezione non è più soltanto un elenco di FlashCard, ma anche un elenco di Argomenti.



Come annunciato nella ***Scelta dell'Architettura***, è stata introdotta la classe *CollezioneProxy*, che eredita dalla classe *Collezione*:



Per quanto riguarda le classi legate ai log, queste sono rimaste invariate da quanto visibile nel ***Modello di Dominio*** e pertanto non sono riportate. E' stata inoltre introdotta una classe che ha il compito di fornire i diversi tipi di Entry su richiesta, detta *EntryFactory*.



## Struttura: Interfacce

Opportune interfacce sono state introdotte come ulteriore livello di astrazione che renda i controller non vincolati dalle loro dipendenze. Così facendo, è stato possibile avvicinarsi al *The Dependency Inversion Principle*.

«interface»
I-ContenutoController
+ nuovaCarta(Collezione c, String d, String r, Argomento a)
+ rimuoviCarta(Collezione c, FlashCard f)
+ modificaCarta(FlashCard f)
+ nuovoArgomento(Collezione c, String titolo)
+ rimuoviArgomento(Collezione c, Argomento a)
+ modificaArgomento(Argomento a)
+ scaricaContenuto(int IDC)
+ print(Entry e)

«interface»
I-CollezioniController
+ nuovaCollezione(Utente u, String s)
+ rimuoviCollezione(Collezione c)
+ inizializzaHome(Utente u)
+ print(Entry e)

«interface»
I-EsercitazioneController
+ inizioEsercitazione(Collezione c, int lunghezza, Argomento a)
+ fineEsercitazione(Esercitazione e)

«interface»
I-SimulazioneController
+ inizioSimulazione(Collezione c, int lunghezza, Argomento a)
+ aggiungiValutazione(FlashCard f, int value)
+ fineSimulazione(Simulazione s)

«interface»
I-AmministrazioneController
+ List<Utente> elencoUtenti()
+ eliminaUtente(Utente u)

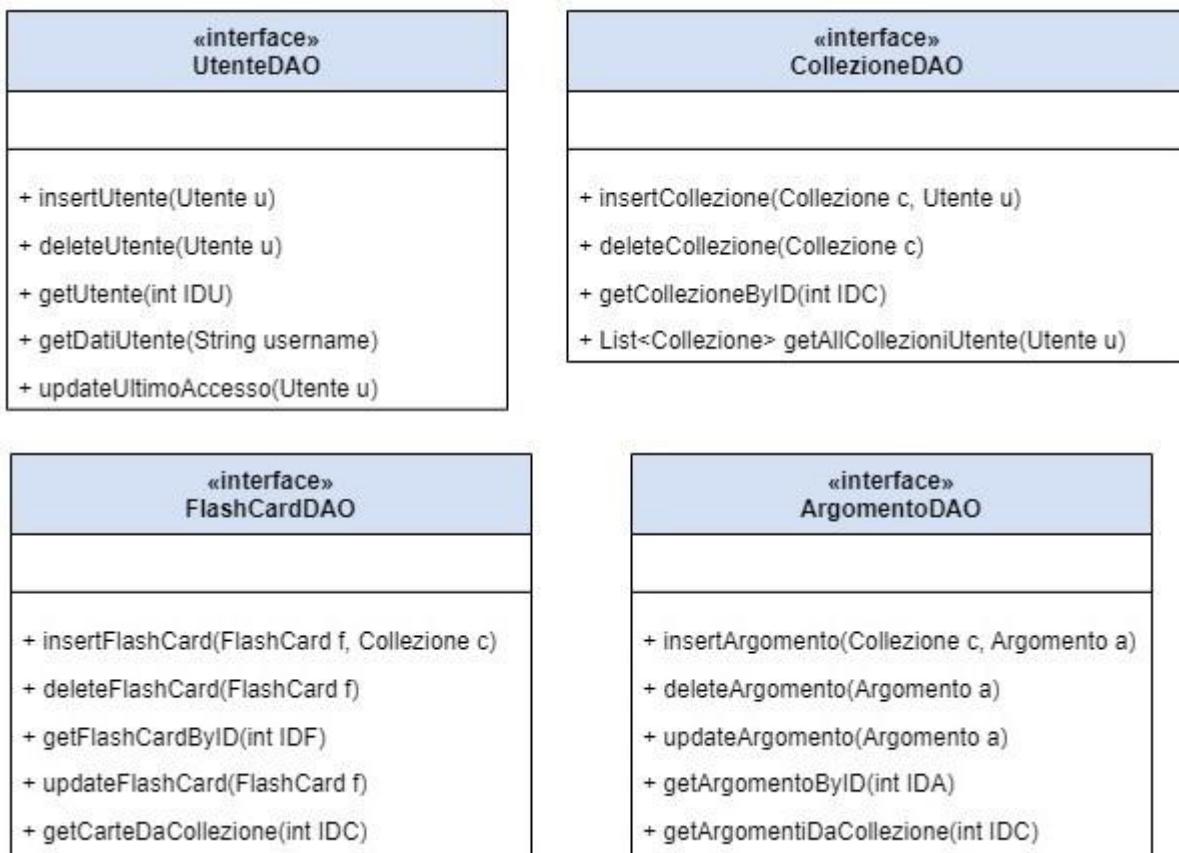
«interface»
I-LogController
+ Entry getEntry(DateTime dt1, DateTime dt2)
+ Entry getEntry(Date d)
+ Entry getEntryCreazione(Date d)
+ Entry getEntryModifica(Date d)
+ Entry getEntryRegistrazione(Date d)
+ Entry getEntryLogin(Date d)

«interface»
I-StatisticaController
+ concludi(Simulazione s)

«interface»
I-LoginController
+ verificaCredenziali(String username, String password)
+ print(Entry e)

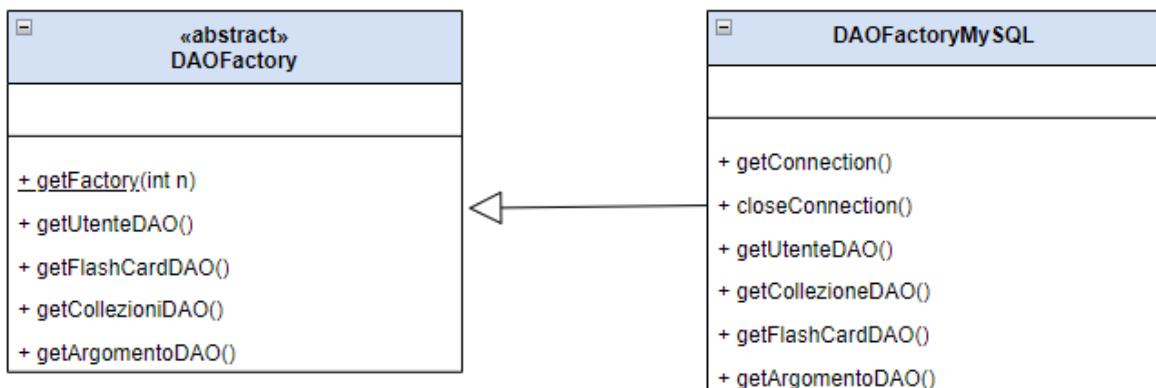
«interface»
I-RegistraController
+ registraCredenziali(String username, String password)
+ print(Entry e)

Inoltre, nella figura sottostante sono anche riportate le interfacce realizzate per l'implementazione del **pattern DAO** nella comunicazione con Database.



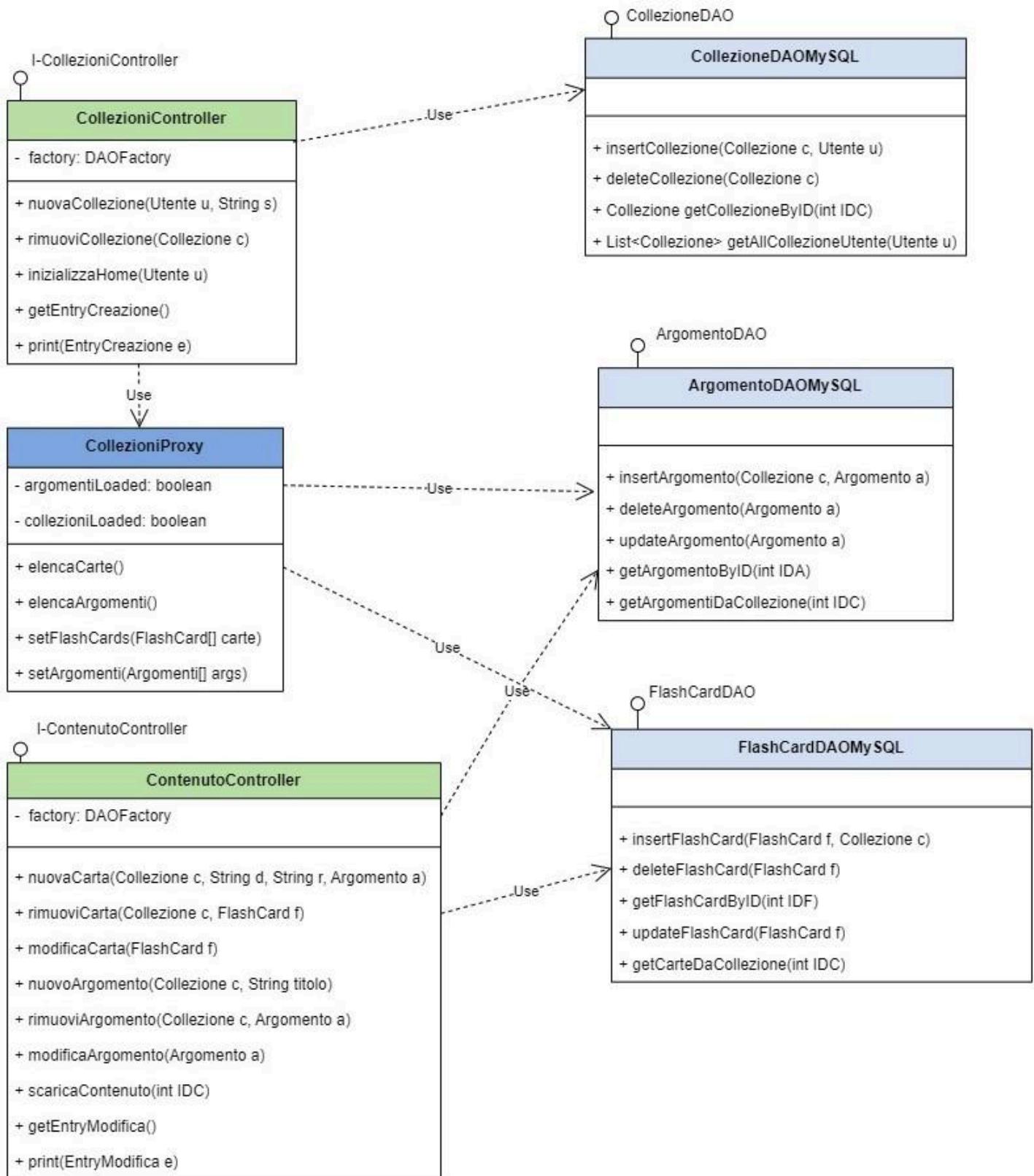
## Struttura: Controller

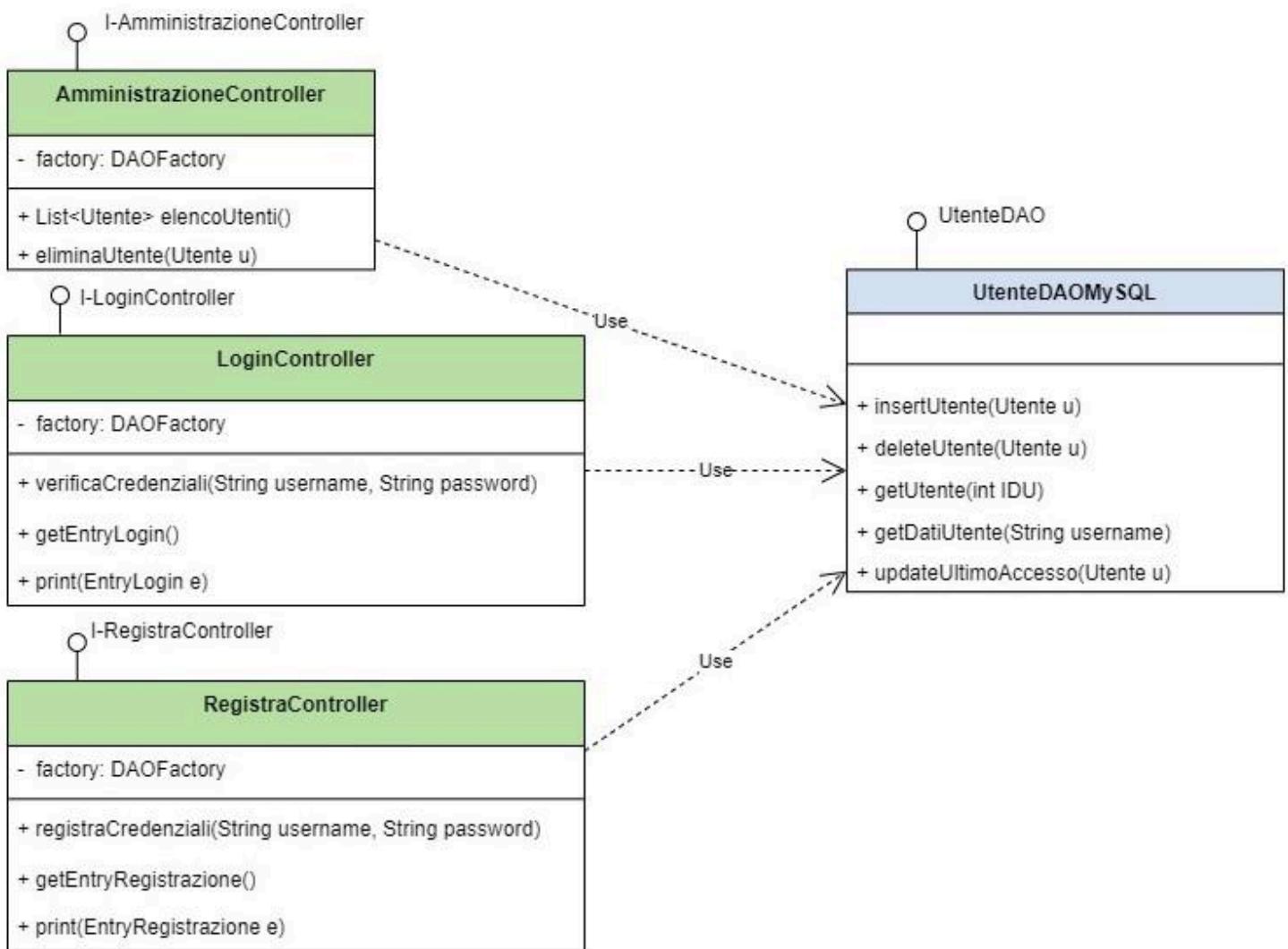
Come già anticipato, si ha una *DAOFactory* con il ruolo di consegnare le opportune istanze delle classi DAO ai diversi controller. In quanto nel nostro caso si è scelto di affidarsi ad una Base di Dati relazionale, questa viene concretizzata nella (quasi omonima) classe mostrata a fianco.



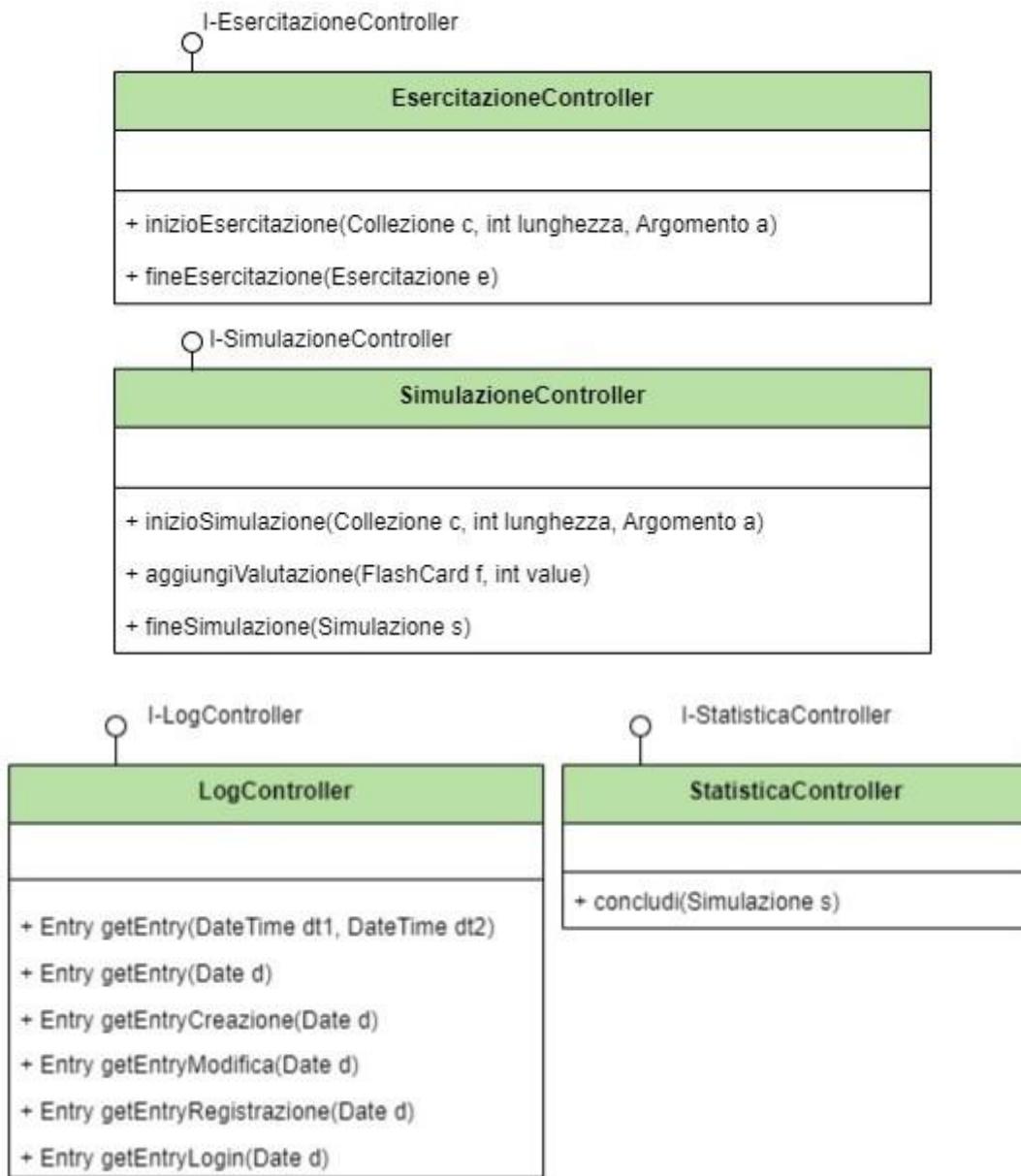
Se si volesse cambiare la tecnologia di gestione del DB, si dovrebbe soltanto creare una classe analoga a *DAOFactoryMySQL* lasciando la *DAOFactory* inalterata.

Successivamente, l'elenco di Controller che implementano le interfacce DAO presentate:





Infine, i restanti controller:

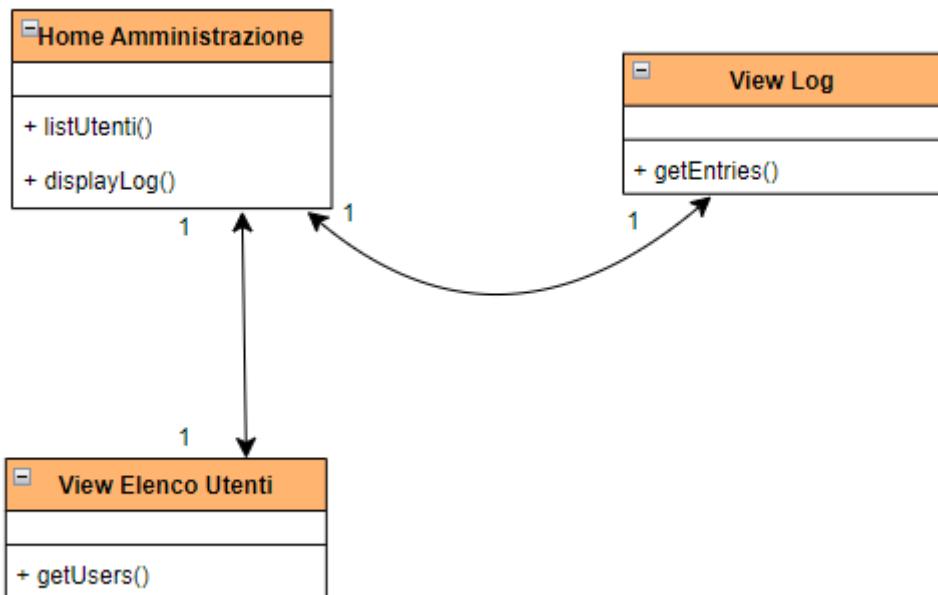


Si noti che il *GestioneController* appartenente al package *Gestione* mostrato in *Analisi* non è riportato. Si è infatti ritenuto opportuno non realizzarlo, in quanto i controller che ereditano da questo sono troppo eterogenei.

Per quanto detto in *Scelte tecnologiche*, non viene mostrata la struttura del Broker di Apache Tomcat.

## View ed Interfacce

### Interfaccia Amministratore e Log



Benvenuto  
Amministratore

NOME UTENTE      ULTIMO ACCESSO

Astolfo      15/02/2024

ELIMINA

Cadmo      24/01/2024

ELIMINA

LasagnaLover      15/02/2022

ELIMINA

[TORNA ALLA HOME](#)

Benvenuto  
Amministratore

ELENCO LOG

5-12-24 16:54 - L'utente PierClaudio ha effettuato il Login

5-12-24 16:22 - L'utente Astio ha creato la collezione Geografia

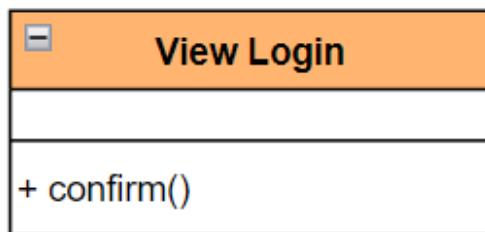
...

...

...

[TORNA ALLA HOME](#)

## Login e Registrazione



PAGINA DI LOGIN

Inserisci username

Inserisci password

Login

Non sei registrato?  
[Registrati](#)

PAGINA DI REGISTRAZIONE

Inserisci username

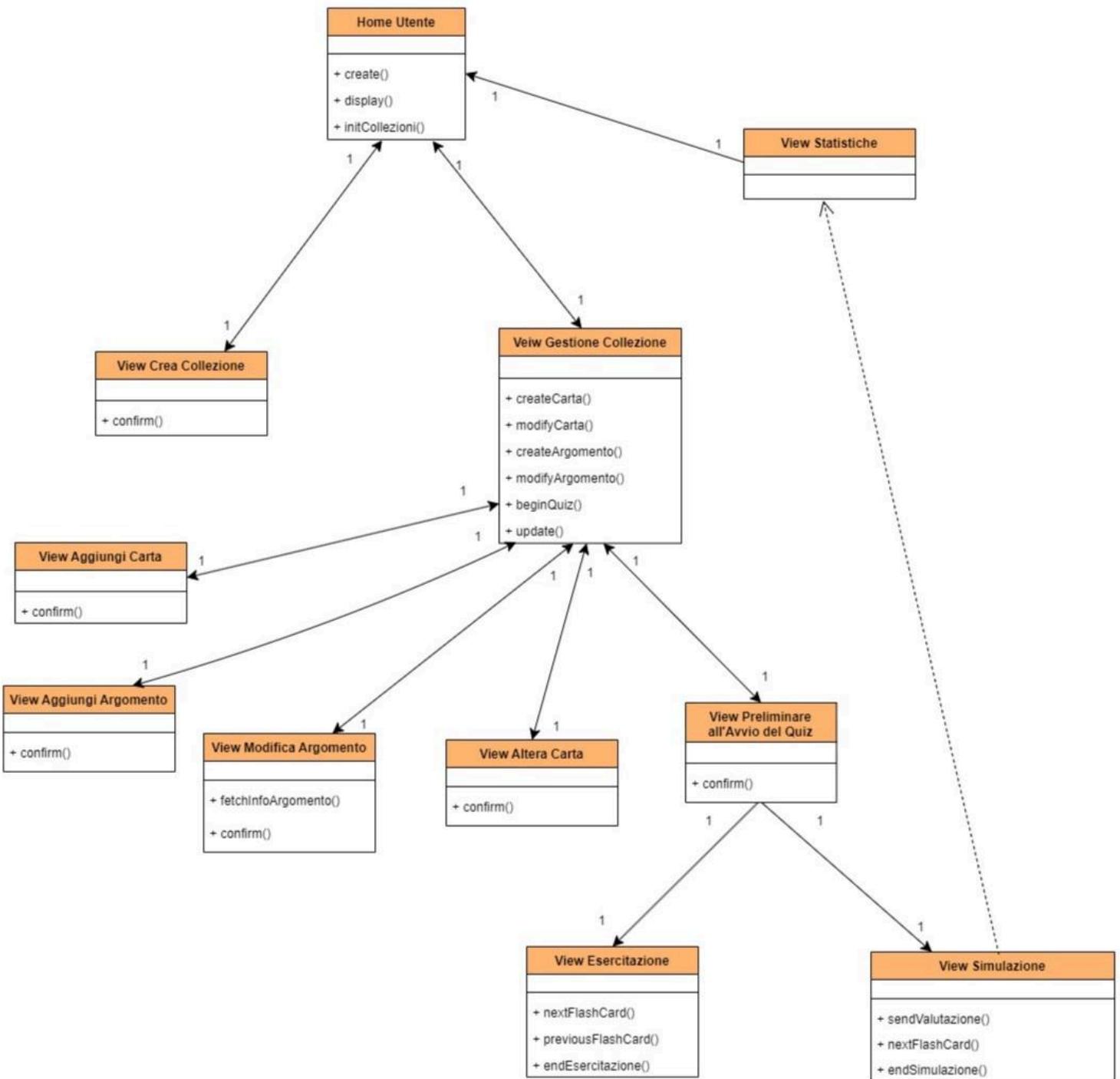
Inserisci una password

Conferma password

Registra account

Sei già registrato? [Accedi](#)

## Interfaccia Utente



Saluti Pablo!

[Logout](#)

Elenco di tutte le collezioni

Storia

ELIMINA

Matematica

ELIMINA

Geografia

ELIMINA

Crea nuova  
collezione

Saluti Pablo!

[Logout](#)

Creazione nuova collezione

Inserisci il nome  
della collezione

Conferma creazione

Saluti Pablo!

[Logout](#)

## Creazione nuovo argomento

Inserisci il nome  
dell'argomento

Conferma creazione  
argomento

## Modifica Flashcard

Modifica domanda

In che anno ebbe inizio la rivoluzione francese?

Modifica risposta

1789

Modifica argomento

--INSERISCI EVENTUALE ARGOMENTO--



Conferma modifiche

Indietro

## Crea FlashCard

Inserisci la domanda

Inserisci la risposta

Assegna eventuale  
argomento

--INSERISCI EVENTUALE ARGOMENTO--



Conferma

Indietro

Saluti nomeUtente

[Logout](#)

Elenco delle FlashCards di Geografia

Qual'è la capitale dell'Ucraina?

[ELIMINA](#)

Dove si trova il Kosovo?

[ELIMINA](#)

[NUOVA  
FLASHCARD](#)

[NUOVO  
ARGOMENTO](#)

[EFFETTUA QUIZ](#)

### Inserimento dati quiz

lunghezza Quiz



seleziona eventuale  
argomento



seleziona modalita

- Esercitazione
- Simulazione

**INIZIA QUIZ**

### SIMULAZIONE

Qual'è la formula della Trasformata di Fourier  
per segnali tempo-discreti?

Stai visualizzando la carta 12 di 35

Conosco la risposta

Non conosco la  
risposta

CARTA PRECEDENTE

CARTA SUCCESSIVA

## STATISTICHE QUIZ

TEMPO IMPIEGATO	35:17
ARGOMENTO PEGGIORE	Rivoluzione Americana
% POSITIVE	75
% NEGATIVE	25

TORNA ALLA  
HOME  
COLLEZIONE

## ESERCITAZIONE

Qual'è la formula della Trasformata di Fourier  
per segnali tempo-discreti?

Stai visualizzando la carta 12 di 35

Visualizza risposta

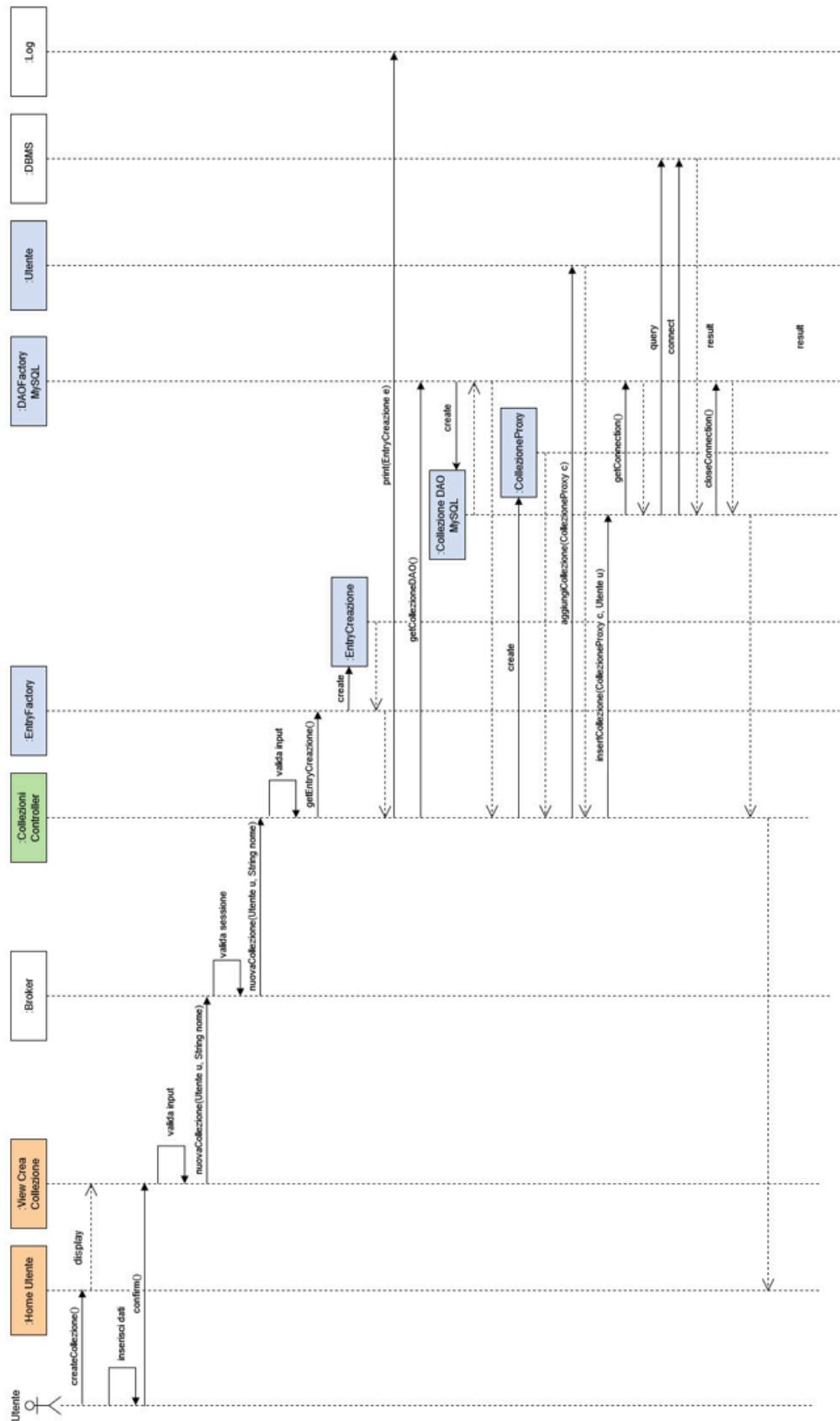
CARTA PRECENTE

Termina  
Esercitazione

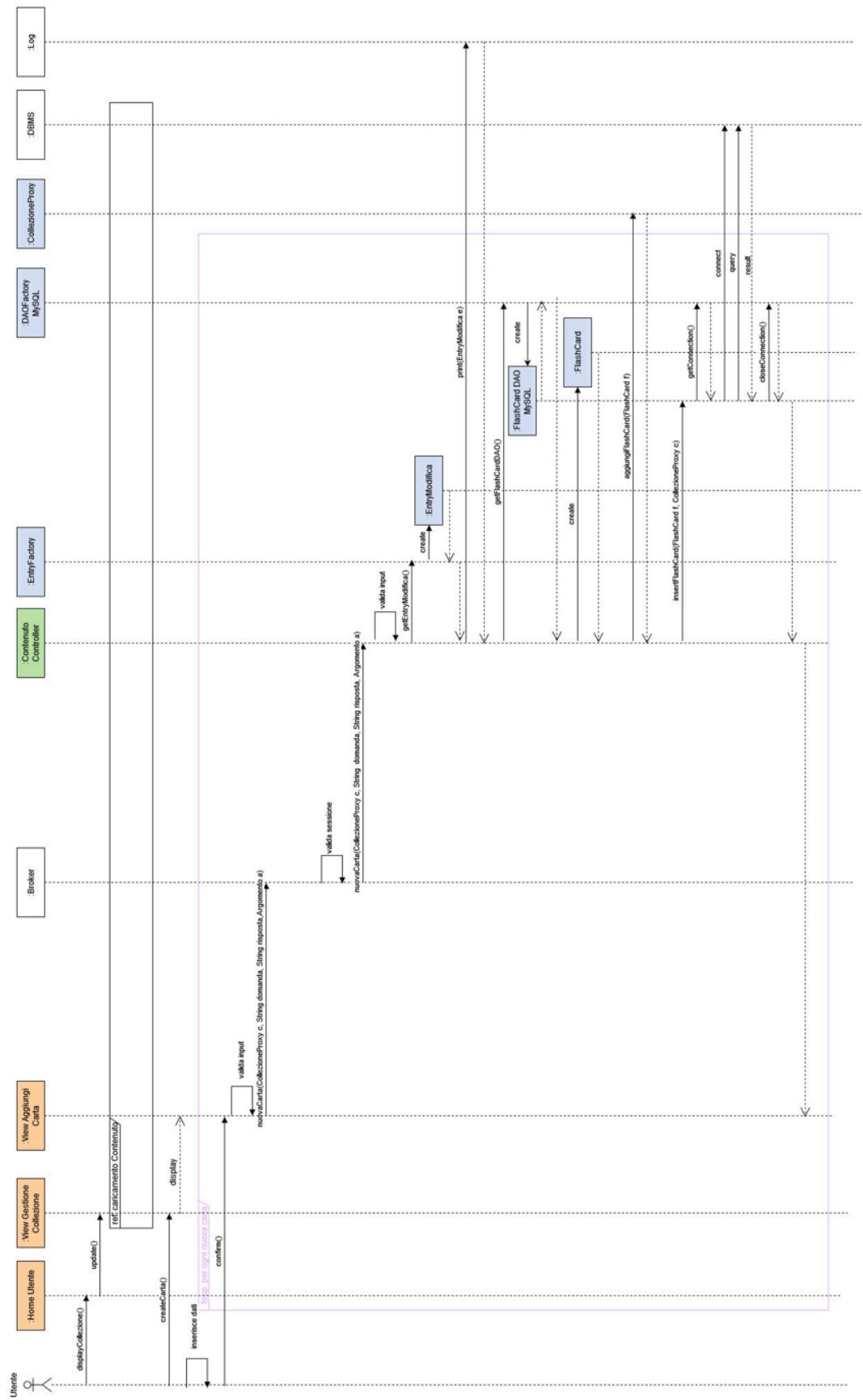
CARTA SUCCESSIVA

# Interazione

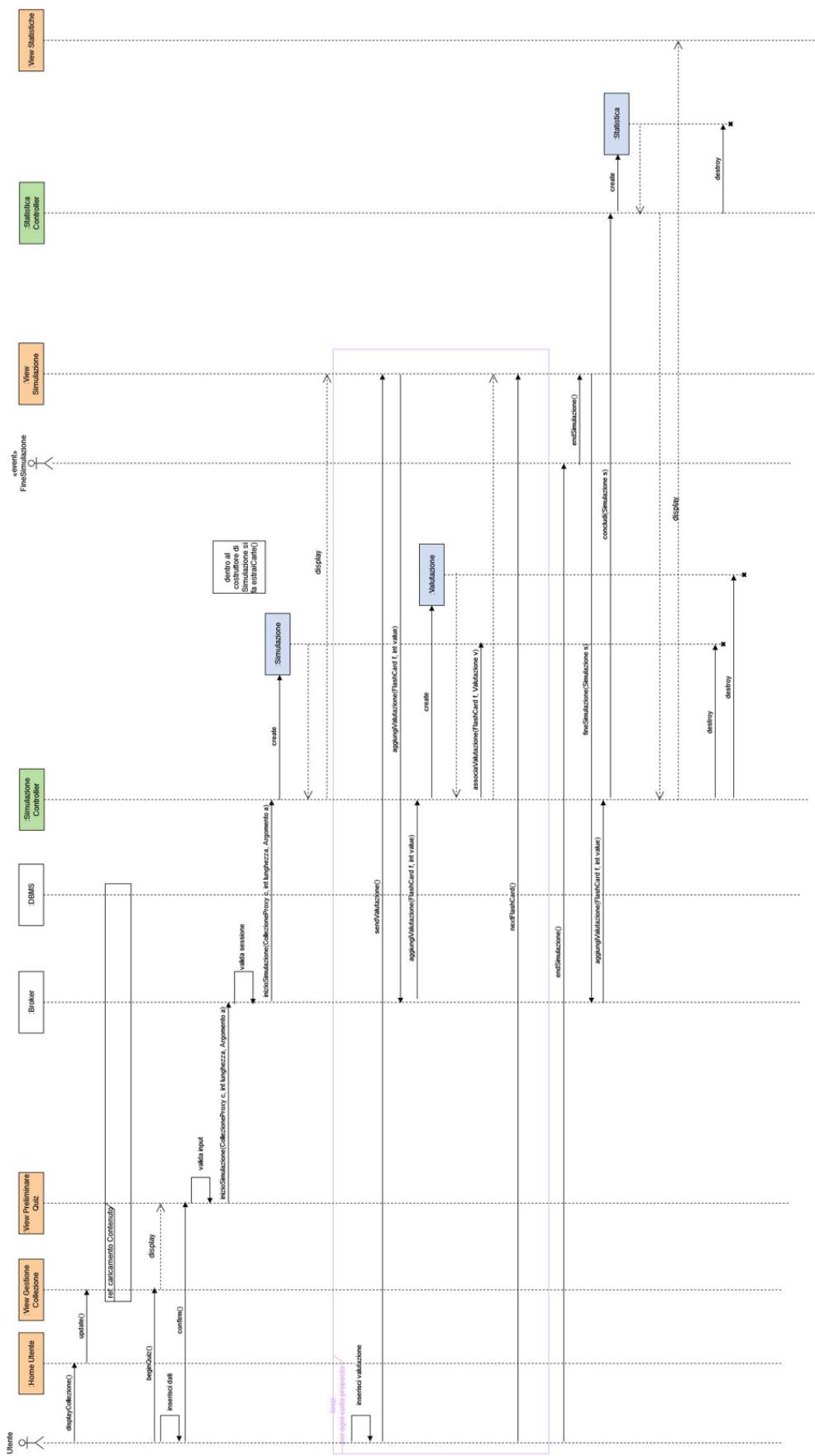
## Creazione Collezione



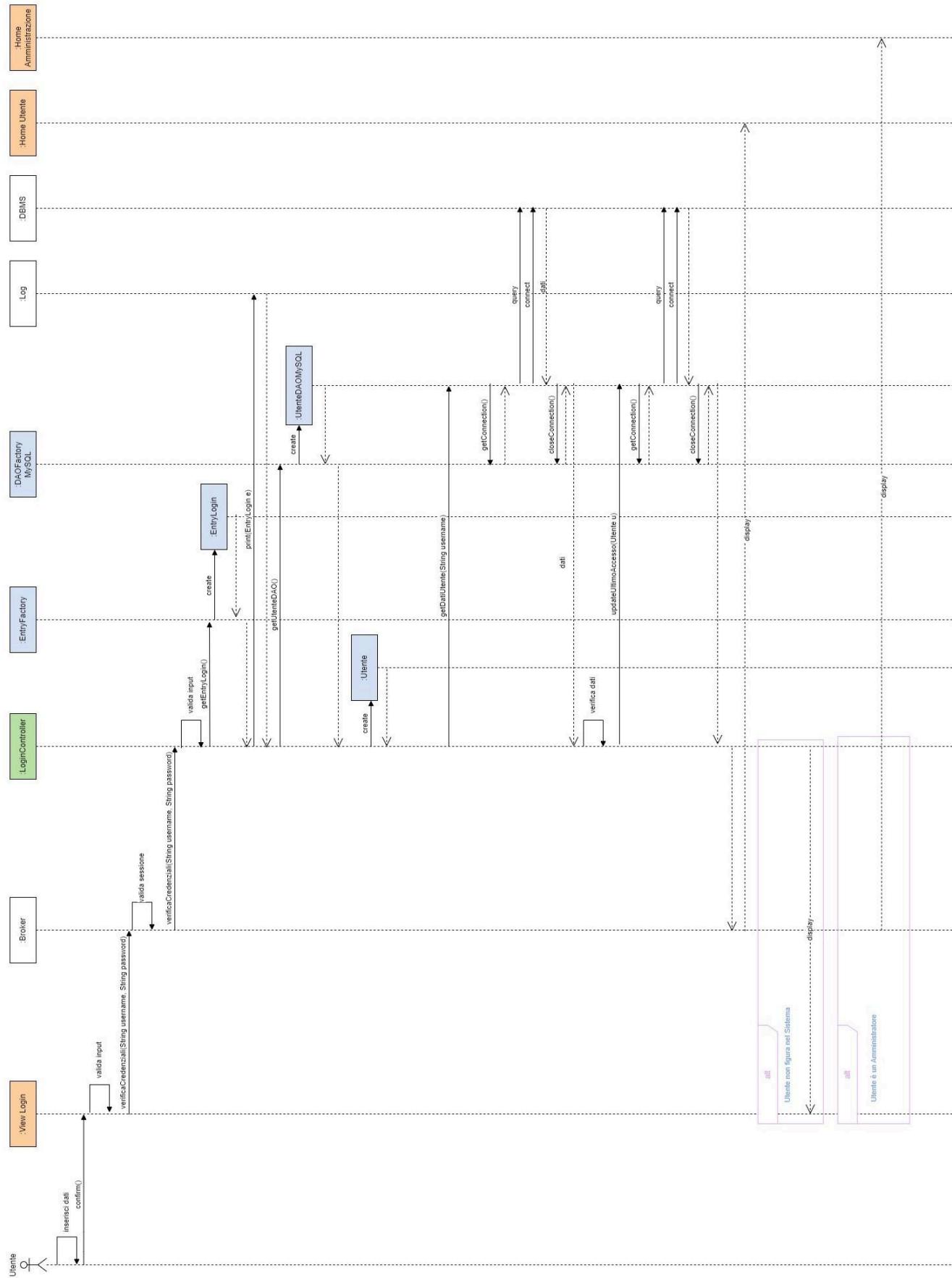
## Aggiunta di una Carta



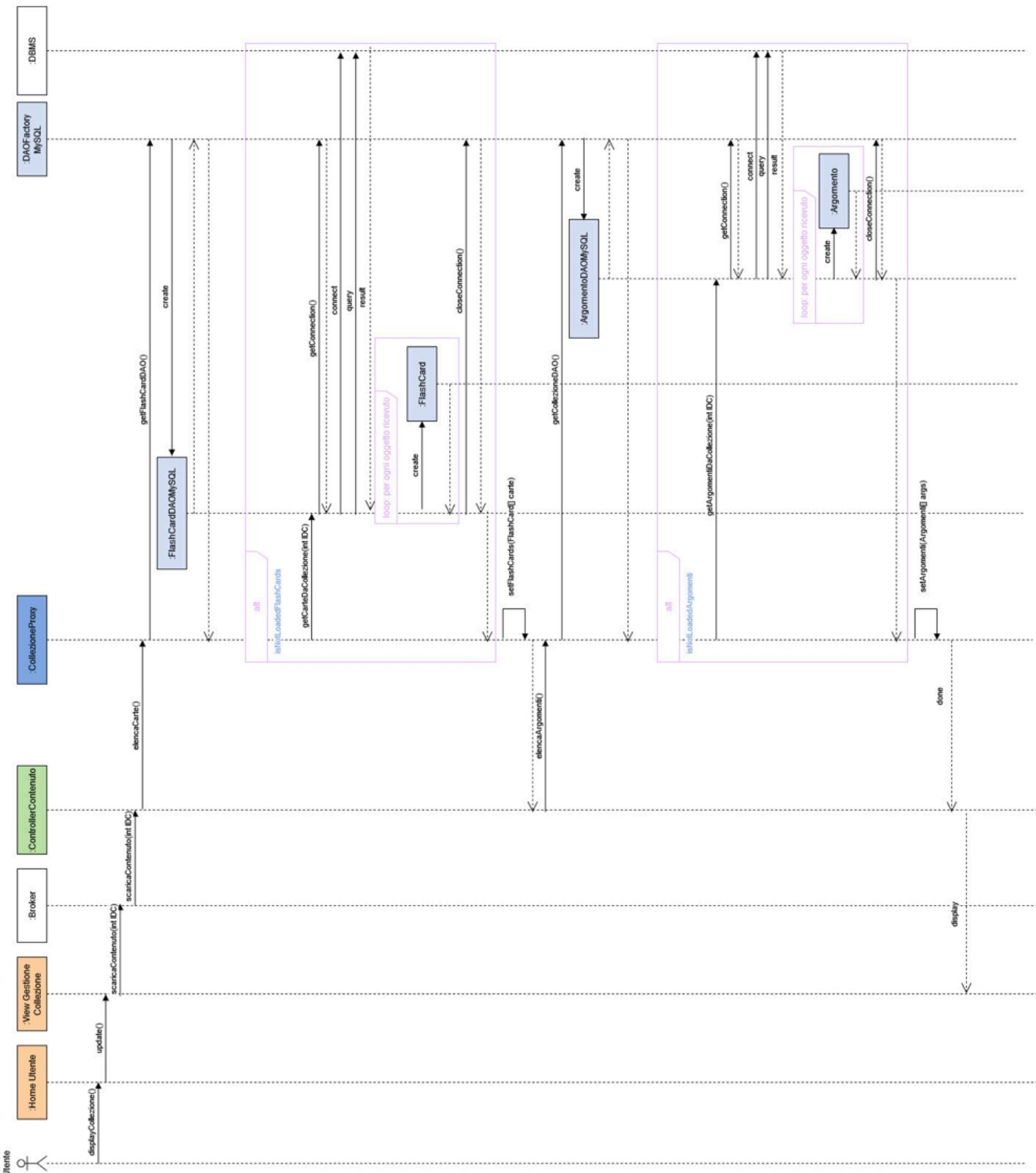
## Simulazione



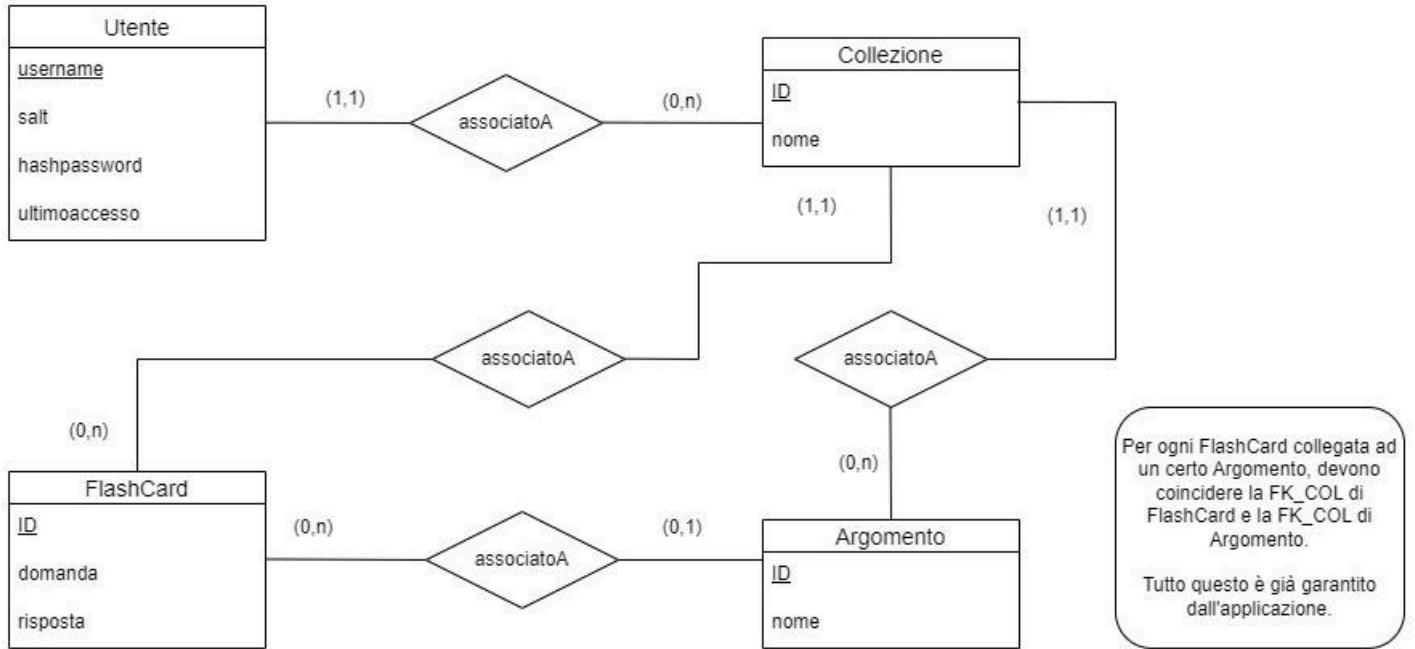
## Login



## Caricamento dei dati di una Collezione



# Progettazione della Persistenza



## Formato dei File di Log

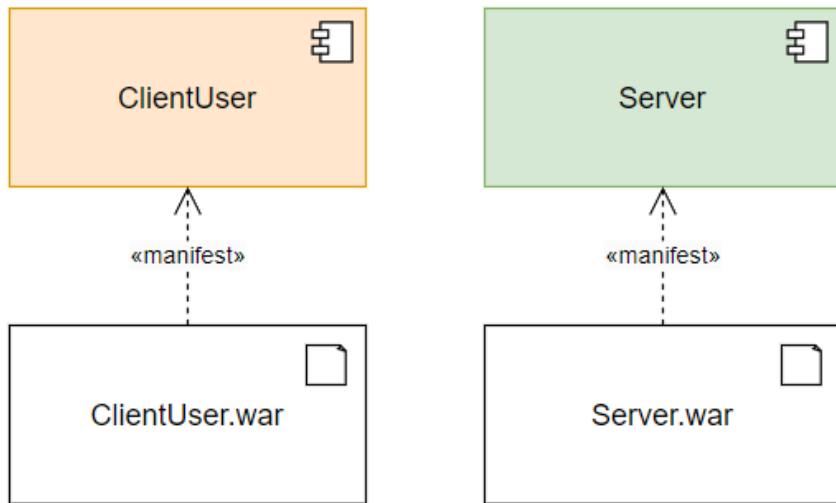
- Entry Log Creazione  
“[Timestamp] - L’utente [ Utente ] ha creato la collezione [ Collezione ]”
- Entry Log Modifica  
“[Timestamp] - L’utente [ Utente ] ha modificato la collezione [ Collezione ]”
- Entry Log Registrazione  
“[Timestamp] - Viene registrato utente: [ Utente ]”  
“[Timestamp] - Viene eliminato utente: [ Utente ]”
- Entry Log Login  
“[Timestamp] - L’utente [ Utente ] è stato autenticato”

## Progettazione di Collaudo

```
public class CollezioneTest {
    private Collezione collezione;
    private FlashCard flashCard;
    @Before
    public void setUp() {
        collezione = new Collezione(1, "Collezione1");
    }
    @Test
    public void testCostruttore() {
        Assert.assertEquals(1, collezione.getId());
        Assert.assertEquals("Collezione1", collezione.getName());
        Assert.assertEquals(0, collezione.getArgomenti().size());
        Assert.assertEquals(0, collezione.getFlashCards().size());
    }
    @Test
    public void testgetId() {
        Assert.assertEquals(1, collezione.getId());
    }
    @Test
    public void testGetName() {
        Assert.assertEquals("Collezione1", collezione.getName());
    }
    @Test
    public void testAddArgomento() {
        collezione.addArgomento(new Argomento(1, "Argomento1"));
        Assert.assertEquals(1, collezione.getArgomenti().size());
    }
    @Test
    public void testAddFlashCard() {
        collezione.addFlashCard(new FlashCard(1, "Domanda", "Risposta"));
        Assert.assertEquals(1, collezione.getFlashCards().size());
    }
    @Test
    public void testRemoveArgomento() {
        collezione.addArgomento(new Argomento(1, "Argomento1"));
        collezione.removeArgomento(1);
        Assert.assertEquals(0, collezione.getArgomenti().size());
    }
    @Test
    public void testRemoveFlashCard() {
        collezione.addFlashCard(new FlashCard(1, "Domanda", "Risposta"));
        collezione.removeFlashCard(1);
        Assert.assertEquals(0, collezione.getFlashCards().size());
    }
}
```

# Deployment

Artefatti



Deployment Type-Level

