



# Requirement Analysis Document

|                                 |          |
|---------------------------------|----------|
| <b>Introduzione</b>             | <b>2</b> |
| Dominio                         | 2        |
| Obiettivi e criteri di successo | 2        |
| <b>Sistema attuale</b>          | <b>3</b> |
| <b>Sistema proposto</b>         | <b>4</b> |
| Panoramica                      | 4        |
| Requisiti funzionali            | 5        |
| Requisiti non funzionali        | 8        |
| Modelli di sistema              | 10       |
| Scenari                         | 10       |
| Modello dei casi d'uso          | 16       |

## 1. Introduzione

Il committente ha l'obiettivo di dare la possibilità a qualsiasi individuo parlante lingua inglese e dotato di accesso al **World Wide Web** di poter esprimere apertamente e condividere le proprie opinioni e preferenze sul mondo musicale. Vuole quindi costruire una piattaforma web-based che permetta ai suoi utenti di poter recensire qualsiasi progetto musicale in maniera semplice, accessibile e amichevole.

### 1.1. Dominio

Esistono sul Web diverse piattaforme volte alla recensione di contenuto multimediale, ma raramente queste si specializzano per una particolare categoria di media: la musica.

Siti che si occupano di poter far recensire esclusivamente contenuto musicale esistono, ma dimostrano di essere sistemi datati e non user-friendly, dando l'impressione di essere utilizzati esclusivamente da un'utenza elitista molto limitata.

La piattaforma commissionata **soundrate** punta ad attrarre qualsiasi categoria di utenza, con preferenze e musicali quanto più disperate, e permettere ai suoi utenti di pubblicare recensioni in modo semplice e agevole.

Oltre a poter pubblicare recensioni di progetti musicali, gli utenti potranno prendere visione e assegnare un voto (che potrà essere *positivo* o *negativo*) alle recensioni altrui, e effettuare segnalazioni se il contenuto di queste è ritenuto inappropriato.

L'utente disporrà inoltre di un proprio profilo, dove saranno raccolte tutte le recensioni pubblicate da esso, e di una personale lista d'ascolto di cui potrà usufruire per tenere traccia dei progetti musicali che intende ascoltare e, eventualmente, recensire.

### 1.2. Obiettivi e criteri di successo

L'obiettivo del progetto è di agevolare il pubblico della scena musicale nell'espressione dei propri pareri e preferenze.

Il sistema proposto punta a costruire una piattaforma contenente quante più voci contenenti informazioni di progetti musicali pubblicati, che permetta ai suoi utenti di pubblicare recensioni sotto ognuna di queste voci. Il catalogo della piattaforma va quindi continuamente aggiornato, tenendo in considerazione il flusso costante di progetti musicali pubblicati ogni giorno; ciò tuttavia non deve inficiare sulle prestazioni della piattaforma, che all'utente deve apparire rapida ed elegante.

## 2. Sistema attuale

Il sistema corrente consiste di piattaforme esistenti da diverso tempo e mostranti un design arcaico e poco accessibile per un'utenza informale. Inoltre, non dispongono di funzionalità quali il poter esprimere un giudizio positivo o negativo sulle recensioni pubblicate, o fornire all'utenza una propria lista d'ascolto, o poter effettuare segnalazioni di contenuto inappropriato.

### 3. Sistema proposto

#### 3.1. Panoramica

Il sistema proposto è una piattaforma web-based, accessibile da qualsiasi utente dotato di un dispositivo fornito di un web browser.

La funzionalità principale che la piattaforma dovrà fornire consiste nella pubblicazione di recensioni per progetti musicali. Ogni utente registrato può pubblicare una singola recensione per ogni progetto individuale. Una recensione è composta da un voto intero che può variare da **1** a **10**, e da un corpo di testo esprimente l'effettivo contenuto della recensione.

Ogni utente registrato della piattaforma potrà assegnare una valutazione, positiva o negativa, ad ogni individuale recensione; ogni singola recensione è quindi dotata di un punteggio intero, corrispondente a **0** al momento della pubblicazione, che può essere positivo o negativo in base al numero di voti positivi e negativi accumulati. Le recensioni con un punteggio più alto vengono messe in evidenza.

Gli utenti autenticati potranno segnalare recensioni se ritenute inappropriate; le segnalazioni vengono gestite dagli utenti dotati di privilegi di moderazione.

Ogni utente registrato disporrà di una personale lista di ascolto, in e da cui potranno inserire e rimuovere qualsiasi progetto musicale presente nel catalogo della piattaforma, per tenerne comodamente traccia.

Gli utenti dotati di privilegi di moderazione potranno gestire le segnalazioni e rimuovere qualsiasi recensione, segnalata o meno. Gli utenti dotati di privilegi di amministrazione potranno assegnare e revocare i privilegi di moderazione ad altri utenti; questi inglobano inoltre i privilegi di moderazione.

Per le precedenti funzionalità descritte è sempre necessaria l'autenticazione dell'utente, e per alcune sono previste dei privilegi necessari ad usufruire della funzionalità; tuttavia, gli utenti non autenticati potranno comunque prendere visione di qualsiasi recensione pubblicata.

## 3.2. Requisiti funzionali

### 3.2.1. **RF\_UA\_1: Sign up**

**Descrizione:** il sistema deve permettere a qualsiasi utente di registrarsi alla piattaforma.

**Attori:** utente non autenticato.

### 3.2.2. **RF\_UA\_2: Log in**

**Descrizione:** il sistema deve permettere a qualsiasi utente registrato di autenticarsi alla piattaforma utilizzando le proprie credenziali.

**Attori:** utente non autenticato dotato di credenziali di accesso.

### 3.2.3. **RF\_UA\_3: Log out**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato alla piattaforma di de-autenticarsi invalidando la sessione.

**Attori:** utente autenticato.

### 3.2.4. **RF\_UA\_4: Recupero credenziali**

**Descrizione:** il sistema deve permettere a qualsiasi utente registrato alla piattaforma di recuperare le proprie credenziali d'accesso.

**Attori:** utente non autenticato dotato di un account registrato.

### 3.2.5. **RF\_UA\_5: Aggiornamento credenziali**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato di aggiornare le proprie credenziali dall'area utente.

**Attori:** utente autenticato.

### 3.2.6. **RF\_UA\_6: Visualizzazione profilo utente**

**Descrizione:** il sistema deve permettere a qualsiasi utente di poter visualizzare una pagina apposita per ogni utente contenente informazioni su di questo e le sue recensioni pubblicate.

**Attori:** utente non autenticato, utente autenticato.

**3.2.7. *RF\_R\_1: Pubblicazione recensione***

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato di pubblicare una recensione sotto la voce di qualsiasi progetto musicale.

**Attori:** utente autenticato.

**3.2.8. *RF\_R\_2: Aggiornamento recensione***

**Descrizione:** il sistema deve permettere a qualsiasi utente che ha pubblicato una recensione di poterne aggiornare il contenuto e il voto in qualsiasi momento.

**Attori:** utente autenticato.

**3.2.9. *RF\_R\_3: Rimozione recensione***

**Descrizione:** il sistema deve permettere a qualsiasi utente che ha pubblicato una recensione di poterne effettuare la rimozione in qualsiasi momento.

**Attori:** utente autenticato.

**3.2.10. *RF\_R\_4: Votazione recensione***

**Descrizione:** il sistema deve permettere a qualsiasi autenticato utente di poter assegnare, revocare e aggiornare una valutazione positiva o negativa a qualsiasi recensione pubblicata sulla piattaforma in qualsiasi momento.

**Attori:** utente autenticato.

**3.2.11. *RF\_R\_5: Visualizzazione recensione***

**Descrizione:** il sistema deve permettere a qualsiasi utente di poter visualizzare qualsiasi recensione pubblicata sul sito.

**Attori:** utente non autenticato, utente autenticato.

**3.2.12. *RF\_R\_6: Segnalazione recensione***

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato di poter segnalare una recensione se ne viene ritenuto inappropriato il contenuto .

**Attori:** utente autenticato.

**3.2.13. RF\_C\_1: Ricerca artisti e progetti musicali**

**Descrizione:** il sistema deve permettere a qualsiasi utente di poter effettuare ricerche che dovranno restituire un insieme di artisti e un insieme di progetti musicali corrispondenti alla ricerca effettuata.

**Attori:** utente non autenticato, utente autenticato.

**3.2.14. RF\_C\_2: Visualizzazione progetto musicale**

**Descrizione:** il sistema deve permettere a qualsiasi utente di poter visualizzare individualmente qualsiasi progetto musicale presente nel catalogo della piattaforma e le informazioni relativo a questo.

**Attori:** utente non autenticato, utente autenticato.

**3.2.15. RF\_C\_3: Visualizzazione artista**

**Descrizione:** il sistema deve permettere a qualsiasi utente di poter visualizzare individualmente qualsiasi artista registrato nel catalogo della piattaforma e le informazioni relative a questo.

**Attori:** utente non autenticato, utente autenticato.

**3.2.16. RF\_C\_4: Inserimento progetto in lista d'ascolto**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato di poter inserire qualsiasi progetto musicale nella propria lista d'ascolto.

**Attori:** utente autenticato.

**3.2.17. RF\_C\_5: Rimozione progetto da lista d'ascolto**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato di poter rimuovere qualsiasi progetto musicale dalla propria lista d'ascolto.

**Attori:** utente autenticato.

**3.2.18. RF\_C\_6: Visualizzazione lista d'ascolto**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato di visualizzare la propria lista d'ascolto in qualsiasi momento.

**Attori:** utente autenticato.



**3.2.19.    [RF\\_A\\_1: Aggiornamento privilegi utenti](#)**

**Descrizione:** il sistema deve permettere a qualsiasi utente dotato di privilegi di amministrazione di assegnare, aggiornare e revocare i privilegi associati agli account di altri utenti.

**Attori:** utente autenticato dotato di privilegi di amministrazione.

**3.2.20.    [RF\\_A\\_2: Rimozione recensione altrui](#)**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato dotato di privilegi di moderazione di rimuovere qualsiasi recensione in qualsiasi momento.

**Attori:** utente autenticato dotato di privilegi di moderazione.

**3.2.21.    [RF\\_A\\_3: Visualizzazione segnalazioni](#)**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato dotato di privilegi di moderazione di visualizzare le segnalazioni di recensioni registrate sul sistema.

**Attori:** utente autenticato dotato di privilegi di moderazione.

**3.2.22.    [RF\\_A\\_4: Rimozione segnalazione](#)**

**Descrizione:** il sistema deve permettere a qualsiasi utente autenticato dotato di privilegi di moderazione di rimuovere le segnalazione registrate per qualsiasi recensione.

**Attori:** utente autenticato dotato di privilegi di moderazione.

### 3.3. Requisiti non funzionali

#### 3.3.1. Usabilità

Il sistema dovrà essere intuitivo nell'utilizzo e presentare un'interfaccia semplice ed attraente che non risulti sopraffacente all'utente; anche le categorie di utenza meno esperte dovranno essere in grado di usufruire del sistema. Il sito dovrà essere in lingua inglese.

L'interfaccia dovrà guidare l'utente in maniera intuitiva e richiedere da questo un numero di passi possibili particolarmente ridotto per portare a termine qualsiasi azione.

#### 3.3.2. Affidabilità

Il sistema richiederà agli utenti il collegamento di un indirizzo e-mail alla registrazione. Le password degli utenti dovranno rispettare dei criteri di sicurezza di una certa efficacia (lunghezza tra gli 8 e i 72 caratteri alfanumerici e contenente almeno 2 cifre). Le password degli utenti saranno memorizzate su base di dati utilizzando la funzione di password hashing irreversibile *bcrypt*. Il sistema dovrà comunicare con i suoi client tramite protocollo di trasporto sicuro HTTPS.

Il sistema memorizzerà le informazioni sensibili degli utenti esclusivamente se necessario; queste non dovranno essere rese pubbliche o accedute in maniera non autorizzata sotto alcuna circostanza. Il sistema dovrà impedire ad utenti non autorizzati di accedere alle funzionalità il cui accesso sarà riservato esclusivamente agli utenti dotati di determinati privilegi.

Le funzionalità del sistema dovranno essere testate e funzionanti prima della consegna dello stesso; le funzionalità interne implementate dal sistema dovranno inoltre essere idempotenti, e di conseguenza prevedibili. Il sistema dovrà impedire l'accesso alle funzionalità richiedenti autenticazione se ne viene richiesto l'utilizzo da utenti che non hanno effettuato l'accesso alla piattaforma utilizzando delle credenziali valide. Il sistema dovrà permettere agli utenti che hanno smarrito le proprie credenziali di poterne effettuare il recupero in qualsiasi momento.

Il sistema dovrà essere in grado di rispondere alle richieste degli utenti anche sotto elevato carico, e eventualmente mostrare messaggi di errore per segnalare all'utente di ritentare la richiesta in un successivo momento dove possibile.

I periodi di inattività a seguito di manutenzione o failure del sistema dovranno essere quanto più ristretti.

#### 3.3.3. Prestazioni

Il sistema dovrà avere tempi di risposta ragionevolmente contenuti, entro il secondo per la maggior parte dei tipi di richiesta. Dovrà supportare delle ottimizzazioni per essere quanto più reattivo anche sotto elevato carico, implementando una propria cache interna, e scalare quando necessario.

### 3.3.4. Supportabilità

Il sistema dovrà agevolare eventuali sviluppi futuri ed essere flessibile a eventuali modifiche dovute a correzioni o evoluzioni delle tecnologie utilizzate. Ci si riserva il diritto di continuare a mantenere il sistema dopo la consegna al committente per un periodo di tempo indeterminato.

### 3.3.5. Implementazione

Le tecnologie utilizzate saranno, per il front-end:

- pagine *HTML5* con elaborazione *JavaServer Pages (JSP)* usando *Java Standard Tag Library (JSTL)* e *Expression Language (EL)*;
- *CSS3* per la presentazione delle pagine;
- *ECMAScript 6* per le interazioni asincrone con il back-end ed altre azioni dell'utente;
- lo UI toolkit *Semantic UI* per uno sviluppo rapido di pagine web responsive.

Mentre per il back-end saranno usate:

- la piattaforma *Java EE 8* con implementazione *Apache Tomee 8*;
- una base di dati relazionale con DBMS *MariaDB*;

La codifica utilizzata per la stesura del sistema sarà *UTF-8*.

### 3.3.6. Interfaccia

Il sistema dovrà interfacciarsi all'API pubblica del servizio di streaming *Deezer* per la gestione automatizzata del catalogo musicale della piattaforma. Il protocollo di comunicazione utilizzato sarà *HTTPS*.

### 3.3.7. Consegna

Il sistema funzionante e testato verrà consegnato e installato presso il committente al termine del suo processo di sviluppo. I deliverable costituenti la documentazione saranno consegnati periodicamente, rispettando le scadenze imposte dal committente.

### 3.3.8. Legali

Il sistema non usufruirà componenti richiedenti una licenza a pagamento.

Il sistema sarà sviluppato in maniera da garantire il rispetto delle norme vigenti sulla privacy, facendo riferimento al Regolamento (UE) n. 2016/679 del Parlamento europeo (GDPR), relativa alla tutela delle persone e di altri soggetti rispetto al trattamento dei dati personali.

### 3.4. Modelli di sistema

#### 3.4.1. Scenari

##### 3.4.1.1. *SC-UA\_1 Sign up*

Questo scenario descrive come un utente si registra al sistema.

###### Istanze di attori partecipanti

- Samuele (*utente non autenticato*).

###### Flusso di eventi

1. Samuele vuole registrare un account sulla piattaforma per cominciare a pubblicare recensioni. Samuele invia al sistema il proprio username, password e indirizzo e-mail richiedendo la registrazione di un account utente.
2. Il sistema riceve i dati forniti e, dopo aver controllato che Samuele non sia autenticato, aver validato i dati e infine controllato che non ci siano conflitti di username o indirizzo e-mail con altri utenti, registra l'account di Samuele.
3. Il sistema autentica Samuele alla piattaforma.

##### 3.4.1.2. *SC-UA\_2 Log in*

Questo scenario descrive come un utente si autentica al sistema.

###### Istanze di attori partecipanti

- Alessia (*utente non autenticato*).

###### Flusso di eventi

1. Alessia vuole recensire un album pubblicato recentemente, quindi decidere di effettuare l'accesso alla piattaforma con il proprio account. Alessia invia al sistema le proprie credenziali richiedendo l'autenticazione.
2. Il sistema riceve le credenziali e, dopo aver controllato che Alessandra non sia autenticata e che queste appartengano ad un account utente registrato, autentica Alessia alla piattaforma.

#### 3.4.1.3. **SC\_UA\_3 Log out**

Questo scenario descrive come un utente si de-autentica dal sistema.

##### Istanze di attori partecipanti

- Gianni (*utente autenticato*).

##### Flusso di eventi

1. Gianni ha appena pubblicato una recensione e vuole effettuare il log out dalla piattaforma, quindi richiede al sistema di de-autenticarsi.
2. Il sistema controlla che Gianni sia autenticato e ne invalida la sessione de-autenticandolo dalla piattaforma.

#### 3.4.1.4. **SC\_UA\_4 Recupero credenziali**

Questo scenario descrive come un utente ripristina l'accesso al proprio account.

##### Istanze di attori partecipanti

- Alfredo (*utente non autenticato*).

##### Flusso di eventi

1. Alfredo si accorge di aver dimenticato la propria password per accedere alla piattaforma, quindi decide di ripristinare l'accesso al proprio account. Alfredo invia al sistema il proprio indirizzo e-mail associato al proprio account utente richiedendone il recupero delle credenziali.
2. Il sistema controlla che Alfredo non sia autenticato e che l'indirizzo e-mail ricevuto appartenga all'account di uno dei suoi utenti, quindi invia una mail contenente un collegamento per il recupero della password all'indirizzo fornito da Alfredo.
3. Alfredo accede alla propria casella di posta elettronica e apre il collegamento fornito nell'e-mail inviata dal sistema.
4. Il sistema controlla nuovamente che Alfredo non sia autenticato e che il collegamento non sia invalido o scaduto, quindi richiede ad Alfredo di fornire una nuova password.
5. Alfredo invia la nuova password al sistema.
6. Il sistema controlla nuovamente che Alfredo non sia autenticato e che il collegamento non sia invalido o scaduto, quindi convalida la nuova password e aggiorna la password dell'account utente di Alfredo con la nuova password.

#### 3.4.1.5. *SC\_UA\_5* Aggiornamento credenziali

Questo scenario descrive come un utente aggiorna la password utilizzata per accedere al suo account utente.

##### Istanze di attori partecipanti

- Daniele (*utente autenticato*).

##### Flusso di eventi

1. Daniele decide di aggiornare la password associata al proprio account utente perché ne vorrebbe utilizzare una più sicura. Daniele invia al sistema la nuova password accompagnata dalla sua password attuale richiedendone l'aggiornamento.
2. Il sistema controlla che l'utente sia autenticato e che la password attuale ricevuta corrisponda a quella usata per accedere all'account utente di Daniele, quindi aggiorna la password utilizzata per accedere all'account.

#### 3.4.1.6. **SC\_R\_1 Pubblicazione recensione**

Questo scenario descrive come un utente pubblica una recensione per un album.

##### Istanze di attori partecipanti

- Enrico (*utente autenticato*).

##### Flusso di eventi

1. Enrico, dopo aver ascoltato un album diverse volte, decide di pubblicarne una recensione. Enrico invia al sistema una recensione composta da contenuto di testo e da un voto intero compreso tra 1 e 10, e ne richiede la pubblicazione sotto l'album specificato.
2. Il sistema controlla che Enrico sia autenticato, che l'album sia presente nel catalogo della piattaforma e infine convalida la recensione ricevuta, quindi registra la recensione.

#### 3.4.1.7. **SC\_R\_2 Aggiornamento recensione**

Questo scenario descrive come un utente aggiorna una recensione pubblicata dallo stesso in precedenza.

##### Istanze di attori partecipanti

- Giuseppe (*utente autenticato*).

##### Flusso di eventi

1. Giuseppe ha rivisitato dopo molto tempo un album di cui aveva pubblicato una recensione, quindi decide di aggiornarla. Giuseppe invia al sistema il nuovo contenuto di testo e voto della recensione, e ne richiede l'aggiornamento.
2. Il sistema controlla che Giuseppe sia autenticato, che la recensione da aggiornare esista e sia pubblicata da Giuseppe, e che il nuovo contenuto della recensione rispetti i criteri di validazione, quindi la aggiorna.

#### 3.4.1.8. **SC\_R\_3 Rimozione recensione**

Questo scenario descrive come un utente elimina una recensione pubblicata dallo stesso in precedenza.

##### Istanze di attori partecipanti

- Luca (*utente autenticato*).

##### Flusso di eventi

1. Luca non si sente soddisfatto con una recensione che ha pubblicato diverso tempo fa, né è incentivato nell'aggiornarla, quindi richiede al sistema di eliminarla.
2. Il sistema controlla che Luca sia autenticato e che la recensione da eliminare esista e sia stata pubblicata da Luca, quindi la elimina.

#### 3.4.1.9. **SC\_R\_4** *Votazione recensione*

Questo scenario descrive come un utente assegna un voto alla recensione di un altro utente.

##### Istanze di attori partecipanti

- Alessandra (*utente autenticato*).

##### Flusso di eventi

1. Ad Alessandra piace particolarmente una recensione, quindi decide di assegnarle una valutazione positivo.
2. Il sistema controlla che la recensione da votare esista, quindi registra il voto.

#### 3.4.1.10. **SC\_R\_5** *Segnalazione recensione*

Questo scenario descrive come un utente segnala una recensione per contenuto inappropriato.

##### Istanze di attori partecipanti

- Francesco (*utente autenticato*).

##### Flusso di eventi

1. Francesco vuole segnalare una recensione perché la ritiene inappropriata, quindi invia una segnalazione al sistema.
2. Il sistema controlla che Francesco sia autenticato e che la recensione da segnalare esista, quindi registra la segnalazione.



#### 3.4.1.11. **SC\_C\_1 Inserimento progetto in lista d'ascolto**

Questo scenario descrive come un utente aggiunge un progetto musicale nella propria lista d'ascolto.

##### Istanze di attori partecipanti

- Davide (*utente autenticato*)

##### Flusso di eventi

1. Davide ha sentito parlare in maniera entusiasta di un album pubblicato di recente, quindi richiede al sistema di inserirlo in lista d'ascolto.
2. Il sistema controlla che Davide sia autenticato e che l'album sia presente nel catalogo, quindi lo inserisce in lista d'ascolto.

#### 3.4.1.12. **SC\_C\_2 Rimozione progetto da lista d'ascolto**

Questo scenario descrive come un utente rimuove un progetto musicale dalla propria lista d'ascolto.

##### Istanze di attori partecipanti

- Lisa(*utente autenticato*).

##### Flusso di eventi

1. Lisa ha finalmente ascoltato un album che aveva inserito tempo fa in lista d'ascolto, quindi richiede al sistema di rimuoverlo da questa.
2. Il sistema controlla che Lisa sia autenticata e elimina l'album dalla lista d'ascolto.

#### 3.4.1.13. **SC\_A\_1 Aggiornamento privilegi account**

Questo scenario descrive come un utente il cui account è dotato di privilegi di amministrazione aggiorna i privilegi dell'account di un altro utente.

##### Istanze di attori partecipanti

- Alessandro (*utente autenticato dotato di privilegi di amministrazione*).

##### Flusso di eventi

1. Alessandro vuole aumentare il numero di moderatori che si occupano di eliminare contenuto inappropriato dalla piattaforma. Alessandro richiede al sistema di assegnare privilegi di moderazione all'account di un altro utente da lui specificato.
2. Il sistema controlla che Alessandro sia autenticato e sia dotato dei necessari privilegi di amministrazione, quindi aggiorna i privilegi dell'account utente specificato.

#### 3.4.1.14. **SC\_A\_2 Rimozione recensione da parte di moderatore**

Questo scenario descrive come un utente dotato di privilegi di moderazione elimina una recensione di un altro utente.

##### Istanze di attori partecipanti

- Alex (*utente autenticato dotato di privilegi di moderazione*).

##### Flusso di eventi

1. Alex riceve una segnalazione di una recensione e si accorge che questa contiene contenuto inappropriato. Richiede al sistema di rimuovere la recensione.
2. Il sistema controlla che Alex sia autenticato e dotato di privilegi di moderazione e che la recensione da eliminare esista, quindi la rimuove.

### 3.4.2. Modello dei casi d'uso

#### 3.4.2.1. *UC\_UA\_1 Sign up*

Questo caso d'uso descrive come un utente può registrarsi al sistema.

##### Istanze di attori partecipanti

- Utente non autenticato.

##### Precondizioni

- L'utente non è autenticato al sistema.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di registrare un account al sistema.

1. L'utente invia al sistema il proprio username, password e indirizzo e-mail richiedendo la registrazione di un account utente.
2. Il sistema riceve i dati forniti e, dopo aver controllato che l'utente non sia autenticato, aver validato i dati e infine controllato che non ci siano conflitti di username o indirizzo e-mail con altri utenti, registra l'account dell'utente.
3. Il sistema autentica l'utente alla piattaforma.

##### Condizioni d'uscita

- L'account utente viene registrato.
- L'utente viene autenticato al sistema.

##### Eccezioni

- Le credenziali fornite non rispettano i criteri di validazione.
- Il nome utente fornito è già in uso da un altro account utente.
- L'indirizzo e-mail fornito è già in uso da un altro account utente.

#### 3.4.2.2. *UC\_UA\_2 Log in*

Questo scenario descrive come un utente può autenticarsi al sistema.

##### *Istanze di attori partecipanti*

- Utente non autenticato.

##### *Precondizioni*

- L'utente non è autenticato al sistema.

##### *Flusso di eventi*

Il caso d'uso viene iniziato quando un utente dotato di un account decide di accedere al sistema per usufruire delle sue funzionalità richiedenti autenticazione.

1. L'utente invia al sistema il proprio username e password richiedendo l'autenticazione al suo account utente.
2. Il sistema riceve le credenziali e, dopo aver controllato che l'utente non sia autenticato e che queste appartengano ad un account utente registrato, autentica l'utente alla piattaforma.

##### *Condizioni d'uscita*

- L'utente viene autenticato al sistema.

##### *Eccezioni*

- Le credenziali fornite non corrispondono a nessun account utente registrato.

#### 3.4.2.3. *UC\_UA\_3 Log out*

Questo caso d'uso descrive come un utente può de-autenticarsi dal sistema.

##### *Istanze di attori partecipanti*

- Utente autenticato.

##### *Precondizioni*

- L'utente è autenticato al sistema.

##### *Flusso di eventi*

Il caso d'uso viene iniziato quando un utente autenticato decide di de-autenticarsi dal sistema.

1. L'utente richiede al sistema di de-autenticarsi dalla piattaforma.
2. Il sistema controlla che l'utente sia autenticato, quindi ne invalida la sessione de-autenticandolo dalla piattaforma.

##### *Condizioni d'uscita*

- L'utente è de-autenticato dalla piattaforma.

#### 3.4.2.4. **UC\_UA\_4 Recupero credenziali**

Questo caso d'uso descrive come un utente può ripristinare l'accesso al proprio account.

##### Istanze di attori partecipanti

- Utente non autenticato.

##### Precondizioni

- L'utente non è autenticato al sistema.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente che non ricorda la password per accedere al proprio account vuole effettuare il ripristino.

1. L'utente invia al sistema il proprio indirizzo e-mail associato al proprio account utente richiedendone il recupero delle credenziali.
2. Il sistema controlla che l'utente non sia autenticato e che l'indirizzo e-mail ricevuto appartenga all'account di uno dei suoi utenti, quindi invia una mail contenente un collegamento per il recupero della password all'indirizzo fornito dall'utente.
3. L'utente accede alla propria casella di posta elettronica e apre il collegamento fornito nell'e-mail inviata dal sistema.
4. Il sistema controlla nuovamente che l'utente non sia autenticato e che il collegamento non sia invalido o scaduto, quindi richiede all'utente di fornire una nuova password.
5. L'utente invia la nuova password al sistema.
6. Il sistema controlla nuovamente che l'utente non sia autenticato e che il collegamento non sia invalido o scaduto, quindi convalida la password ricevuta e aggiorna la password dell'account dell'utente con la nuova password.

##### Condizioni d'uscita

- La password utilizzata per accedere all'account dell'utente viene aggiornata.

##### Eccezioni

- L'utente fornisce un indirizzo e-mail non associato ad alcun account.
- La nuova password fornita non rispetta i criteri di validazione.
- Il collegamento utilizzato è invalido o è scaduto.

#### 3.4.2.5. **UC\_UA\_5 Aggiornamento credenziali**

Questo caso d'uso descrive come un utente può aggiornare la password utilizzata per accedere al proprio account utente.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di aggiornare la password utilizzata per accedere al proprio account utente.

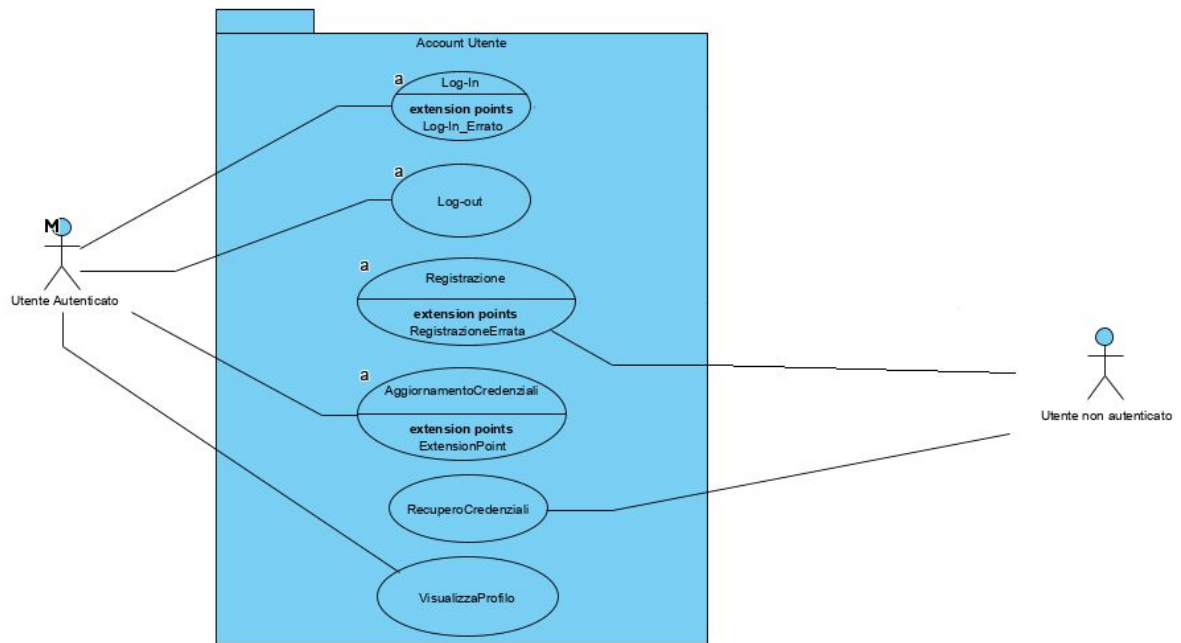
1. L'utente invia al sistema una nuova password accompagnata dalla sua password attuale richiedendone l'aggiornamento.
2. Il sistema controlla che l'utente sia autenticato e che la password ricevuta corrisponda a quella usata per accedere all'account utente dell'utente, quindi aggiorna l'account con la nuova password.

##### Condizioni d'uscita

- L'indirizzo e-mail associato all'account dell'utente viene aggiornato.

##### Eccezioni

- La password fornita dall'utente non corrisponde a quella utilizzata per accedere al suo account.
- L'indirizzo e-mail fornito è già associato ad un altro account utente.





#### 3.4.2.6. **UC\_R\_1 Pubblicazione recensione**

Questo caso d'uso descrive come un utente può pubblicare una recensione per un progetto musicale.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- Il progetto musicale da recensire è presente nel catalogo della piattaforma.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di pubblicare una recensione per un progetto musicale.

1. L'utente invia al sistema una recensione composta da contenuto di testo e da un voto intero compreso tra 1 e 10, e ne richiede la pubblicazione sotto il progetto musicale specificato.
2. Il sistema controlla che l'utente sia autenticato, che il progetto musicale sia presente nel catalogo della piattaforma e infine convalida la recensione ricevuta, quindi registra la recensione.

##### Condizioni d'uscita

- La recensione viene registrata.

##### Eccezioni

- La recensione non rispetta i criteri di validazione.

#### 3.4.2.7. **UC\_R\_2 Aggiornamento recensione**

Questo caso d'uso descrive come un utente può aggiornare una recensione pubblicata dallo stesso in precedenza.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- La recensione da aggiornare è registrata.
- La recensione da aggiornare è stata pubblicata dall'utente stesso.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di aggiornare una recensione pubblicata dallo stesso in precedenza.

1. L'utente invia al sistema il nuovo contenuto di testo e voto della recensione, e ne richiede l'aggiornamento.
2. Il sistema controlla che l'utente sia autenticato, che la recensione da aggiornare esista e sia pubblicata dall'utente stesso, e che il nuovo contenuto della recensione rispetti i criteri di validazione, quindi la aggiorna.

##### Condizioni d'uscita

- La recensione viene aggiornata.

##### Eccezioni

- Il nuovo contenuto della recensione non rispetta i criteri di validazione.

#### 3.4.2.8. **UC\_R\_3 Rimozione recensione**

Questo caso d'uso descrive come un utente può eliminare una recensione pubblicata dallo stesso in precedenza.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- La recensione da rimuovere è registrata.
- La recensione da rimuovere è stata pubblicata dall'utente stesso.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di eliminare una recensione pubblicata dallo stesso in precedenza.

1. L'utente richiede al sistema la rimozione di una recensione pubblicata dallo stesso in precedenza.
2. Il sistema controlla che l'utente sia autenticato, che la recensione da eliminare esista e sia pubblicata dall'utente stesso, quindi la rimuove.

##### Condizioni d'uscita

- La recensione viene rimossa.

#### 3.4.2.9. **UC\_R\_4** *Votazione recensione*

Questo caso d'uso descrive come un utente può assegnare un voto alla recensione di un altro utente.

##### *Istanze di attori partecipanti*

- Utente autenticato.

##### *Precondizioni*

- L'utente è autenticato al sistema.
- La recensione da votare è registrata.
- Un voto identico non è già registrato.

##### *Flusso di eventi*

Il caso d'uso viene iniziato quando un utente decide di assegnare un voto ad una recensione.

1. L'utente invia al sistema una valutazione positiva o negativa richiedendo la registrazione del voto alla recensione specificata.
2. Il sistema controlla che l'utente sia autenticato, che la recensione da votare esista e che un voto identico non sia già registrato, quindi registra il voto.

##### *Condizioni d'uscita*

- Il voto dell'utente alla recensione viene registrato.

##### *Eccezioni*

- Un voto identico da parte dello stesso utente per quella recensione è già registrato.

#### 3.4.2.10. **UC\_R\_5 Segnalazione recensione**

Questo caso d'uso descrive come un utente può segnalare una recensione per contenuto inappropriato.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- La recensione da segnalare è registrata.

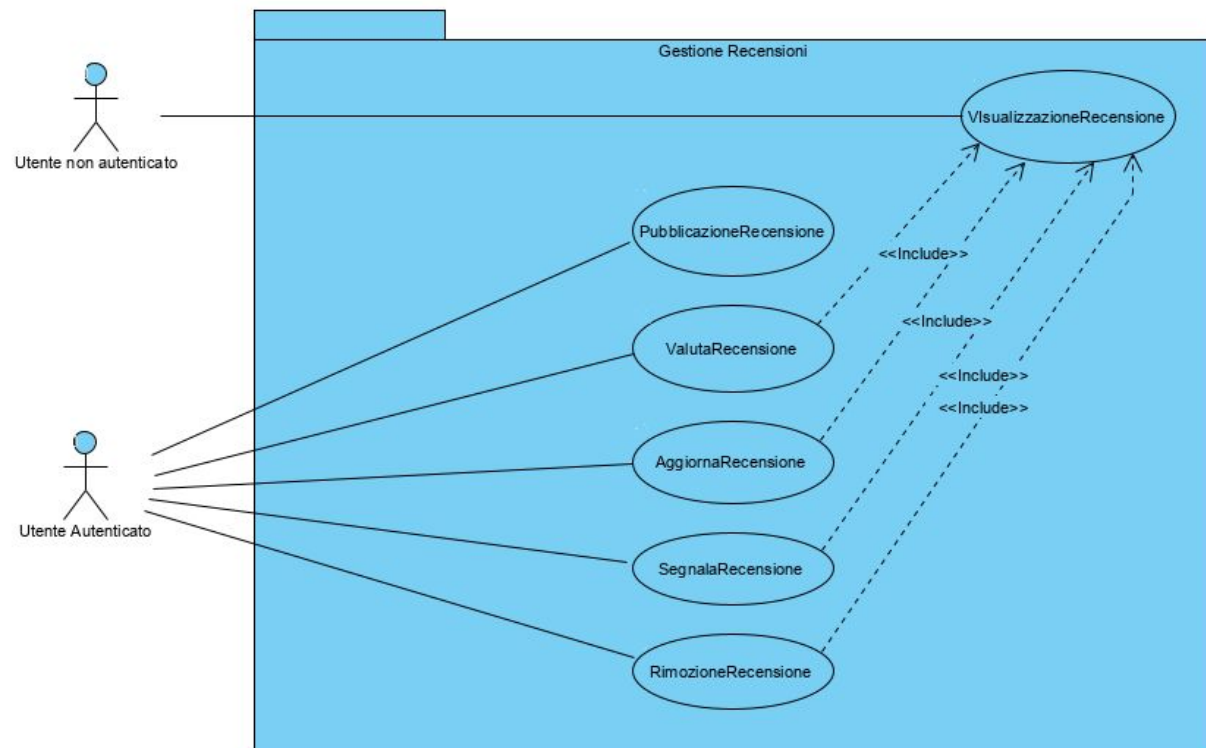
##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di segnalare una recensione per contenuto inappropriato.

1. L'utente richiede al sistema la registrazione di una segnalazione per una recensione specificata.
2. Il sistema controlla che l'utente sia autenticato e che la recensione da segnalare esista, quindi registra la segnalazione.

##### Condizioni d'uscita

- La segnalazione viene registrata.



#### 3.4.2.11. **UC\_C\_1 Inserimento progetto in lista d'ascolto**

Questo caso d'uso descrive come un utente può inserire un progetto musicale nella propria lista d'ascolto.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- Il progetto da inserire in lista d'ascolto è presente nel catalogo della piattaforma.
- Il progetto non è già presente nella lista d'ascolto dell'utente.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di inserire un progetto musicale nella propria lista d'ascolto.

1. L'utente richiede al sistema l'inserimento di un progetto musicale nella propria lista d'ascolto.
2. Il sistema controlla che l'utente sia autenticato, che il progetto sia presente nel catalogo della piattaforma e che questo non sia già nella lista d'ascolto dell'utente, quindi ne effettua l'inserimento in lista d'ascolto.

##### Condizioni d'uscita

- Il progetto viene inserito nella lista d'ascolto dell'utente.

##### Eccezioni

- Il progetto è già presente nella lista d'ascolto dell'utente.

#### 3.4.2.12. **UC\_C\_2 Rimozione progetto da lista d'ascolto**

Questo caso d'uso descrive come un utente può rimuovere un progetto musicale dalla propria lista d'ascolto.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- Il progetto da rimuovere dalla lista d'ascolto è presente nel catalogo della piattaforma.
- Il progetto da rimuovere è presente nella lista d'ascolto dell'utente.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente decide di rimuovere un progetto musicale dalla propria lista d'ascolto.

3. L'utente richiede al sistema la rimozione di un progetto musicale dalla propria lista d'ascolto.
4. Il sistema controlla che l'utente sia autenticato, che il progetto sia presente nel catalogo della piattaforma e che questo sia presente nella lista d'ascolto dell'utente, quindi ne effettua la rimozione dalla lista d'ascolto.

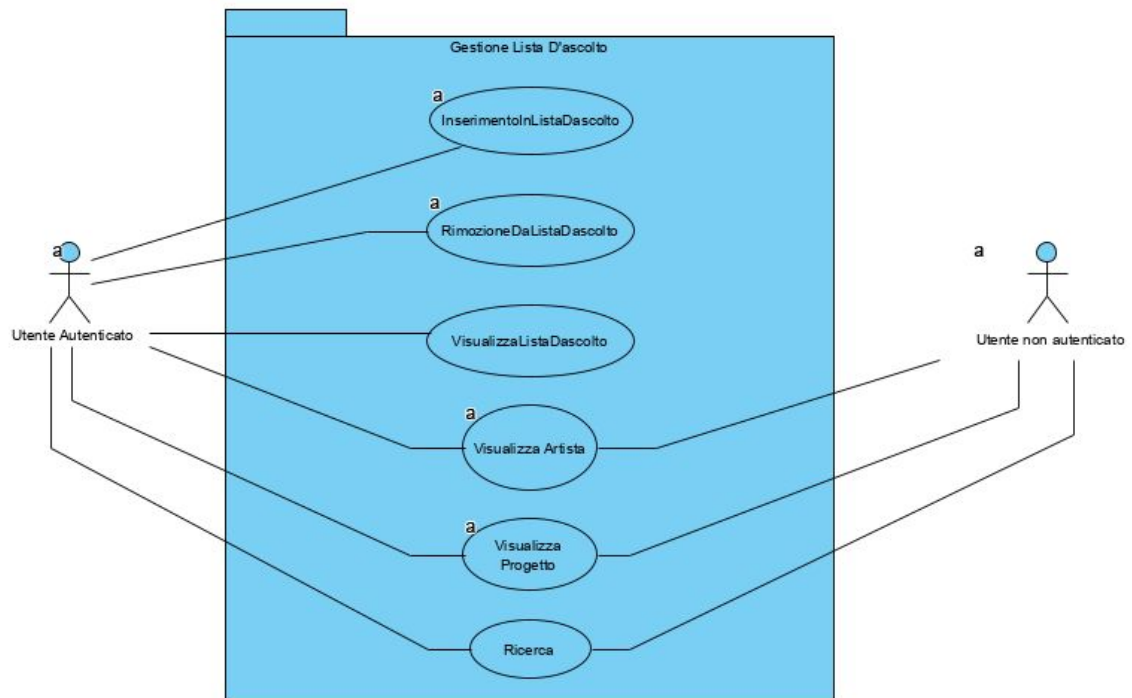
##### Condizioni d'uscita

- Il progetto viene rimosso dalla lista d'ascolto dell'utente.

##### Eccezioni

- Il progetto non è presente nella lista d'ascolto dell'utente.





#### 3.4.2.13. **UC\_A\_1 Aggiornamento privilegi account**

Questo caso d'uso descrive come un utente il cui account è dotato di privilegi di amministrazione può aggiornare i privilegi dell'account di un altro utente.

##### Istanze di attori partecipanti

- Utente autenticato dotato di privilegi di amministrazione.

##### Precondizioni

- L'utente è autenticato al sistema.
- L'utente è dotato dei privilegi di amministrazione.
- L'account utente di cui si devono aggiornare i privilegi è registrato al sistema.

##### Flusso di eventi

Il caso d'uso viene iniziato quando un utente autenticato dotato di privilegi di amministrazione decide di aggiornare i privilegi associati all'account di un altro utente..

1. L'utente richiede al sistema di aggiornare i privilegi dell'account di un altro utente da lui specificato.
2. Il sistema controlla che l'utente sia autenticato e sia dotato dei necessari privilegi di amministrazione, quindi controlla che l'account utente specificato esista e ne aggiorna i privilegi.

##### Condizioni d'uscita

- I privilegi dell'account utente specificato vengono aggiornati.

#### 3.4.2.14. **UC\_A\_2 Rimozione recensione da parte di moderatore**

Questo caso d'uso descrive come un utente dotato di privilegi di moderazione può eliminare una recensione di un altro utente.

##### Istanze di attori partecipanti

- Utente autenticato.

##### Precondizioni

- L'utente è autenticato al sistema.
- L'utente è dotato di privilegi di moderazione.

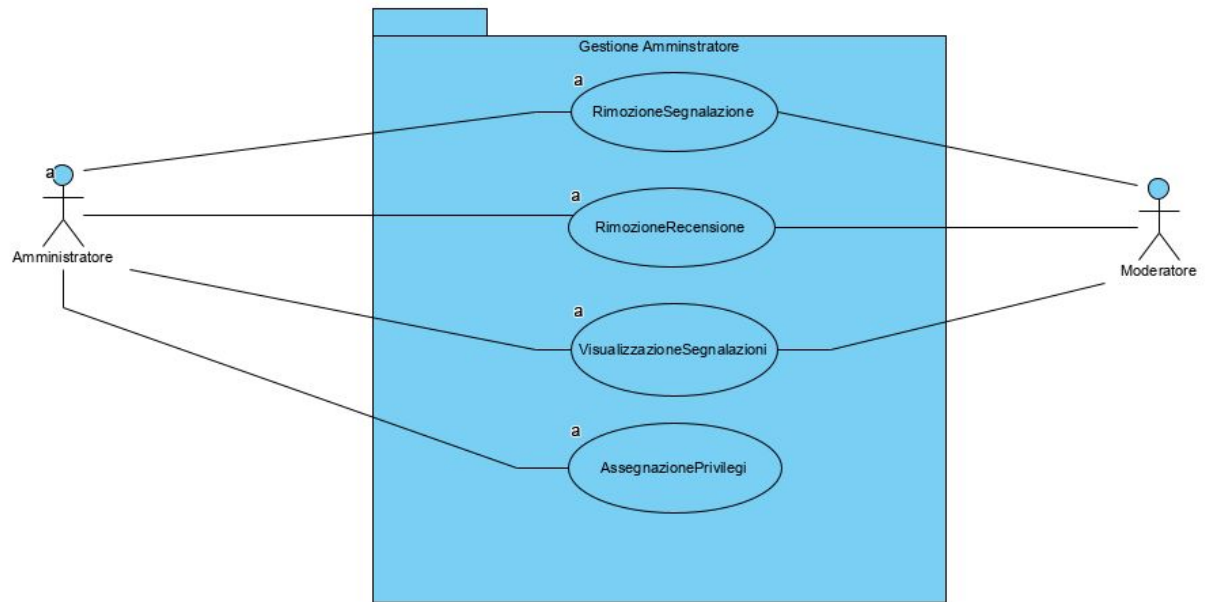
##### Flusso di eventi

Questo caso d'uso viene iniziato quando un utente dotato di privilegi di moderazione decide di eliminare una recensione ritenuta inappropriata.

5. L'utente richiede al sistema la rimozione di una recensione specificata.
6. Il sistema controlla che l'utente sia autenticato e dotato di privilegi di moderazione e che la recensione da eliminare esista, quindi la rimuove.

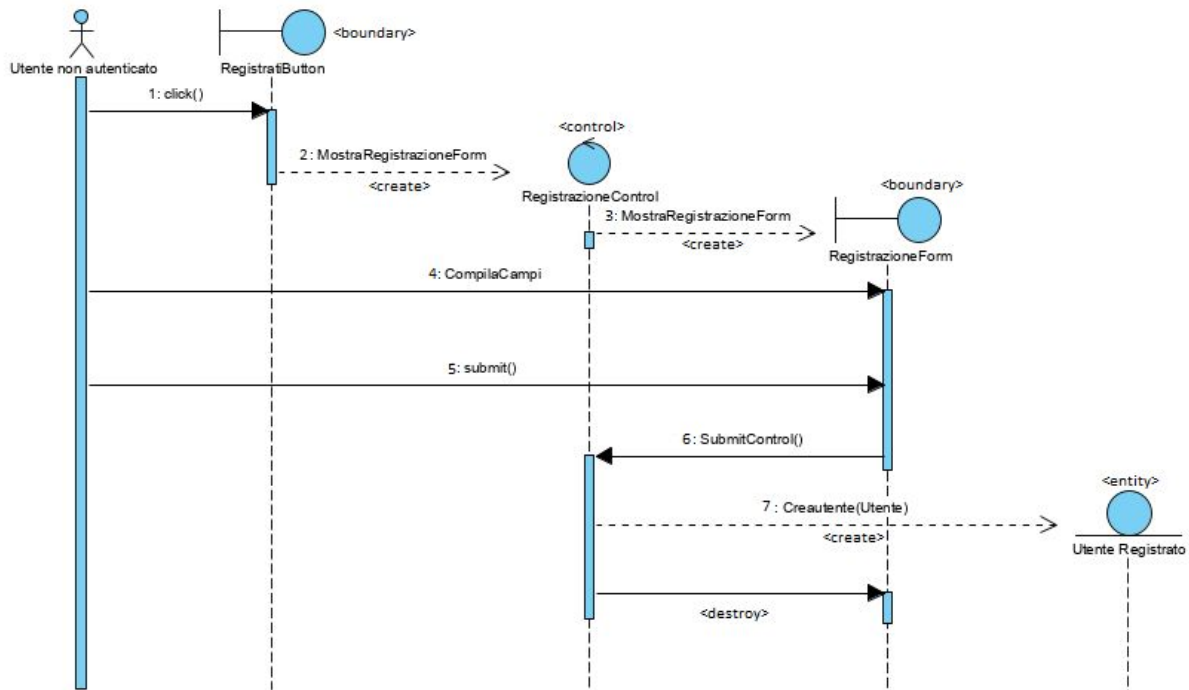
##### Condizioni d'uscita

La recensione viene rimossa.

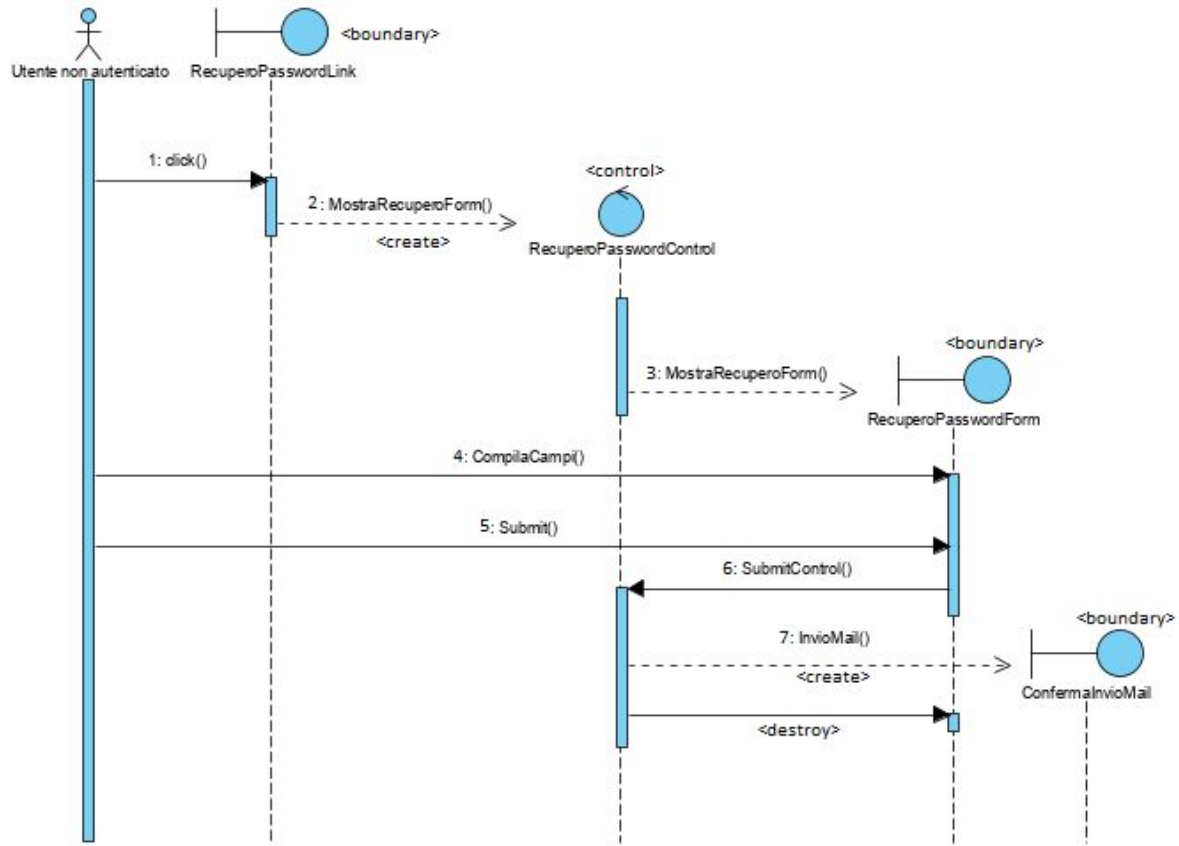


### 3.4.3. Modello dinamico

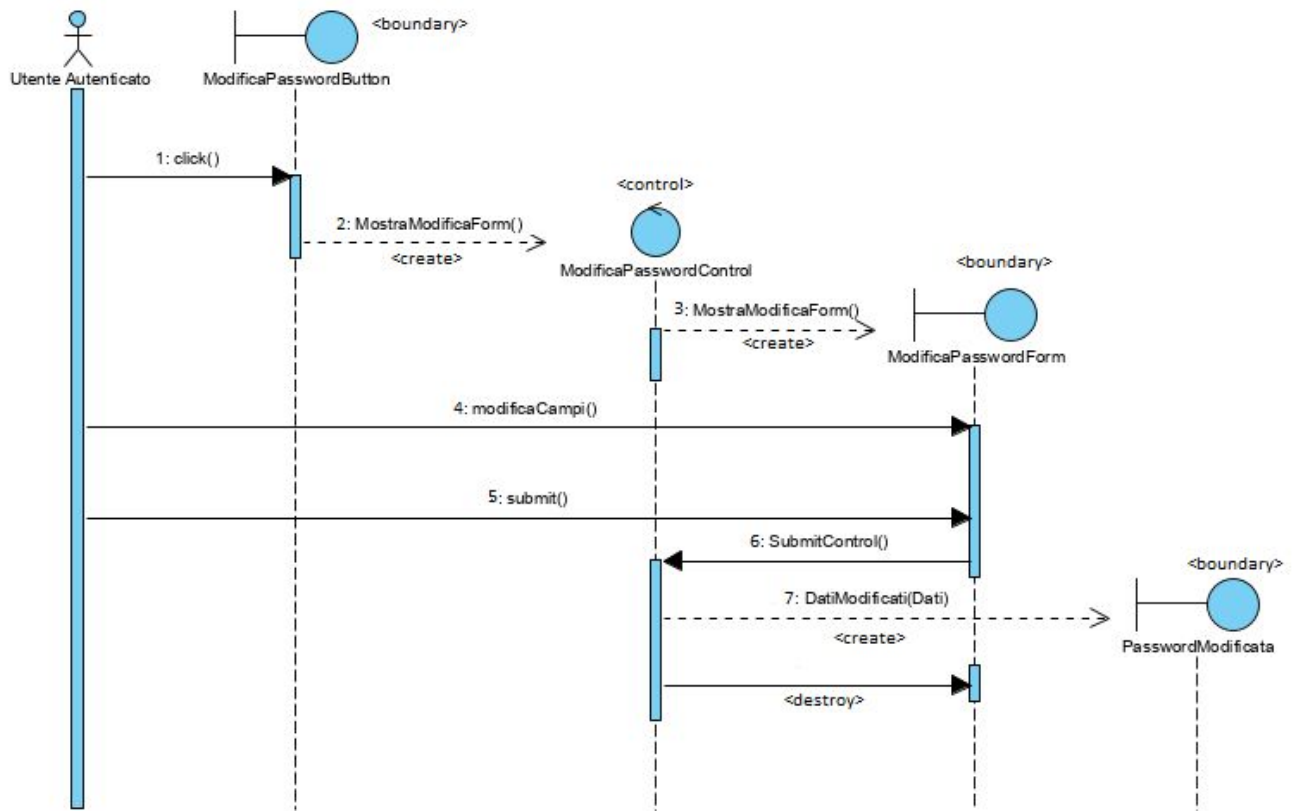
#### 3.4.3.1. *SD\_UA\_1* Sign up



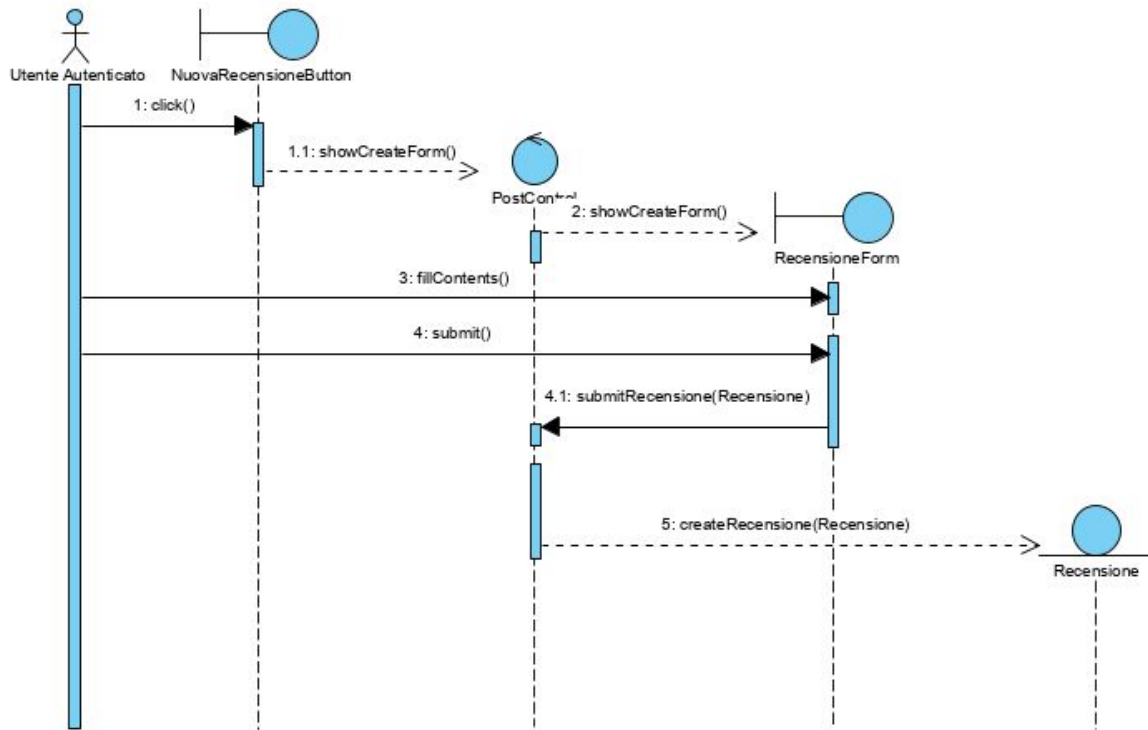
### 3.4.3.2. *SD\_UA\_4* Recupero credenziali



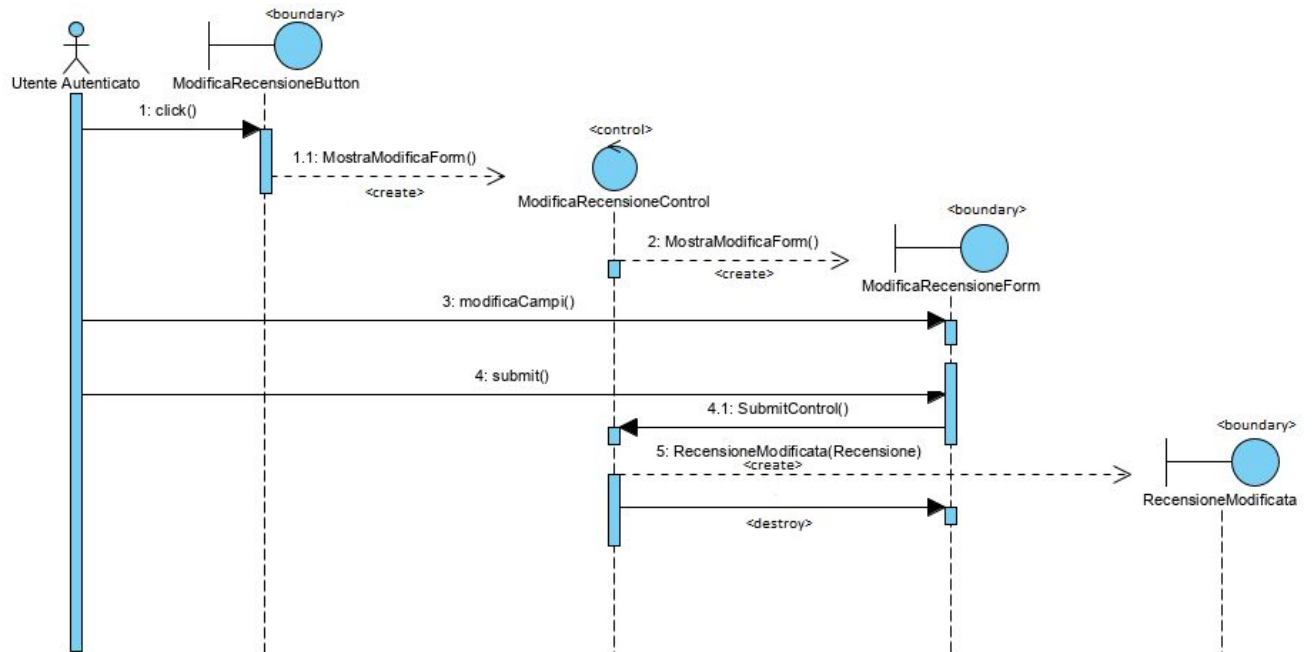
### 3.4.3.3. *SD\_UA\_5* Aggiornamento credenziali



#### 3.4.3.4. *SD\_R\_1* Pubblicazione recensione

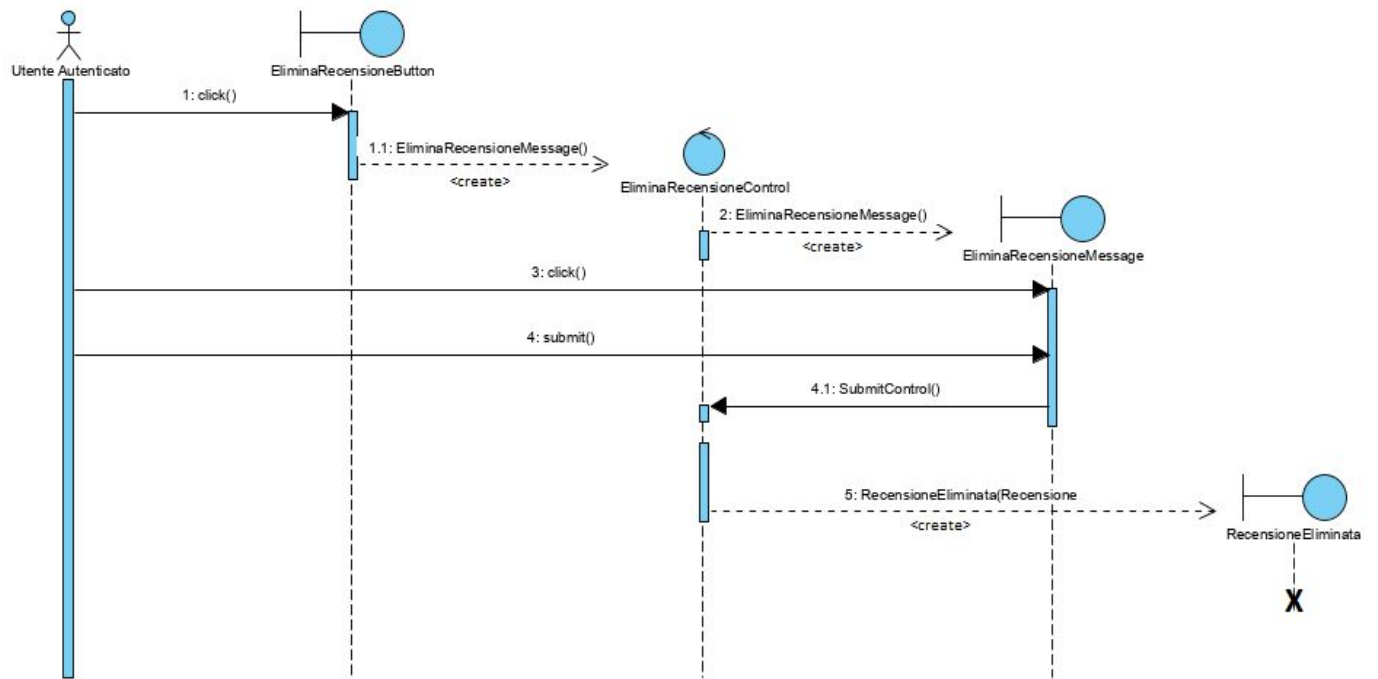


### 3.4.3.5. *SD\_R\_2* Aggiornamento recensione

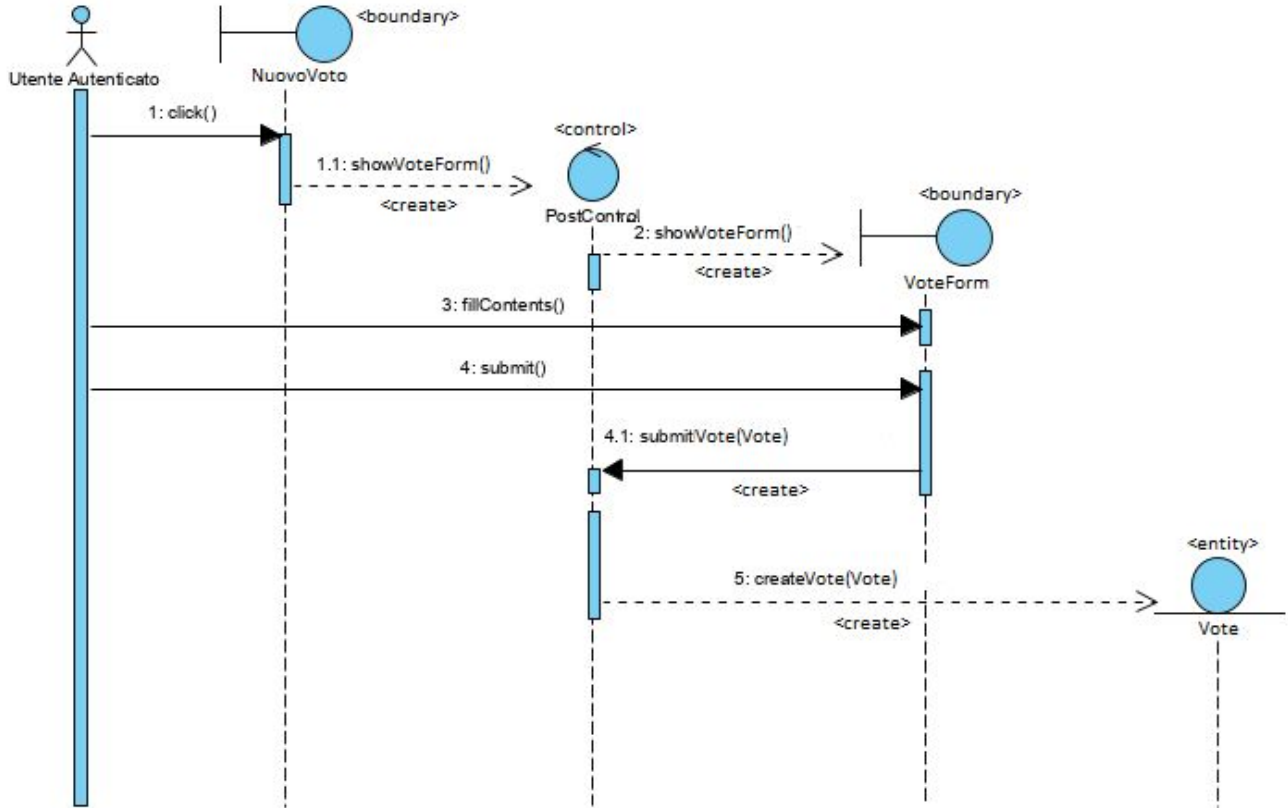




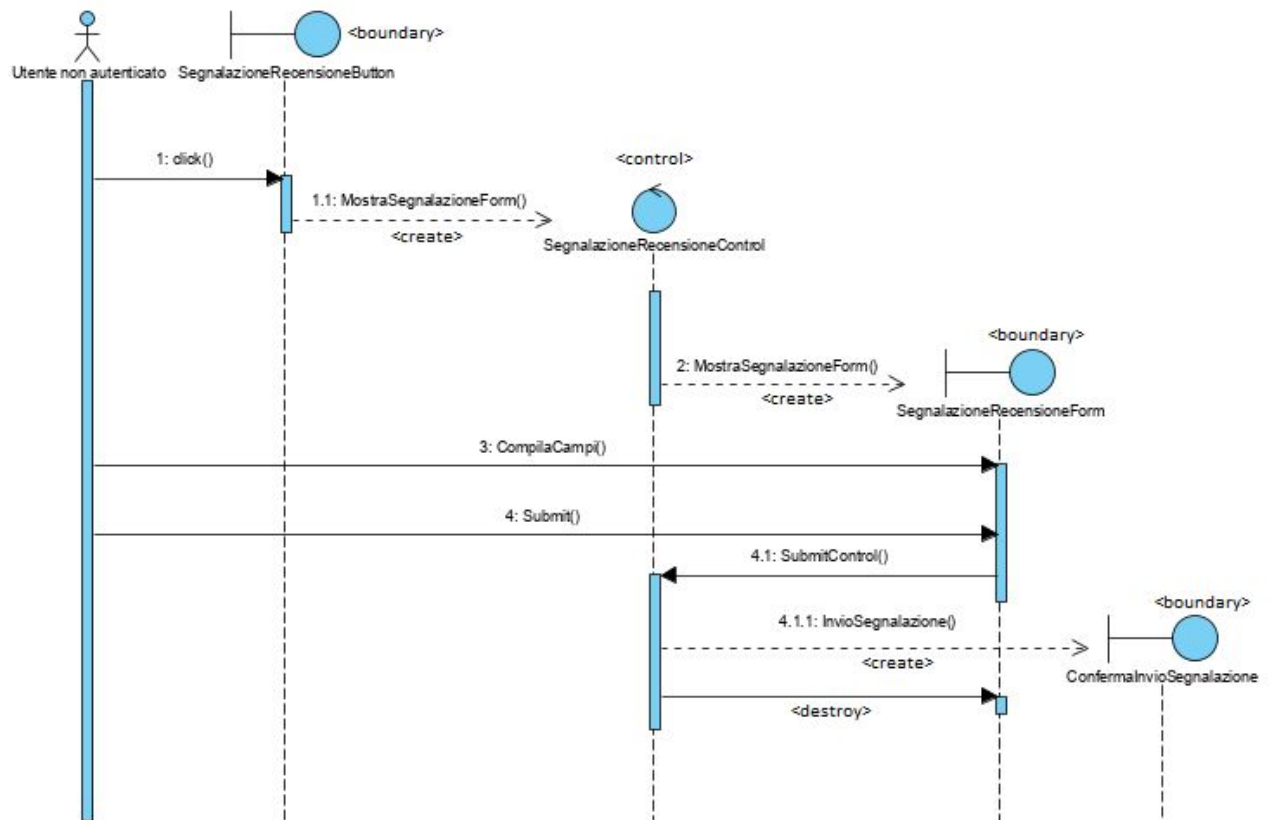
### 3.4.3.6. *SD\_R\_3* Eliminazione recensione



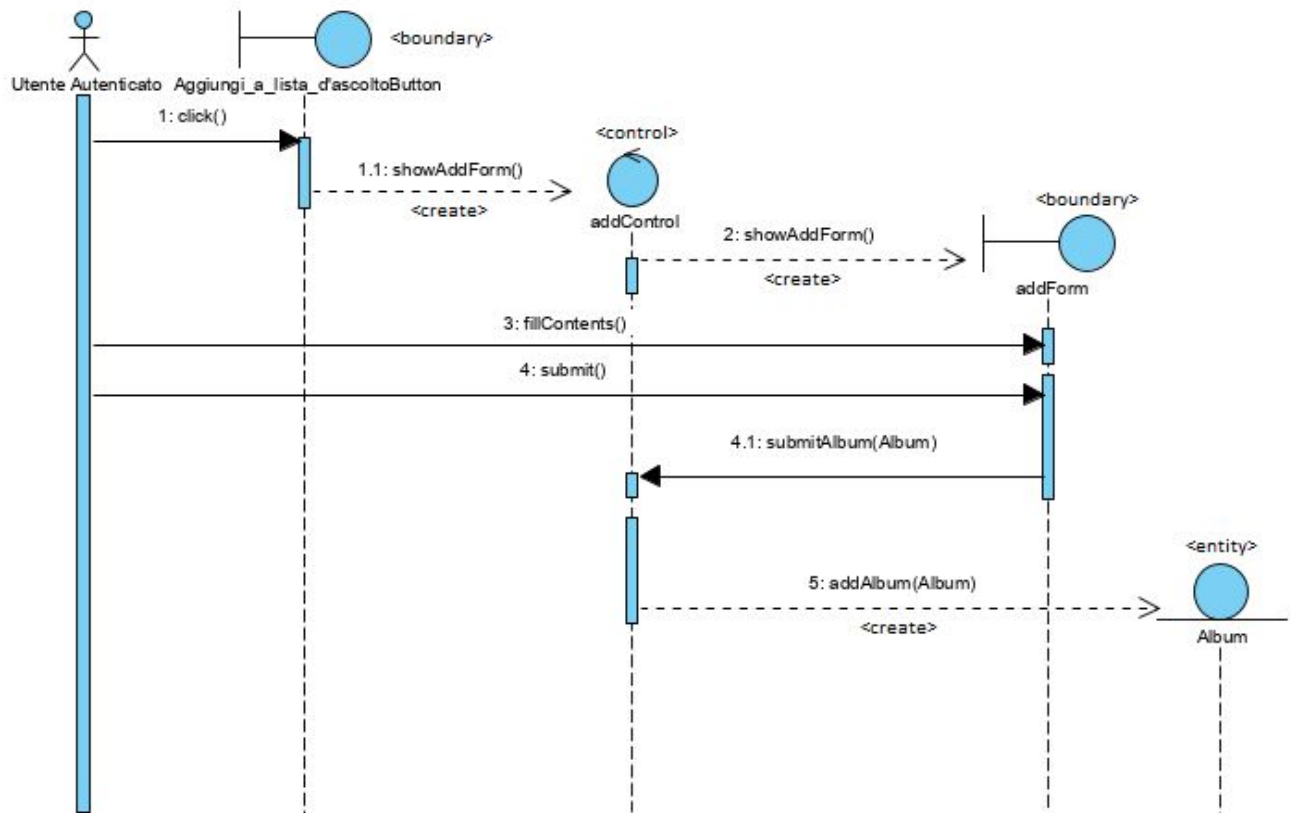
### 3.4.3.7. *SD\_R\_4* Votazione recensione



### 3.4.3.8. *SD\_R\_5* Segnalazione recensione



### 3.4.3.9. *SD\_LB\_1* Inserimento progetto in lista d'ascolto



### 3.4.3.10. *SD\_LB\_2* Rimozione progetto da lista d'ascolto

