*Progetto Ingegneria del Software*

# TSR: Top Scientist Review



Prof. Orazio Tomarchio

Gianluca Cristaudo

Samuele Gulino

Ingegneria Informatica Magistrale LM-32 2019/2020

Sommario

[TSR: Top Scientist Review 1](#_Toc35464681)

[Prefazione 3](#_Toc35464682)

[1. Ideazione e analisi dei requisiti 4](#_Toc35464683)

[1.1 Introduzione 4](#_Toc35464684)

[1.2 Analisi dei requisiti 4](#_Toc35464685)

[1.3 Obiettivi e casi d’uso 5](#_Toc35464686)

[1.4 Modello dei casi d’uso 5](#_Toc35464687)

[1.5 Documento di visione 18](#_Toc35464688)

[1.6 Regole di business 18](#_Toc35464689)

[1.7 Specifiche supplementari 19](#_Toc35464690)

[1.8 Glossario 20](#_Toc35464691)

[2. Analisi orientata agli oggetti 21](#_Toc35464692)

[2.1 Modello di Dominio 21](#_Toc35464693)

[2.2 Diagrammi di sequenza di sistema e contratti delle operazioni 24](#_Toc35464694)

[3. Progettazione orientata agli oggetti 29](#_Toc35464695)

[3.1 Diagramma delle classi 29](#_Toc35464696)

[3.2 Diagrammi di sequenza 29](#_Toc35464697)

[4. Testing 30](#_Toc35464698)

[4.1 Introduzione 30](#_Toc35464699)

[4.2 Individuazione dei casi di test e Testing Unitario 31](#_Toc35464700)

[4.3 Test di Sistema 32](#_Toc35464701)

[5. Persistenza 32](#_Toc35464702)

# Prefazione

Il seguente documento presenta una descrizione complessiva del processo di sviluppo dell’applicazione Top Scientist Review relativa al progetto di Ingegneria del Software. Il software è stato implementato in linguaggio Java sfruttando l’IDE Eclipse, mentre la modellazione e progettazione UML è stata svolta con il software Astah Professional. Vengono di seguito presentate le versioni finali dei vari elaborati prodotti ad ogni iterazione dello sviluppo. Vengono infine descritte le fasi di testing e di aggiunta di persistenza di alcune classi.

# 1. Ideazione e analisi dei requisiti

## 1.1 Introduzione

Durante la fase di ideazione si analizza il problema al fine di ricavarne un’idea generale del progetto riguardo la sua realizzazione e la sua fattibilità. Per raggiungere questo obiettivo si analizzano i seguenti modelli: Modello dei Casi d’Uso, Documento di Visione, Specifiche Supplementari, Regole di Business e Glossario.

## 1.2 Analisi dei requisiti

Viene richiesta la realizzazione di un software per la gestione di invio e referaggio di una rivista scientifica. Il software deve consentire: agli autori di inviare articoli per la loro valutazione ed eventuale pubblicazione; ai revisori di valutare i lavori sottomessi; all’editore di supervisionare l’intero processo.

Più dettagliatamente:

* un autore registrato sulla piattaforma deve poter sottomettere un articolo;
* gli articoli sottomessi vengono assegnati ad un revisore senior ed almeno 2 junior appartenenti al comitato editoriale;
* i revisori forniscono valutazione sugli articoli a loro assegnati, in particolare:

1. i revisori senior esprimono un giudizio e un commento in forma libera;
2. i revisori junior assegnano un voto sulla base di varie caratteristiche.

* sulla base dei giudizi e voti dei revisori un articolo può essere candidato all’accettazione o al rifiuto;
* l’editore prende la decisione finale, in particolare:
  + 1. se l’articolo è candidato all’accettazione può essere accettato definitivamente o tornare sotto esame;
    2. se è candidato al rifiuto può essere rifiutato o tornare sotto esame.
* a decisione presa, gli autori vengono messi a conoscenza dell’esito e dei giudizi e voti assegnati dai revisori, che rimangono anonimi;
* il sistema deve mantenere traccia di ogni attività.

## 1.3 Obiettivi e casi d’uso

Nel seguito definiamo i vari casi d’uso emersi dall’analisi dei requisiti sopra riportata, evidenziandone gli attori principale e lo scopo.

|  |  |  |
| --- | --- | --- |
| **Attore** | **Obiettivo** | **Caso d’uso** |
| Autore | Registrarsi sulla piattaforma | UC1: Registrazione autore |
| Revisore | Registrarsi sulla piattaforma | UC2: Registrazione revisore |
| Autore | Sottoporre un articolo a revisione al fine di pubblicarlo sulla rivista | UC3: Sottomissione articolo |
| Revisore | Fornire una valutazione ad un articolo assegnato | UC4: Attribuzione valutazione |
| Autore | Accedere ai giudizi ricevuti sugli articoli già valutati, revisori anonimi | UC5: Accesso valutazioni |
| Editore | Leggere le valutazioni dei revisori ed esprimere una decisione finale | UC6: Valutazione finale editore |
| Editore | Visualizzare articoli candidati per l’accettazione | UC7: Visualizzazione articoli |

## 1.4 Modello dei casi d’uso

Seguono le descrizioni dettagliate dei vari casi d’uso.

**UC1: Registrazione autore**

|  |  |
| --- | --- |
| **Nome del caso d’uso** | UC1: Registrazione autore |
| **Portata** | Applicazione TSR |
| **Livello** | Obiettivo **AUTORE** |
| **Attore primario** | Autore |
| **Parti interessate e interessi** | Autore: vuole registrarsi in maniera rapida e efficiente. Vuole una visualizzazione chiara dei dati inseriti e un messaggio esplicativo che indica che la registrazione è stata effettuata senza errori.  Sistema: vuole memorizzare accuratamente le registrazioni effettuate e soddisfare gli interessi degli **AUTORI**. Vuole che l’elenco degli **AUTORI** registrati sia aggiornato in tempo reale. |
| **Pre-condizioni** | L’autore non è ancora autenticato presso il sistema. |
| **Garanzia di successo:** | La registrazione viene effettuata correttamente. L’autore presenta un “account” presso il sistema, contraddistinto da una username e da una password. I dati vengono aggiornati all’interno del sistema. |
| **Scenario principale di successo** | 1. L’autore accede al sistema; 2. L’autore (non ancora registrato) accede alla pagina di registrazione; 3. L’autore inserisce i propri dati all’interno dei vari form richiesti (nome, cognome, indirizzo e-mail, affiliazione);   4. L’autore inserisce una password;  5. Il sistema controlla che l’e-mail sia univoca;  6. Il sistema controlla che l’e-mail sia valida;  7. Il sistema controlla che la password soddisfi i vincoli imposti;  8. Il sistema controlla che tutti i campi obbligatori siano stati inseriti;  9. Il sistema mostra un riepilogo dei dati inseriti.  10. L’autore conferma i dati inseriti.  11. Il sistema registra correttamente il nuovo utente.  12. Il sistema mostra un messaggio che conferma l’avvenuta registrazione. |
| **Estensioni** | 1. L’autore inserisce un username già censita al sistema: 2. L’autore non viene registrato presso il sistema; 3. Viene visualizzato un errore esplicativo; 4. Viene fornita all’autore la possibilità di inserire un altro username. 5. L’autore inserisce una password che non rispetta i vincoli: 6. L’autore non viene registrato presso il sistema; 7. Viene visualizzato un errore esplicativo; 8. Viene fornita all’autore la possibilità di inserire una nuova password. 9. L’autore inserisce una e-mail non valida: 10. L’autore non viene registrato presso il sistema; 11. Viene visualizzato un errore esplicativo; 12. Viene fornita all’autore la possibilità di inserire una nuova e-mail. 13. L’autore non compila tutti i campi obbligatori: 14. L’autore non viene registrato presso il sistema; 15. Viene visualizzato un errore esplicativo; 16. Viene fornita all’autore la possibilità di inserire i campi mancanti. 17. In qualsiasi momento, il sistema fallisce e ha un arresto improvviso: 18. L’autore riavvia il software e richiede il ripristino dello stato precedente del sistema; 19. Il sistema ricostruisce lo stato precedente.   2a. Il sistema rileva delle anomalie che impediscono il ripristino:   1. Il sistema segnala un errore all’autore, registra l’errore, e passa in uno stato pulito; 2. L’autore inizia una nuova registrazione. |
| **Requisiti speciali** | Internazionalizzazione della lingua sul testo visualizzato. |
| **Elenco delle varianti tecnologiche e dei dati** | Per ciascun autore devono essere visualizzati i campi: nome, cognome, e-mail e affiliazione. |
| **Frequenza di ripetizione** | Potrebbe essere quasi ininterrotta. |
| **Varie** |  |

**UC2: Registrazione revisore**

1. Il revisore accede alla pagina di registrazione;

2. Il revisore inserisce i propri dati;

3. Il sistema controlla che i dati inseriti siano idonei;

4. Il sistema registra correttamente il nuovo utente.

**UC3: Sottomissione articolo**

|  |  |
| --- | --- |
| **Nome del caso d’uso** | UC3: Sottomissione articolo |
| **Portata** | Applicazione TSR |
| **Livello** | Obiettivo autore |
| **Attore primario** | Autore |
| **Parti interessate e interessi** | Autore: vuole inviare il proprio articolo in maniera rapida e efficiente. Vuole una visualizzazione chiara dei dati inseriti e un messaggio esplicativo che indica che l’articolo sia stato mandato correttamente.  Sistema: vuole registrare accuratamente gli articoli inviati dagli autori. Vuole che le informazioni relative agli articoli inviati siano aggiornate. |
| **Pre-condizioni** | L’autore deve essere autenticato presso il sistema.  L’articolo deve essere scritto almeno da un autore registrato. |
| **Garanzia di successo** | L’articolo viene inviato correttamente al sistema.  L’articolo sottomesso viene assegnato ad un revisore senior e ad almeno 2 revisori junior, tutti appartenenti al comitato editoriale. |
| **Scenario principale di successo** | 1. L’autore accede al sistema; 2. L’autore inserisce e-mail e password; 3. Il sistema controlla la correttezza dei dati inseriti; 4. Il sistema mostra le sezioni rese disponibili per l’autore. 5. L’autore seleziona sottomissione articolo. 6. Il sistema chiede di inserire gli autori. 7. L’autore inserisce eventuali autori dell’articolo diversi da sé stesso, uno per volta. 8. Il passo 7 è ripetuto fin quando ci sono altri autori.   9. Il sistema chiede i dati dell’articolo da inserire (titolo, abstract, dimensione, nome del file);  10. L’autore inserisce le informazioni e conferma;  11. Il sistema controlla che tutti i dati inseriti siano corretti;  12. Il sistema mostra un riepilogo e chiede all’autore se è sicuro di voler consegnare l’articolo;  13. L’autore conferma la sottomissione dell’articolo;  14. Il sistema salva l’articolo;  15. Il sistema crea una traccia della sottomissione effettuata.  16. Il sistema associa dei revisori all’articolo.  17. Il sistema crea una traccia dell’assegnazione effettuata (una per ciascun revisore).  18. Il sistema visualizza a schermo un messaggio per far capire all’autore che tutto è andato a buon fine. |
| **Estensioni** | 1. L’autore inserisce un username non registrata nel sistema: 2. Il revisore non viene autenticato presso il sistema; 3. Viene visualizzato un errore esplicativo; 4. Viene fornita all’autore la possibilità di inserire nuovamente l’username.   3a. L’autore ha dimenticato l’username:   1. L’autore clicca su “username dimenticata”; 2. Il sistema visualizza un form per il recupero dell’username; 3. Il sistema richiede l’e-mail di registrazione; 4. L’autore inserisce l’e-mail e conferma; 5. Il sistema verifica che l’e-mail inserita dal revisore sia registrata; 6. Il sistema manda l’e-mail con i dati per il recupero dell’username.   6a. L’autore inserisce una e-mail non registrata nel sistema:   1. Il sistema non manda l’e-mail per il recupero dell’username; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. L’autore inserisce una password errata: 5. L’autore non viene autenticato presso il sistema; 6. Viene visualizzato un errore esplicativo; 7. Viene fornita all’autore la possibilità di inserire nuovamente la password.   3a. L’autore ha dimenticato la password:   1. L’autore clicca su “password dimenticata”; 2. Il sistema visualizza un form per il recupero della password; 3. Il sistema richiede l’username di registrazione; 4. L’autore inserisce l’username e conferma; 5. Il sistema controlla che l’username sia registrata; 6. Il sistema manda l’e-mail, con i dati per il recupero della password, all’e-mail associata all’username inserita dall’autore.   6a. L’autore inserisce un username non registrato nel sistema:   1. Il sistema non manda l’e-mail per il recupero della password; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. L’autore non inserisce tutti i dati obbligatori: 5. Il sistema visualizza un errore esplicativo, dove “mostra” tutti i dati da inserire; 6. L’autore inserisce i dati mancanti; 7. Il sistema controlla che tutti i dati siano stati inseriti correttamente e salva l’articolo. 8. Il path inserito dall’autore per il file da caricare non esiste nel pc dell’autore: 9. Il sistema mostra l’errore esplicativo; 10. Viene fornita, all’autore, la possibilità di inserire nuovamente il path del file. 11. In qualsiasi momento il sistema fallisce e ha un arresto improvviso: 12. L’autore riavvia il software e richiede il ripristino dello stato precedente del sistema; 13. Il sistema ricostruisce lo stato precedente.   2a. Il sistema rileva delle anomalie che impediscono il ripristino:   1. Il sistema segnala un errore all’autore, registra l’errore, e passa in uno stato pulito; 2. L’autore inizia una nuova sottomissione. |
| **Requisiti principali** | Internazionalizzazione della lingua sul testo visualizzato |
| **Elenco delle varianti tecnologiche e dei dati** | Per ciascun articolo devono essere visualizzati i campi: titolo, abstract, dimensione in Kb del file, nome completo del file e gli autori che hanno scritto l’articolo. |
| **Frequenza di ripetizione** | Potrebbe essere quasi ininterrotta. |
| **Varie** |  |

**UC4: Attribuzione valutazione**

|  |  |
| --- | --- |
| **Nome del caso d’uso** | UC4: Attribuzione valutazione |
| **Portata** | Applicazione TSR |
| **Livello** | Obiettivo revisore |
| **Attore primario** | Revisore |
| **Parti interessate e interesse** | Revisore: vuole rilasciare in maniera rapida e efficiente la propria valutazione. Vuole una visualizzazione chiara dei dati inseriti.  Sistema: vuole registrare accuratamente le valutazioni rilasciate e soddisfare gli interessi dei revisori. Vuole che le valutazioni siano aggiornate. |
| **Pre-condizioni** | Il revisore deve essere autenticato presso il sistema.  L’articolo deve essere assegnato al revisore. |
| **Garanzia di successo** | Il sistema salva correttamente la valutazione e la data in cui è avvenuta.  Il sistema rimuove dalla sezione “articoli” l’articolo valutato dal revisore. |
| **Scenario principale di successo** | 1. Il revisore accede al sistema; 2. Il revisore inserisce e-mail e password; 3. Il sistema controlla la correttezza dei dati inseriti e autentica il revisore; 4. Il revisore seleziona la sezione “articoli”; 5. Il sistema visualizza la lista di articoli da valutare; 6. Il revisore seleziona l’articolo da valutare; 7. Il sistema visualizza l’articolo e mostra l’interfaccia per valutarlo; 8. Il revisore inserisce la valutazione e conferma; 9. Il sistema controlla che tutti i campi siano stati inseriti correttamente; 10. Il sistema salva la valutazione inserita dal revisore; 11. Il sistema visualizza a schermo un messaggio per far capire al revisore che tutto è andato a buon fine. |
| **Estensioni** | 1. Il revisore inserisce un username non registrata nel sistema: 2. Il revisore non viene autenticato presso il sistema; 3. Viene visualizzato un errore esplicativo. 4. Viene fornita al revisore la possibilità di inserire nuovamente l’username.   3a. Il revisore ha dimenticato l’username:   1. Il revisore clicca su “username dimenticata”; 2. Il sistema visualizza un form per il recupero dell’username; 3. Il sistema richiede l’e-mail di registrazione; 4. Il revisore inserisce l’e-mail e conferma; 5. Il sistema verifica che l’e-mail inserita dal revisore sia registrata; 6. Il sistema manda l’e-mail con i dati per il recupero dell’username;          6a. Il revisore inserire una e-mail non registrata               nel sistema:   1. Il sistema non manda l’e-mail per il recupero dell’username; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. Il revisore inserisce una password errata: 5. Il revisore non viene autenticato presso il sistema; 6. Viene visualizzato un errore esplicativo; 7. Viene fornita al revisore la possibilità di inserire nuovamente la password.   3a. Il revisore ha dimenticato la password:   1. Il revisore clicca su “password dimenticata”; 2. Il sistema visualizza un form per il recupero della password; 3. Il sistema richiede l’username di registrazione; 4. Il revisore inserisce l’username e conferma; 5. Il sistema controlla che l’username sia registrata; 6. Il sistema manda l’e-mail, con i dati per il recupero della password, all’e-mail associata all’username inserita dal revisore.   6a. Il revisore inserisce una username non registrata nel sistema:   1. Il sistema non manda l’e-mail per il recupero della password; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. In qualsiasi momento il sistema fallisce e ha un arresto improvviso: 5. Il revisore riavvia il software e richiede il ripristino dello stato precedente del sistema; 6. Il sistema ricostruisce lo stato precedente.   2a. Il sistema rileva delle anomalie che impediscono il ripristino:   1. Il sistema segnala un errore al revisore, registra l’errore, e passa in uno stato pulito; 2. Il revisore inizia una nuova autenticazione. 3. L’autore seleziona la sezione “articoli”, ma non c’è nessun articolo da valutare: 4. Viene visualizzato un messaggio che informa dell’assenza di articoli da valutare.   e. l revisore junior inserisce un voto non compreso fra 0 e 9:   * 1. Il sistema visualizza un errore esplicativo, dove dice di inserire un voto fra 0 e 9;   Il revisore junior reinserisce i voti. |
| **Requisiti speciali** | Internazionalizzazione della lingua sul testo visualizzato. |
| **Elenco delle varianti tecnologiche e dei dati** | Per ciascun revisore devono essere visualizzati i campi: nome, cognome, e-mail, nazionalità e anni di esperienza. |
| **Frequenza di ripetizione** | Potrebbe essere quasi ininterrotta. |
| **Varie** |  |

**UC5:Accesso valutazioni**

|  |  |
| --- | --- |
| **Nome del caso d’uso** | UC5: Accesso valutazioni |
| **Portata** | Applicazione TSR |
| **Livello** | Obiettivo **AUTORE** |
| **Attore primario** | Autore |
| **Parti interessate e interessi** | Autore: vuole che le valutazioni attribuite ai suoi articoli siano visualizzate in modo chiaro.  Sistema: vuole che l’autore possa accedere alle valutazioni in modo chiaro, ma che non possa conoscere l’identità dei revisori. |
| **Pre-condizioni** | Gli articoli mostrati si trovano in uno stato finale, sia esso di accettazione o rifiuto. |
| **Garanzia di successo:** | Vengono stampati correttamente gli articoli con le relative valutazioni. |
| **Scenario principale di successo** | 1. L’autore accede al sistema; 2. L’autore inserisce e-mail e password; 3. Il sistema controlla la correttezza dei dati inseriti e autentica l’autore; 4. Il sistema mostra le sezioni rese disponibili per l’autore; 5. L’autore seleziona Elenco articoli valutati; 6. Il sistema mostra l’elenco degli articoli per i quali l’editore ha preso una decisione finale di accettazione o rifiuto; 7. L’autore seleziona un articolo; 8. Il sistema mostra le valutazioni attribuite all’articolo ma non i revisori. |
| **Estensioni** | 1. L’autore inserisce un username non registrata nel sistema: 2. Il revisore non viene autenticato presso il sistema; 3. Viene visualizzato un errore esplicativo; 4. Viene fornita all’autore la possibilità di inserire nuovamente l’username.   3a. L’autore ha dimenticato l’username:   1. L’autore clicca su “username dimenticata”; 2. Il sistema visualizza un form per il recupero dell’username; 3. Il sistema richiede l’e-mail di registrazione; 4. L’autore inserisce l’e-mail e conferma; 5. Il sistema verifica che l’e-mail inserita dal revisore sia registrata; 6. Il sistema manda l’e-mail con i dati per il recupero dell’username.   6a. L’autore inserisce una e-mail non registrata nel sistema:   1. Il sistema non manda l’e-mail per il recupero dell’username; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. L’autore inserisce una password errata: 5. L’autore non viene autenticato presso il sistema; 6. Viene visualizzato un errore esplicativo; 7. Viene fornita all’autore la possibilità di inserire nuovamente la password.   3a. L’autore ha dimenticato la password:   1. L’autore clicca su “password dimenticata”; 2. Il sistema visualizza un form per il recupero della password; 3. Il sistema richiede l’username di registrazione; 4. L’autore inserisce l’username e conferma; 5. Il sistema controlla che l’username sia registrata; 6. Il sistema manda l’e-mail, con i dati per il recupero della password, all’e-mail associata all’username inserita dall’autore.   6a. L’autore inserisce un username non registrato nel sistema:   1. Il sistema non manda l’e-mail per il recupero della password; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. In qualsiasi momento il sistema fallisce e ha un arresto improvviso: 5. L’autore riavvia il software e richiede il ripristino dello stato precedente del sistema; 6. Il sistema ricostruisce lo stato precedente.   2a. Il sistema rileva delle anomalie che impediscono il ripristino:   1. Il sistema segnala un errore all’autore, registra l’errore, e passa in uno stato pulito;   L’autore inizia una nuova sottomissione. |
| **Requisiti speciali** | Internazionalizzazione della lingua sul testo visualizzato. |
| **Elenco delle varianti tecnologiche e dei dati** | Devono essere visualizzati gli articoli accettati o rifiutato. |
| **Frequenza di ripetizione** | Una volta l’ora circa. |
| **Varie** |  |

**UC6: Valutazione finale editore**

|  |  |
| --- | --- |
| **Nome del caso d’uso** | UC6: Valutazione finale editore |
| **Portata** | Applicazione TSR |
| **Livello** | Obiettivo **EDITORE** |
| **Attore primario** | Editore |
| **Parti interessate e interessi** | Editore: vuole prendere la decisione finale in maniera rapida ed efficiente. Vuole una visualizzazione chiara delle valutazioni attribuite all’articolo candidato dai revisori. Vuole un messaggio esplicativo che assicuri che la decisione finale è stata presa con successo.  Sistema: vuole memorizzare accuratamente le decisioni finale prese dall’editore.  Autore: vuole che la decisione finale presa sul suo articolo sia conforme alle valutazioni ad esso attribuite. |
| **Pre-condizioni** | L’articolo di cui va presa la decisione è stato candidato all’accettazione o al rifiuto. |
| **Garanzia di successo:** | Lo stato dell’articolo è aggiornato a “accettato”, “rifiutato”, oppure “da\_valutare” nel caso in cui l’editore decida di rimandare a giudizio l’articolo. |
| **Scenario principale di successo** | 1. L’editore accede al sistema; 2. L’editore inserisce e-mail e password; 3. Il sistema controlla la correttezza dei dati inseriti e autentica l’editore; 4. Il sistema mostra le sezioni rese disponibili per l’editore; 5. L’editore seleziona decisione finale; 6. Il sistema mostra l’elenco degli articoli candidati all’accettazione o al rifiuto; 7. L’editore seleziona un articolo; 8. Il sistema mostra le valutazioni attribuite all’articolo; 9. L’editore effettua la sua decisione; 10. Il sistema chiede una conferma; 11. L’editore conferma; 12. Il sistema salva la sua decisione; 13. Il sistema crea una traccia per la decisione finale; 14. Il sistema visualizza un messaggio di conferma di avvenuta decisione. |
| **Estensioni** | 1. L’editore inserisce una password errata: 2. L’editore non viene autenticato presso il sistema; 3. Viene visualizzato un errore esplicativo; 4. Viene fornita all’editore la possibilità di inserire nuovamente la password.   3a. L’editore ha dimenticato la password:   1. L’editore clicca su “password dimenticata”; 2. Il sistema visualizza un form per il recupero della password; 3. Il sistema richiede l’e-mail di registrazione; 4. L’editore inserisce l’e-mail e conferma; 5. Il sistema controlla che l’e-mail sia registrata; 6. Il sistema manda un e-mail, con i dati per il recupero della password;   6a. L’editore inserisce un’e-mail non registrato nel sistema:   1. Il sistema non manda l’e-mail per il recupero della password; 2. Viene visualizzato un errore esplicativo; 3. Il sistema torna alla pagina di autenticazione. 4. In qualsiasi momento il sistema fallisce e ha un arresto improvviso: 5. L’editore riavvia il software e richiede il ripristino dello stato precedente del sistema; 6. Il sistema ricostruisce lo stato precedente.   2a. Il sistema rileva delle anomalie che impediscono il ripristino:   1. Il sistema segnala un errore all’editore, registra l’errore, e passa in uno stato pulito;   L’editore inizia una nuova sottomissione. |
| **Requisiti speciali** | Internazionalizzazione della lingua sul testo visualizzato. |
| **Elenco delle varianti tecnologiche e dei dati** | Deve essere visualizzata la decisione finale presa. |
| **Frequenza di ripetizione** | Una volta l’ora circa. |
| **Varie** |  |

**UC7: Visualizzazione articoli**

1. L’editore accede alla sezione “Articoli candidati per l’accettazione”;

2. Il sistema mostra l’elenco degli articoli candidati per l’accettazione.

## 1.5 Documento di visione

È stato redatto in parallelo al documento di Ideazione un file di visione, aggiornato nelle successive iterazioni. Il file si chiama “Documento di visione” e si trova nella cartella “Documentazione completa”.

## 1.6 Regole di business

Vanno rispettate le seguenti regole di dominio:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Regola | Modificabilità | Sorgente |
| R1 | I giudizi dei revisori senior vengono tradotti in voti secondo questa equivalenza: accept = 9; weak accept = 6; weak reject = 4; reject = 0. | Bassa, la scala potrebbe variare solo nel caso in cui vengano introdotti nuovi giudizi. | Regola interna del sistema. |
| R2 | I revisori junior esprimono quattro voti da 0 a 9 su ciascuno dei seguenti criteri di valutazione: technical, originality, presentation, overall; il voto complessivo di un revisore senior è la media aritmetica dei quattro. | Bassa, la scala potrebbe variare solo nel caso in cui vengano introdotti nuovi criteri di valutazione. | Regola interna del sistema. |
| R3 | Un articolo viene candidato all’accettazione se la media ponderata del voto espresso dal revisore senior (peso 2) e della media aritmetica dei voti espressi da tutti i revisori junior (peso 1) è >= 4.5, altrimenti viene candidato al rifiuto. | Bassa, la politica di decisione potrebbe variare solo nel caso in cui vengano cambiate i metodi di valutazione dell’articolo. | Regola interna del sistema. |
| R4 | Un articolo viene valutato da un revisore senior e almeno due revisori junior. | Bassa, politica dell’editoria. | Regola interna del sistema. |

## 1.7 Specifiche supplementari

**Usabilità**

* L’interfaccia grafica deve essere intuitiva e facilmente comprensibile anche per utenti con scarse competenze informatiche;
* L’interazione col sistema deve essere semplice;
* Viene inserita la possibilità di scegliere fra due lingue, italiano e inglese, mediante interfaccia grafica.

**Affidabilità**

* Il software deve essere sviluppato in modo affidabile e deve mantenere i dati anche in casi di guasti;
* Per ciascuna attività interna al sistema deve essere mantenuta una traccia della data nella quale essa è avvenuta.

**Vincoli hardware e software**

* Non ci sono particolari requisiti per l’esecuzione del sistema, basta la presenza della JVM.
* È richiesta una connessione ad internet per eseguire le varie operazioni del sistema.

**Aspetti legali**

* Le tecnologie utilizzate per la realizzazione sono di tipo open source o freeware.
* TSRverrà rilasciato con licenza open source GPL v3.

## 1.8 Glossario

Vengono qui riportati i termini più significativi e le relative definizioni.

* Sottomissione: operazione mediante la quale un autore invia un articolo alla rivista.
* Attribuzione: operazione mediante la quale un revisore lascia una valutazione ad un articolo che gli è stato assegnato.
* Assegnazione: operazione interna al sistema mediante la quale per ogni articolo sottomesso, vengono scelti un revisore senior ed almeno due junior, che gli attribuiranno una valutazione.
* Articolo: elaborato prodotto da un autore e sottomesso alla rivista; esso viene assegnato a revisori che gli attribuiranno delle valutazioni.
* Valutazione: termine generale usato per indicare voti, giudizi e commenti.
* Autore: utente registrato che sottomette articoli che verranno sottoposti a valutazione per una loro eventuale pubblicazione nella rivista scientifica.
* Revisore senior: utente appartenente al comitato editoriale che riveste un ruolo di maggiore importanza; attribuisce un giudizio e un commento agli articoli che gli sono stati assegnati per la valutazione.
* Revisore junior: utente appartenente al comitato editoriale che riveste un ruolo di minore importanza; attribuisce un voto agli articoli che gli sono stati assegnati per la valutazione.
* Editore: persona che sceglie se rimandare a giudizio, accettare o rifiutare definitivamente un articolo, sulla base delle valutazioni espresse dai revisori.
* Giudizio: valutazione espressa dal revisore senior, può assumere i seguenti valori: technical, originality, presentation e overall.
* Commento: valutazione espressa dal revisore senior, in forma libera per un massimo di 1000 caratteri.
* Voto: valutazione espressa dal revisore senior, cifra compresa tra 0 e 9.
* Candidatura: operazione interna al sistema che, sulla base delle valutazioni ricevute dai revisori, propone all’editore l’accettazione o il rifiuto degli articoli.
* Accettazione: operazione mediante la quale l’editore accetta definitivamente un articolo.
* Rifiuto: operazione mediante la quale l’editore rifiuta definitivamente un articolo.
* Utente: termine generale usato per indicare autori e revisori.

# 2. Analisi orientata agli oggetti

## 2.1 Modello di Dominio

Ripercorriamo le varie fasi di modellazione che si sono succedute nelle varie iterazioni, al fine di giungere al Modello di Dominio definitivo al quale siamo pervenuti.

* **Iterazione 1**

Sono stati esaminati i casi d’uso UC3 e UC1, che riguardano la sottomissione di un Articolo da parte di un Autore, l’associazione dei Revisori ad un Articolo e la registrazione di quest’ultimo presso il sistema principale (TSR).

* **Iterazione 2**

È stato esaminato il caso d’uso UC4 ed è stato implementato anche il caso d’uso UC2 (analogo al caso d’uso UC1). Sono relativi all’attribuzione di una Valutazione da parte di un Revisore e alla registrazione di quest’ultimo presso il sistema.

* **Iterazione 3**

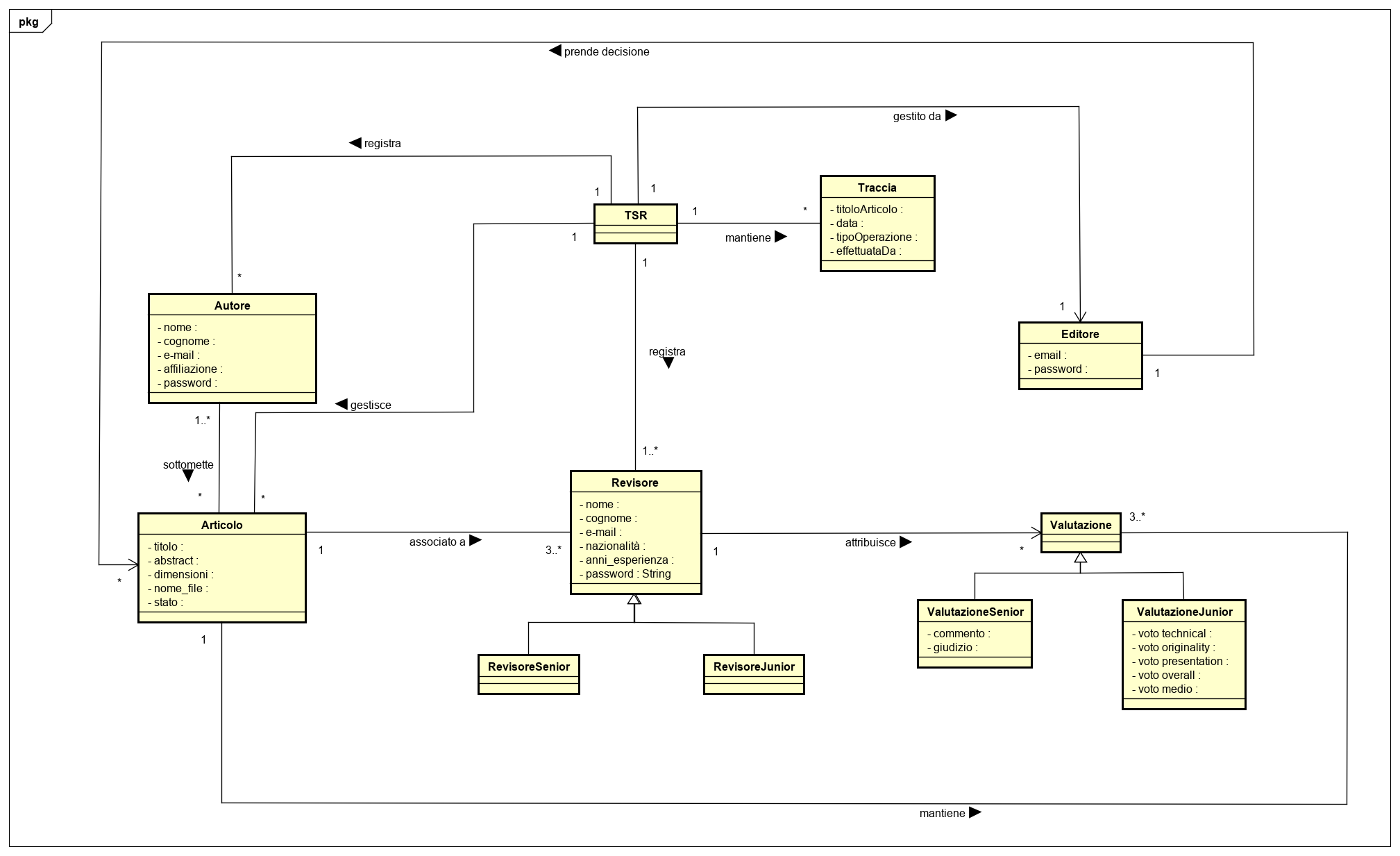
È stato esaminato il caso d’uso UC6, relativo alla decisione finale che l’Editore prende su un Articolo già valutato dai Revisori ad esso assegnati.

* **Iterazione 4**

È stato esaminato il caso d’uso UC5, relativo alla visualizzazione da parte degli Autori degli Articoli da loro sottomessi e per i quali l’Editore ha già preso la decisione finale.

Alla luce di ciò è stato elaborato il Modello di Dominio seguente, visionabile con maggiore chiarezza nel file Iterazione3\_04/Immagini/Modello\_Di\_Dominio\_03.png.

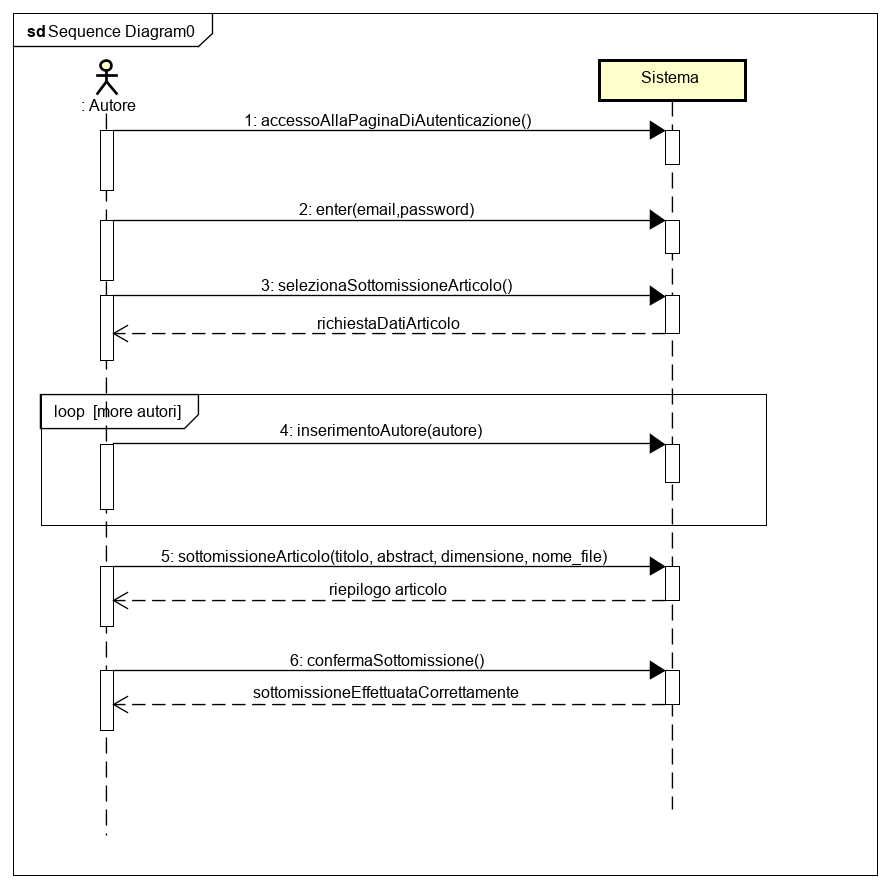
Le classi concettuali individuate sono: TSR (il sistema), Autore (utente che può sottomettere articoli), Articolo, Revisore (utente che attribuisce valutazioni), specializzato in RevisoreSenior e RevisoreJunior, Valutazione (attribuita da un Revisore ad un Articolo), anch’essa specializzata in ValutazioneSenior e ValutazioneJunior, Traccia (mantenuta da TSR per ogni operazione di sistema), Editore (prende la decisione finale su un Articolo).



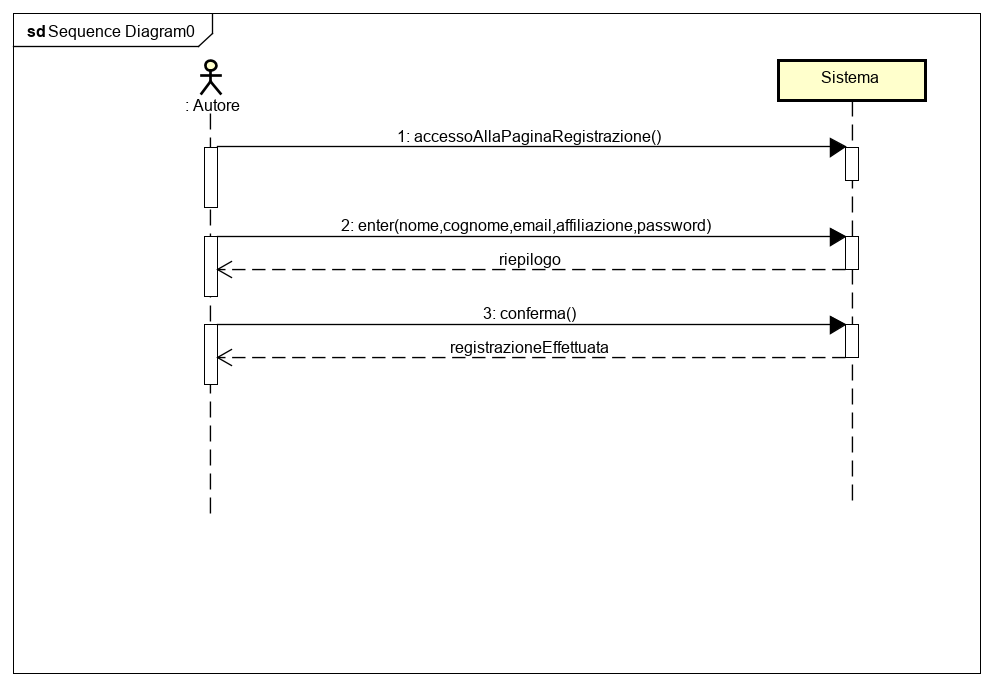
## 2.2 Diagrammi di sequenza di sistema e contratti delle operazioni

* **Iterazione 1**

Relativo al caso d’uso UC3:



Relativo al caso d’uso UC1:



**Contratto CO1: inserimentoAutore**

|  |  |
| --- | --- |
| Operazione | inserimentoAutore: autore: Autore |
| Riferimenti | Caso d’uso: Sottomissione Articolo |
| Pre-condizioni | L’autore è autenticato e ha selezionato l’operazione di Sottomissione Articolo; è stata creata un’istanza articolo corrente di Articolo. |
| Post-condizioni | È stato aggiunto un autore all’elenco di autori dell’articolo che l’autore sta sottomettendo. |

**Contratto CO2: sottomissioneArticolo**

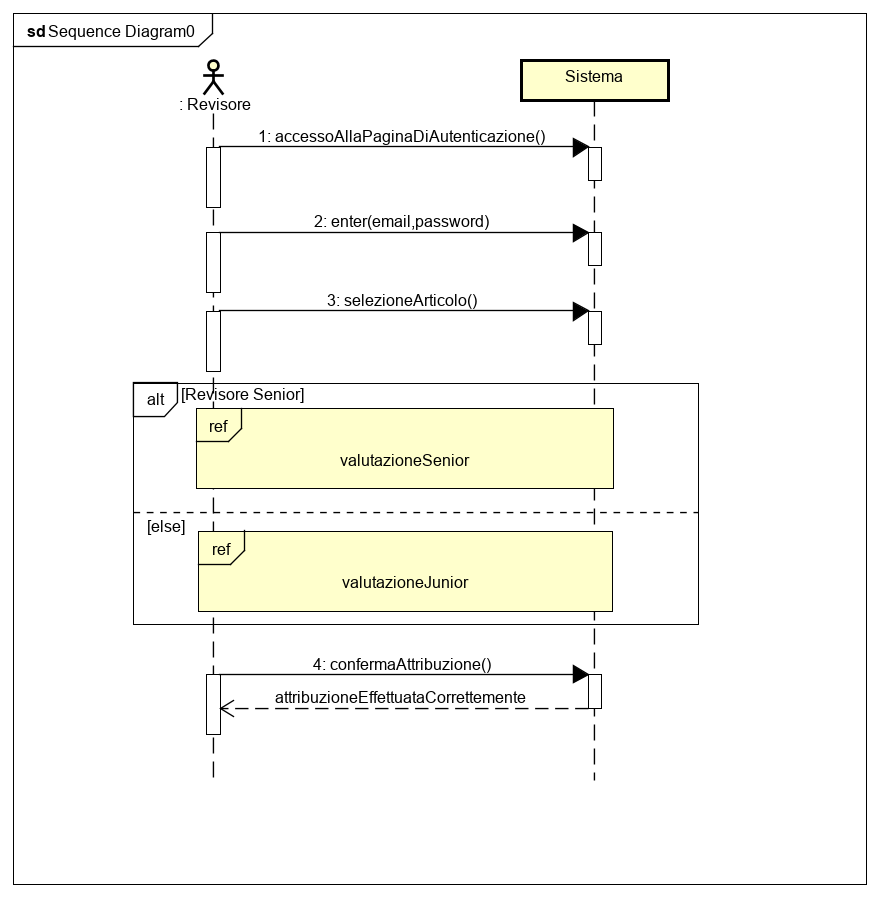
|  |  |
| --- | --- |
| Operazione | sottomissioneArticolo: titolo: String, abstract: String, dimensione: integer, nome\_file: String |
| Riferimenti | Caso d’uso: Sottomissione Articolo |
| Pre-condizioni | L’autore è autenticato e ha selezionato l’operazione di Sottomissione Articolo, l’autore ha inserito l’elenco degli autori. |
| Post-condizioni | Vengono aggiornati i dati relativi all’articolo corrente. |

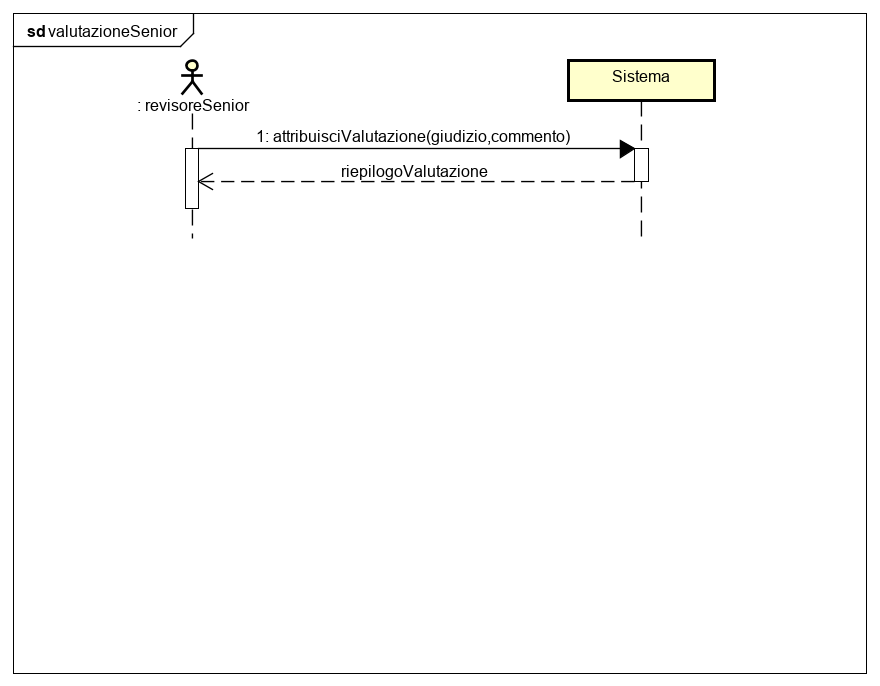
**Contratto CO3:** registrazione

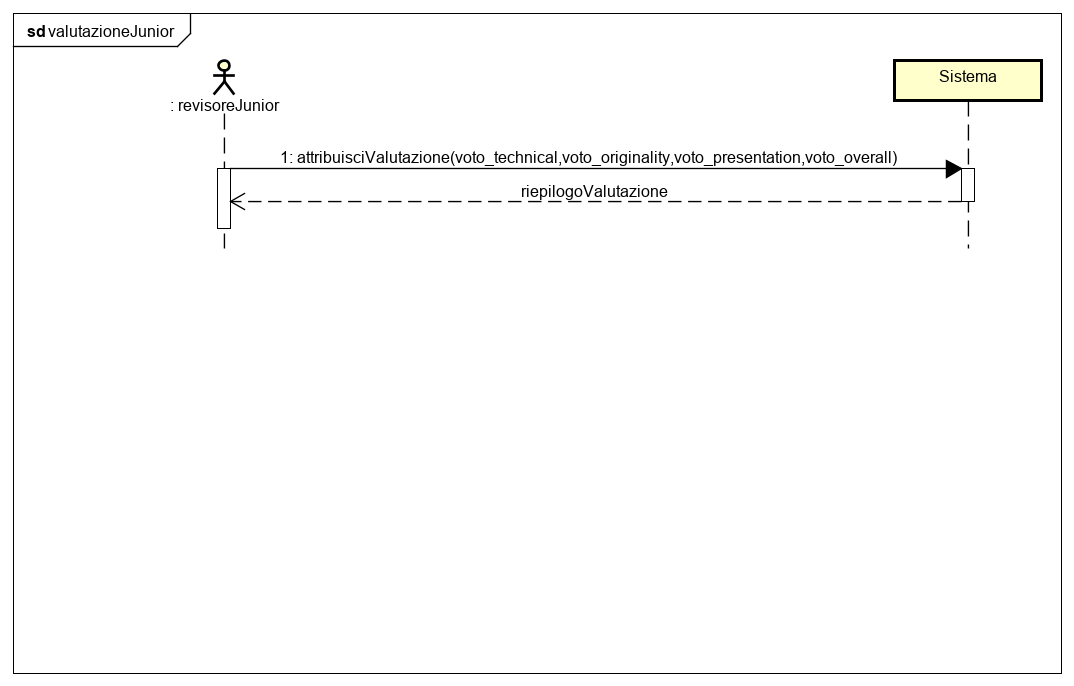
|  |  |
| --- | --- |
| Operazione | enter: nome: String, cognome: String, email: String, affiliazione: String, password: String |
| Riferimenti | Caso d’uso: Registrazione Autore |
| Pre-condizioni | L’autore ha selezionato l’opzione di registrazione |
| Post-condizioni | Viene creata un’istanza di autore con i dati inseriti. |

* **Iterazione 2**

Relativo al caso d’uso UC4:





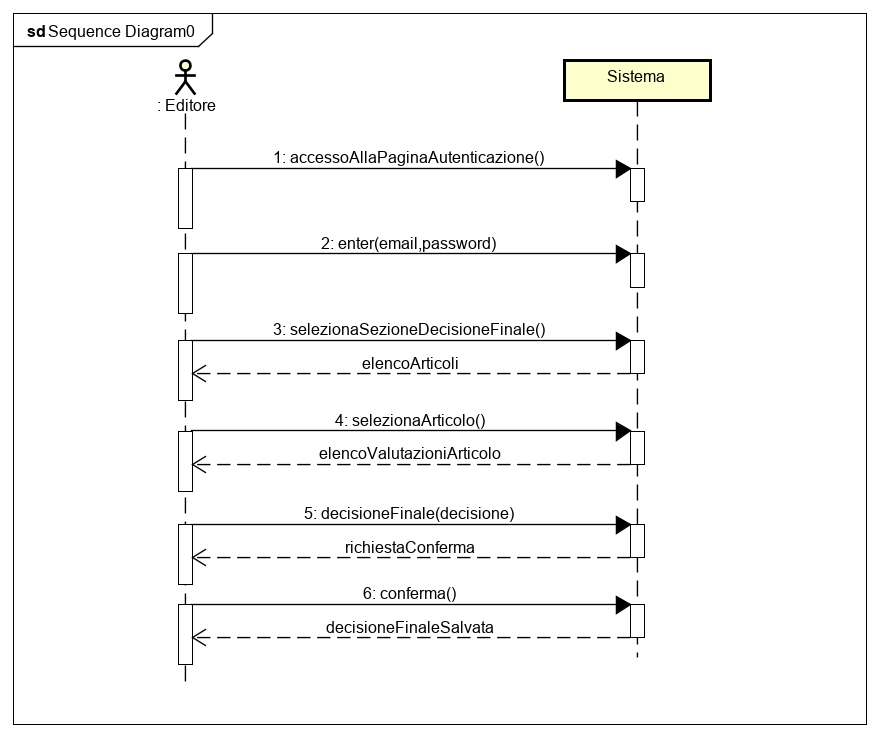


**Contratto CO4: attribuisciValutazione**

|  |  |
| --- | --- |
| Operazione | attribuisciValutazione: revisore: Revisore |
| Riferimenti | Caso d’uso: Attribuzione Valutazione |
| Pre-condizioni | Il revisore è autenticato e ha selezionato l’articolo da valutare fra quegli assegnatigli; è stata creata un’istanza valutazione corrente di Valutazione. |
| Post-condizioni | Sono stati aggiunti i dati inseriti dal revisore all’istanza di valutazione corrente. |

* **Iterazione 3**

Relativo al caso d’uso UC6:

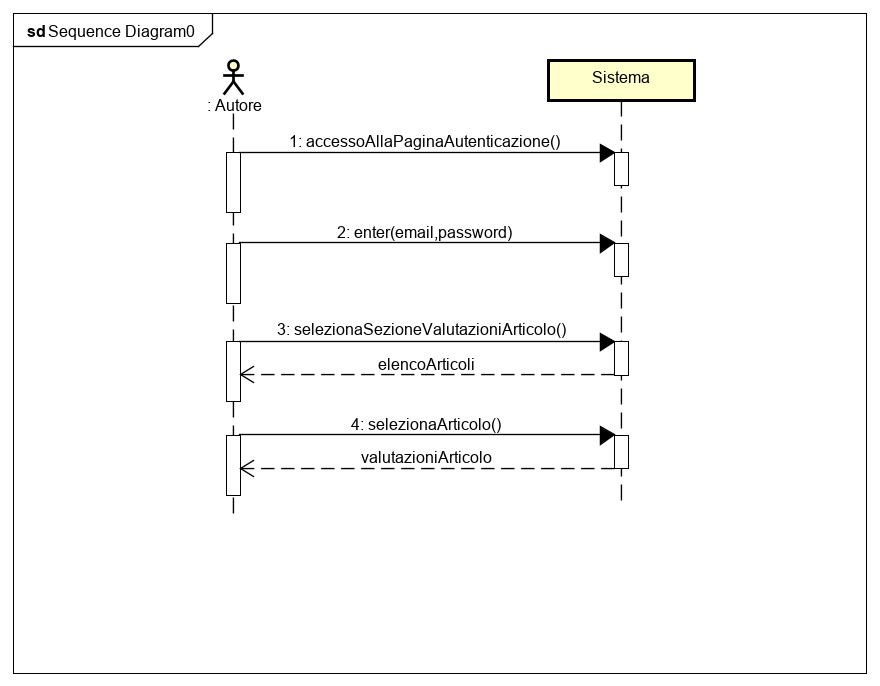


**Contratto CO5: decisioneFinale**

|  |  |
| --- | --- |
| Operazione | decisioneFinale: decisione: String |
| Riferimenti | Caso d’uso: Valutazione finale editore |
| Pre-condizioni | L’editore è autenticato e ha selezionato l’operazione di decisione finale; ha selezionato un articolo fra quelli candidati, assegnato ad un’istanza ArticoloCorrente di Articolo. |
| Post-condizioni | È stato aggiornato lo stato dell’ArticoloCorrente. |

* **Iterazione 4**

Relativo al caso d’uso UC5:



# 3. Progettazione orientata agli oggetti

## 3.1 Diagramma delle classi

Il diagramma delle classi finale si trova nel file “DCD\_Finale.png” della cartella Immagini della documentazione finale.

## 3.2 Diagrammi di sequenza

Essendo i diagrammi di sequenza numerosi e in gran parte difficilmente leggibili senza un adeguato zoom, si rimanda ai file contenuti nelle varie cartelle Immagini delle iterazioni:

* SD\_UC3\_nuovoArticolo.PNG (iterazione 1): creazione di una nuova istanza di Articolo da sottomettere.
* SD\_UC3\_addAutore.PNG (iterazione 1): aggiunta Autore all’istanza ArticoloCorrente di Articolo da sottomettere.
* SD\_UC3\_setDatiArticolo.PNG (iterazione 1): aggiunta degli altri campi all’istanza ArticoloCorrente di Articolo da sottomettere.
* SD\_UC3\_01\_confermaSottomissioneArticolo.png (iterazione 1): conferma della sottomissione dell’istanza ArticoloCorrente di Articolo.
* SD\_UC3\_associazioneRevisore.PNG 8iterazione 1): associazione dei Revisori ad un Articolo sottomesso.
* SD\_UC3\_01\_addTraccia.png (iterazione 1): aggiunta di un’istanza di Traccia nel sistema TSR.
* SD\_addValutazione.png (iterazione 2): attribuzione di una Valutazione da parte di un Revisore ad un Articolo.
* SD\_decisioneFinale.png (iterazione 3): decisione finale presa dall’Editore su un Articolo.
* SD\_Observer.png (iterazione 3): diagramma Observer che informa l’autore che un Articolo da lui sottomesso ha cambiato stato.
* SD\_getValutazioniArticolo.png (iterazione 4): visualizzazione da parte di un Autore delle Valutazioni degli Articoli da lui sottomessi che hanno ricevuto una decisione finale.

# 4. Testing

## 4.1 Introduzione

Il testing è una fase molto importante del processo di sviluppo di un programma. Permette di ridurre anche i costi di manutenzione del software. È difficile testare interamente un programma: non è possibile testare ogni possibile tupla di valori di inuput. Per questo motivo è importante avere un testing progettato accuratamente, in modo da individuare comportamenti anomali senza il bisogno di dover testare ogni possibile ingresso.

I test possono dividersi in due categorie:

* Test unitari (Unit Test): permettono di verificare la correttezza del codice in ogni sua parte. Ciò equivale a valutare ogni metodo scritto in Java, facendo in particolare uso di framework di automazione come JUnit.
* Test funzionali: verificano che il sistema nella sua interezza funzioni correttamente. Sono test a scatola nera, si danno degli input e si verifica la correttezza degli output ottenuti in risposta.

Sono stati scelti principalmente test di tipo unitario, con approccio Bottom Up: si è partiti dal collaudare metodi piccoli, andando via via verso metodi sempre più complessi e articolati.

## 4.2 Individuazione dei casi di test e Testing Unitario

In una prima fase di testing è stata affrontata una rilettura e discussione del codice sorgente al fine di individuare le classi ed i metodi da testare. È stata data priorità ai metodi che si occupano di creazione e inserimento di nuove istanze, di modifiche di istanze già create come l’aggiornamento dello stato di un Articolo. Per ciascuno sono state individuate classe di equivalenza per ridurre il numero di test al minimo necessario.

Più dettagliatamente:

1. Articolo:
   1. associaAutore: l’Autore aggiunto all’istanza ArticoloCorrente di Articolo deve essere registrato nel sistema TSR;
2. Autore:
   1. associaArticolo: l’Articolo associato deve essere presente nel sistema TSR;
3. TSR:
   1. registrazioneAutore, registrazioneRevisoreSenior, registrazioneRevisoreJunior: i campi devono essere inseriti, l’email è valida (contiene una @ e un .) e unica nel sistema, la password è valida (almeno 8 caratteri, una maiuscola, una minuscola e un numero);
   2. inValutazione: il Revisore che attribuisce una Valutazione ad un Articolo deve essere ad esso associato, la Valutazione deve contenere tutti i campi previsti;
   3. nuovoArticolo: i dati dell’Articolo devono essere inseriti in modo corretto;
   4. decisioneFinale: lo stato dell’Articolo è “candidato\_accettazione” o “candidato\_rifiuto”.

## 4.3 Test di Sistema

È stato effettuato un test di sistema complessivo per verificare tutti i vari casi d’uso. Sono emersi i seguenti errori: durante il calcolo del voto medio per candidare un articolo all’accettazione o al rifiuto non veniva presa in considerazione la prima valutazione inserita; il sistema permetteva di sottomettere articoli con campi incompleti; il sistema non prendeva in considerazione l’ipotesi (anche se inverosimile nella realtà) che ci fossero meno revisori registrati al sistema del numero di revisori sorteggiati per essere associati ad un articolo; il sistema non controllava correttamente l’email. È stato anche effettuato un refactoring del metodo addTraccia di TSR, cambiandone i parametri al fine di avere un unico metodo per le varie operazioni che il sistema intende conservare.

# 5. Persistenza

Alla fine è anche stato implementato un piccolo database che non si propone di mantenere persistenza di tutte le classi che compongono il software, ma ha lo scopo di sperimentare alcune funzionalità CRUD del sistema. In particolare, un’interfaccia AutoreDAO definisce due metodi, inserisciAutore e doRetrieveAll. La classe AutoreDAOimpl implementa quest’interfaccia. Il metodo inserisciAutore viene richiamato quando un Autore si registra al sistema (caso d’uso UC1): l’istanza di AutoreCorrente viene passata al metodo, che va a fare una insert scritta in linguaggio SQL nel database databaseTSR. Il metodo doRetrieveAll, invece, viene richiamato all’avvio dell’applicazione, e restituisce a TSR l’elenco degli autori registrati al sistema, ciascuno con la propria lista di articoli associati. Non vengono invece caricate le valutazioni di ciascuno d’essi, secondo il principio Lazy Load.