*Progetto Ingegneria del Software*

**TSR**

**Elaborazione – Iterazione 1**



Prof. Orazio Tomarchio

Gianluca Cristaudo

Samuele Gulino

Ingegneria Informatica Magistrale LM-32 2019/2020

Sommario

[1 Introduzione 3](#_Toc35464515)

[2 Analisi orientata agli oggetti 3](#_Toc35464516)

[2.1 Aggiornamento caso d’uso UC3 3](#_Toc35464517)

[2.2 Modello di dominio 5](#_Toc35464518)

[2.3 Diagrammi di sequenza di sistema 7](#_Toc35464519)

[2.4 Contratti delle operazioni 8](#_Toc35464520)

[3 Progettazione orientata agli oggetti 9](#_Toc35464521)

[3.1 Diagrammi di sequenza 9](#_Toc35464522)

[3.2 Diagramma delle classi 12](#_Toc35464523)

[4 Caso d’uso d’avviamento: UC1 12](#_Toc35464524)

[4.1 Scenario principale di successo 12](#_Toc35464525)

[4.2 Modello di dominio 13](#_Toc35464526)

[4.3 Diagrammi di sequenza di sistema 13](#_Toc35464527)

[4.4 Regole di business 14](#_Toc35464528)

[4.5 Contratto delle operazioni 14](#_Toc35464529)

[4.6 Diagramma delle classi 14](#_Toc35464530)

[5 Implementazione 15](#_Toc35464531)

# Introduzione

Dopo la fase di ideazione, segue quella di elaborazione. Lo scopo delle varie iterazioni che verranno svolte sarà di migliorare il documento di Visione, di implementare il nucleo dell’architettura del software, risolvere le problematiche relative ai rischi maggiori, identificare la maggior parte dei requisiti e la portate, fornire stime del carico di lavoro complessivo e delle risorse necessarie.

Durante la prima iterazione i requisiti principali sui quali concentrarsi sono l’implementazione di uno dei principali scenari di successo, lo scenario UC3 (Sottomissione Articolo) e del caso d’uso UC1 (Registrazione autore) che rappresenta un caso d’uso di avviamento.

# Analisi orientata agli oggetti

L’analisi orientata agli oggetti (OOA) si basa sull’analisi dei concetti del dominio con un approccio a oggetti. Vengono utilizzati diversi strumenti: il Modello di dominio, i diagrammi di sequenza di sistema, i contratti delle operazioni.

## Aggiornamento caso d’uso UC3

Relativamente al caso d’uso UC3 durante l’analisi orientata agli oggetti, in particolare durante la stesura di diagrammi di sequenza e contratti delle operazioni, ci è sembrato opportuno fare alcune significative modifiche allo scenario principale di successo. Innanzitutto abbiamo considerato che l’autore ha possibilità, oltre che di sottomettere articoli, di visionare le valutazioni ricevute per gli articoli già sottomessi e valutati. Pertanto il sistema dovrà mostrare un’interfaccia che consente di scegliere quale delle due operazioni e1ffettuare. Come seconda cosa, la possibilità di poter avere più autori che collaborano allo stesso articolo ci ha posto di fronte all’esigenza di dover predisporre un modo per inserire questi ultimi (possono essercene un numero indefinito, come si evince dalla molteplicità del ruolo nell’associazione fra Autore e Articolo nel Modello di Dominio). Viene fatta l’assunzione che l’autore che effettua la sottomissione sia un autore dell’articolo, quindi l’autore non deve inserire sé stesso: inserisce solo altri autori, se presenti. Oltre a questo, ci siamo preoccupati dell’assegnazione dei revisori all’articolo. Nella nostra analisi, è stata considerata un’operazione che esegue il sistema in automatico appena un articolo viene sottomesso. L’operazione sarà più dettagliatamente chiarita nel relativo contratto. Infine, ci sembra opportuno eliminare l’attributo username relativo all’autore, considerato ridondante, dato che l’e-mail (richiesta dal sistema in fase di registrazione, vedi caso d’uso UC1) è univoca; possiamo pertanto utilizzare quest’ultima come identificatore dell’autore.

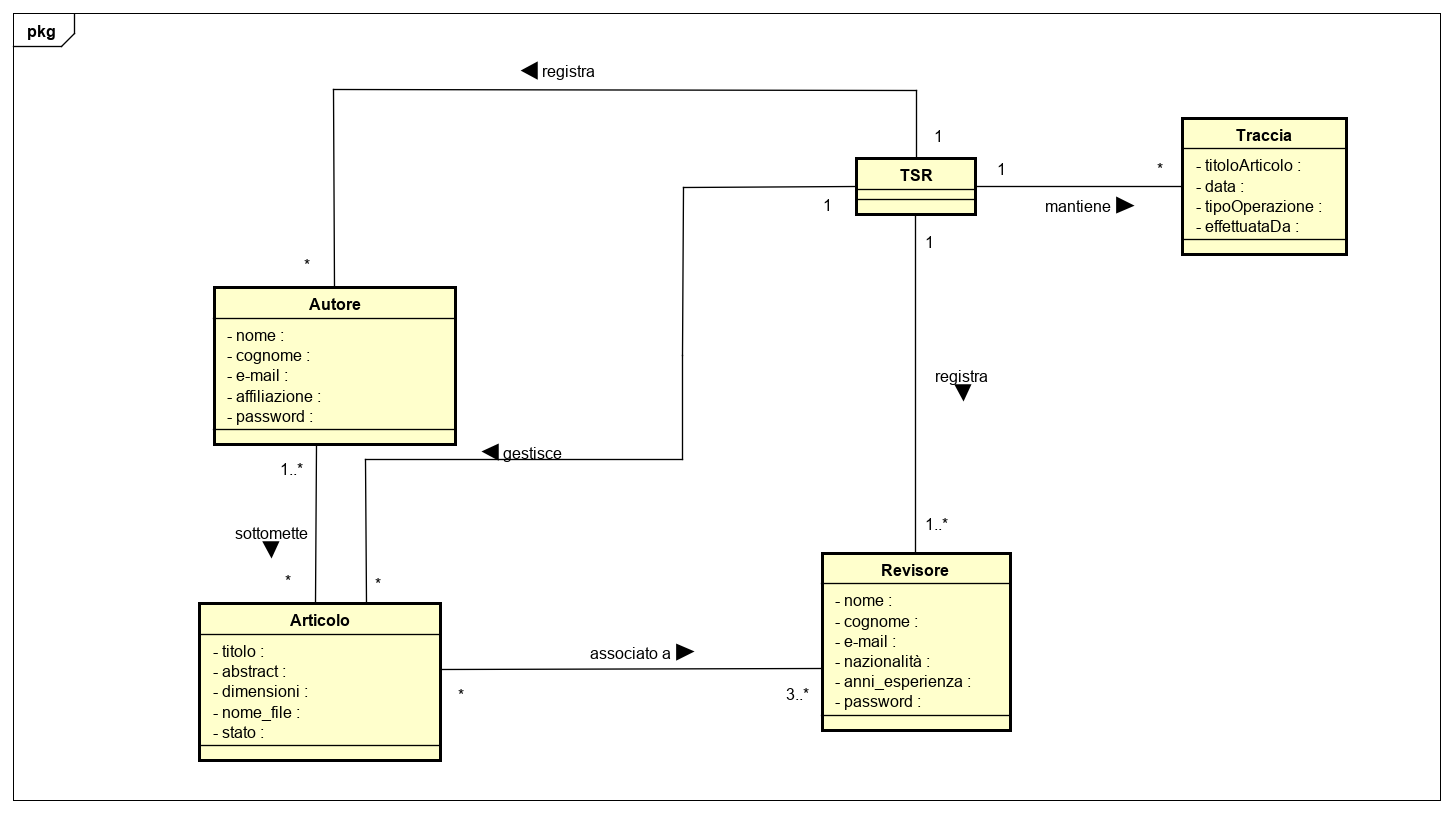
|  |  |
| --- | --- |
| **Scenario principale di successo** | 1. L’autore accede al sistema; 2. L’autore inserisce e-mail e password; 3. Il sistema controlla la correttezza dei dati inseriti; 4. Il sistema mostra le sezioni rese disponibili per l’autore. 5. L’autore seleziona sottomissione articolo. 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. |

Le stesse considerazioni sull’e-mail si applicano alle estensioni individuate in fase di Ideazione.

## Modello di dominio

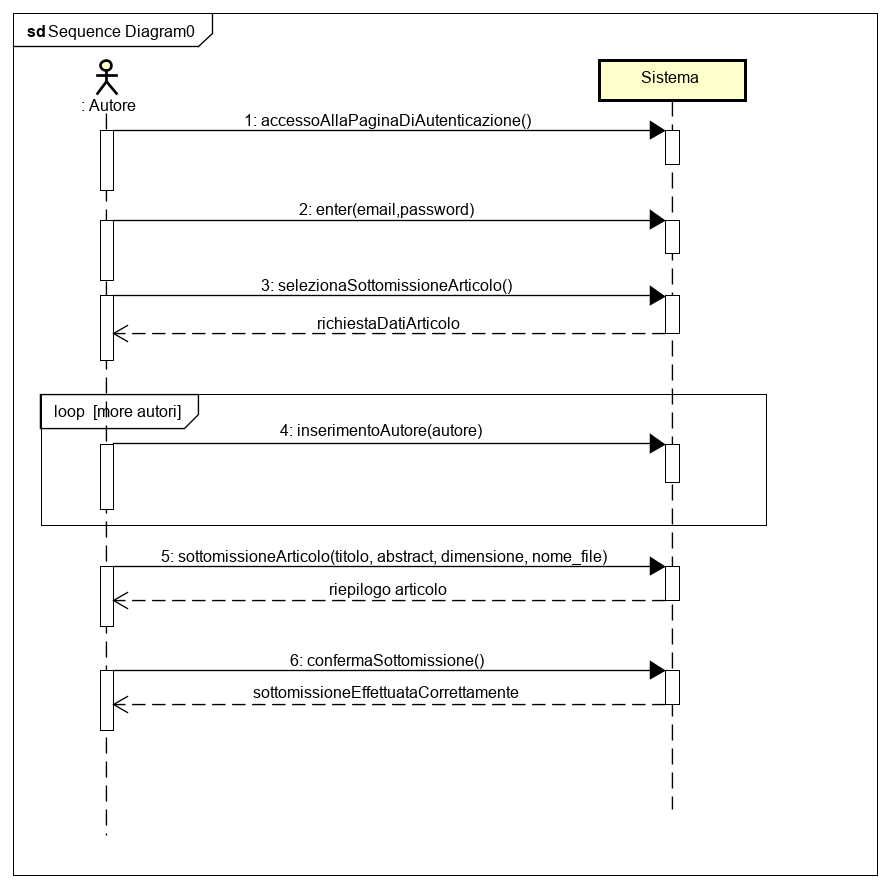
Nell’ambito di Unified Process la disciplina che si occupa di fornire dettagli sul dominio si chiama Modellazione del Business. Dopo aver analizzato in maniera dettagliata il caso d’uso UC3 sono state individuate le seguenti classi concettuali:

* TSR: rappresenta il sistema Top Scientist Review;
* Autore: utilizzatore del software con possibilità di sottomettere articoli;
* Articolo: rappresenta l’articolo sottomesso dall’autore;
* Revisore: rappresenta un utente del sistema appartenente al comitato editoriale associato all’articolo;
* Traccia: rappresenta la traccia che il sistema lascia quando effettua sottomissione e associazione.

È stato elaborato, sulla base di classi, attributi e associazioni, il seguente Modello di Dominio.

## Diagrammi di sequenza di sistema

Il prossimo passo relativo all’analisi orientata agli oggetti è la stesura di SSD per il principale scenario di successo del caso d’uso UC3 considerato. Nel nostro caso non intervengono sistemi esterni, abbiamo solo l’attore principale e il sistema TSR.



## Contratti delle operazioni

Definiamo a questo punto i contratti relativi alle due operazioni principali del diagramma di sequenza di sistema, ovvero l’inserimento di un autore e la sottomissione dell’articolo.

**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. |

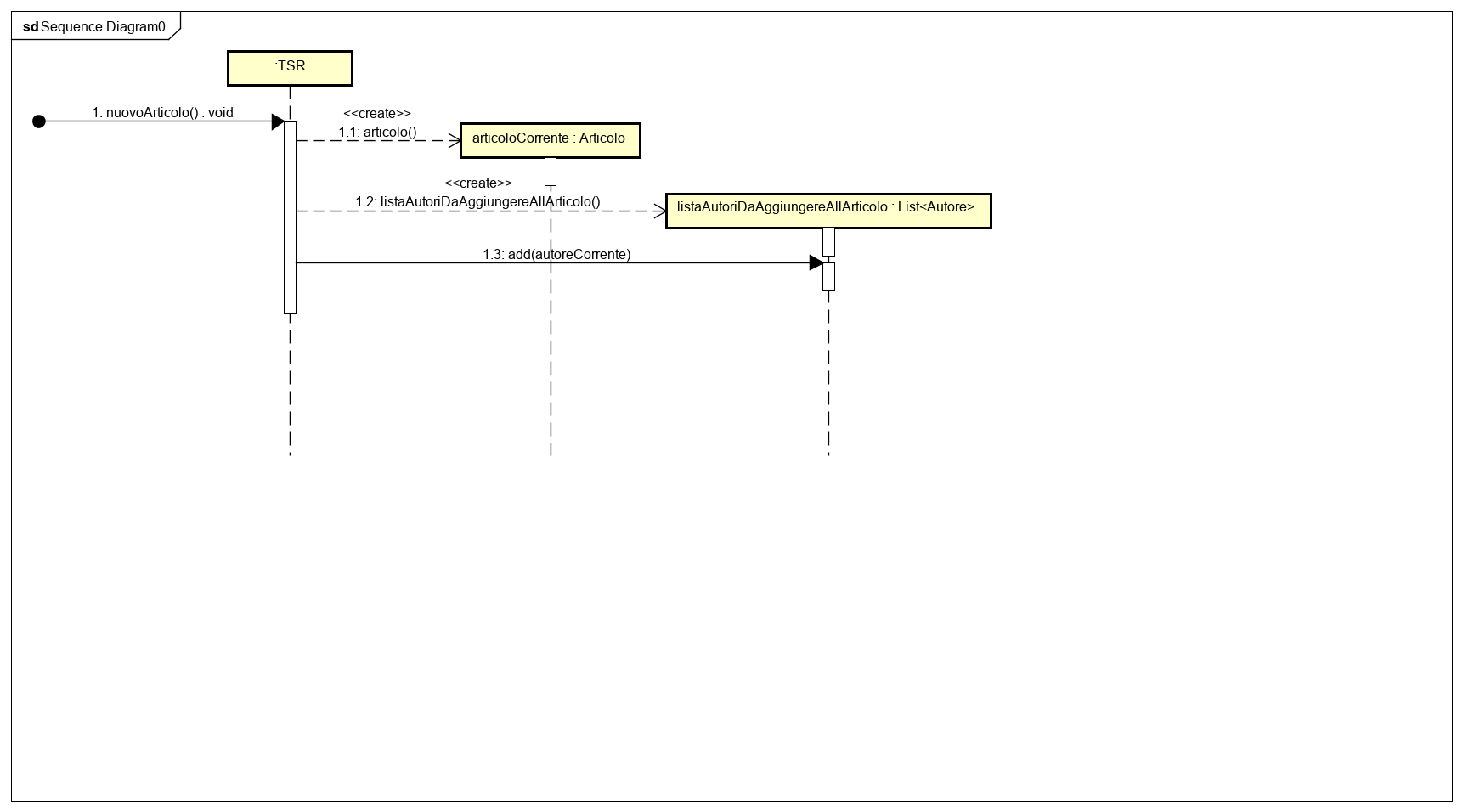
# Progettazione orientata agli oggetti

La progettazione orientata agli oggetti (OOD) è la disciplina di Unified Process che si occupa della definizione di oggetti software e loro responsabilità, in relazione al modello concettuale di dominio elaborato. Verrà definito in questa fase il Modello di Progetto, che comprende i diagrammi di interazione (che mostrano la dinamica del sistema) e quelli delle classi (che mostrano invece la parte statica).

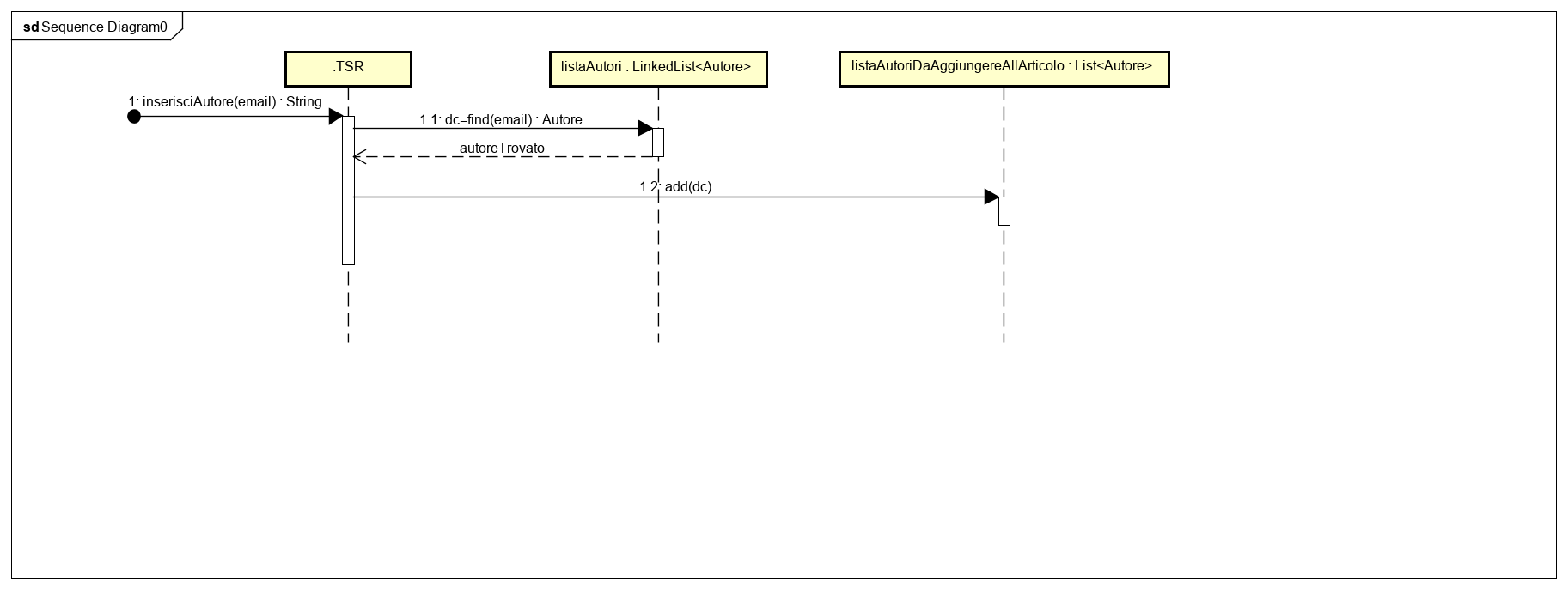
## Diagrammi di sequenza

Mostriamo innanzitutto i diagrammi di interazione. In particolare sono stati tracciati diversi diagrammi di sequenza relativi alle varie fasi e operazioni del del caso d’uso.

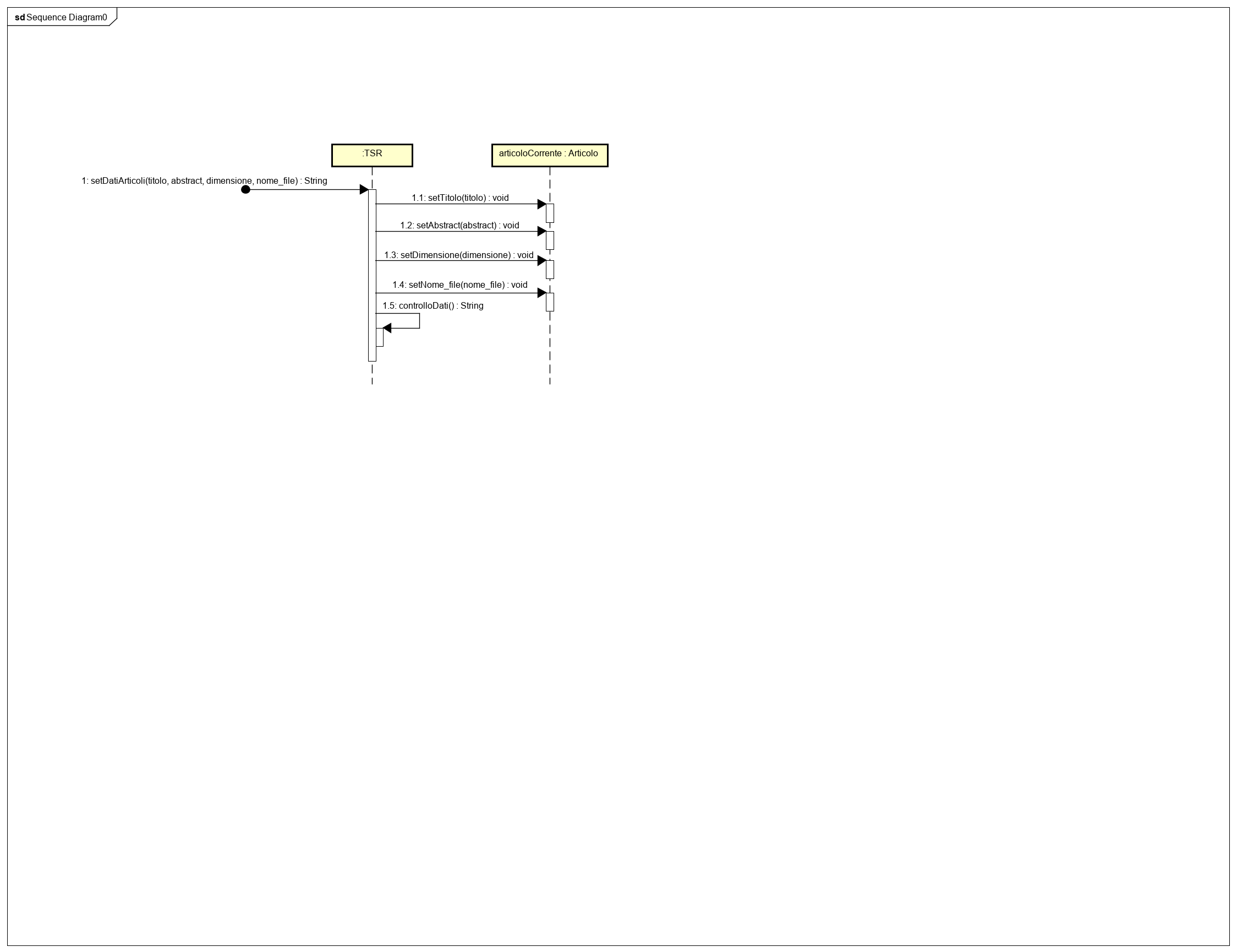
Per primo presentiamo la creazione di una nuova istanza di Articolo. Il messaggio trovato indica che deve essere effettuata una sottomissione: pertanto la prima cosa da fare è creare una nuova istanza della classe Articolo, chiamata articoloCorrente. Viene creata anche una lista di autori da aggiungere all’articolo: l’autore che effettua la sottomissione viene subito aggiunto ad essa.



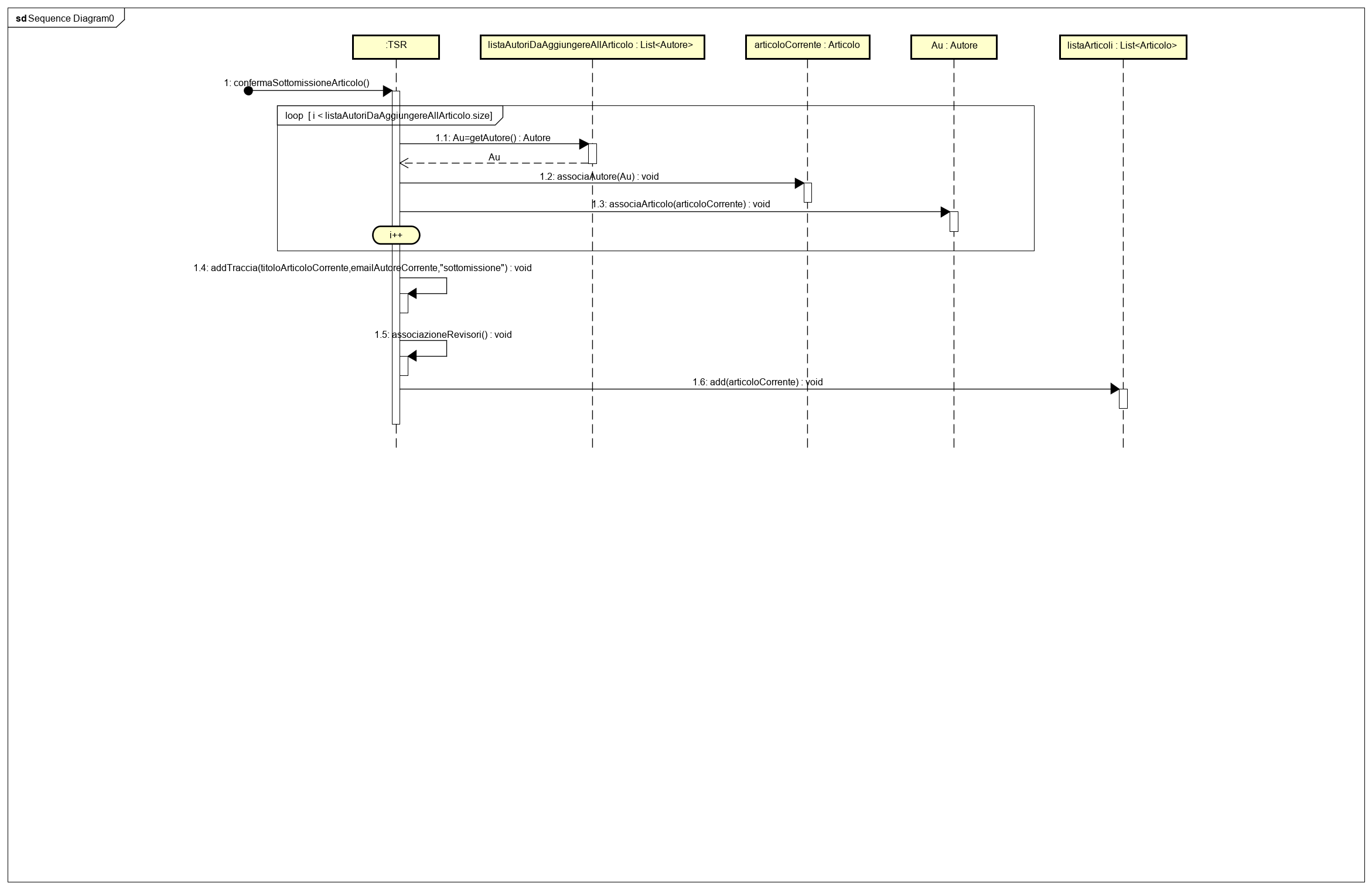
Il passo successivo è quello di aggiungere gli autori alla lista.



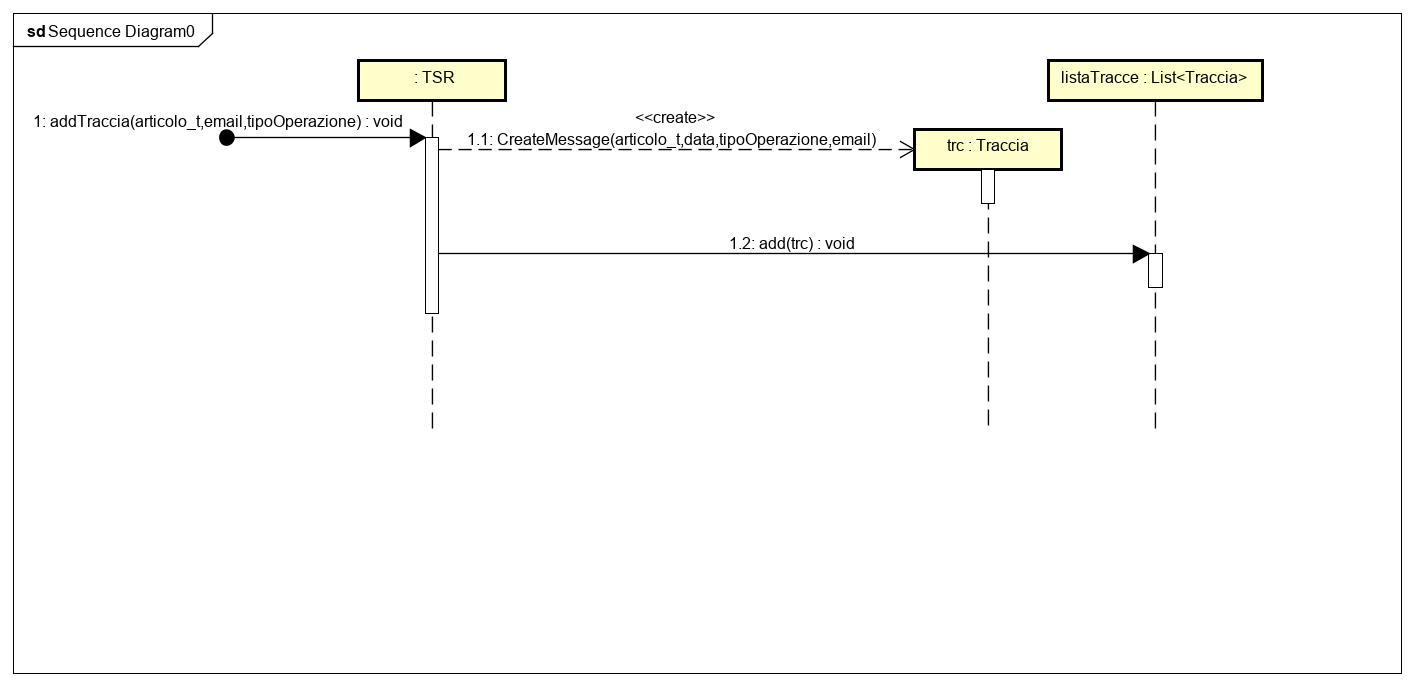
Una volta aggiunto l’elenco di autori, si procede con l’inserimento degli altri dati relativi all’articolo ed il controllo della loro correttezza.



Viene quindi chiesta una conferma della volontà di sottomettere l’articolo. Se avviene la conferma, viene effettuata l’associazione degli autori inseriti nella lista all’articolo corrente, e anche l’associazione opposta, ovvero quella dell’articolo a ciascun autore. Questa ridondanza è giustificata dal minore tempo di risposta e una più semplice implementazione di altre funzionalità del sistema, come quella del caso d’uso UC6. L’articolo viene aggiunto alla lista articoli mantenuta dal sistema.



Viene mantenuta una traccia della sottomissione.



Infine, contestualmente all’operazione di sottomissione, vengono associati i revisori che dovranno attribuire la valutazione. Il sistema associa un revisore senior e un numero random maggiore o uguale di 2 di revisori junior (nell’implementazione si è scelto per semplicità compreso fra 2 e 5, ma non è stato inserito nel modello di dominio dato che è una nostra semplificazione implementativa ma non presente nei requisiti). I revisori mantengono una lista degli articoli a loro associati. Anche per le associazioni vengono create delle tracce nel sistema. Il diagramma di sequenza, per ragioni di leggibilità, si trova nella cartella immagini dell’iterazione corrente, nel file “SD\_UC3\_associazioneRevisore.PNG”.c

## Diagramma delle classi

Visto il Modello di Dominio e i diagrammi di sequenza riportati, è stato elaborato il Diagramma delle Classi che si trova per maggiore chiarezza nel file “DCD\_it1\_1.0.png” all’interno della cartella Immagini dell’iterazione corrente. TSR si configura sin da subito come una classe *Singleton*. Si è scelto di inserire una gerarchia relativa alle classi Revisore già alla prima iterazione, per poter associare correttamente revisore senior e junior agli articoli. Questo concetto verrà ripreso, definito meglio e ampliato all’iterazione successiva, inserendo la generalizzazione anche nel Modello di Dominio concettuale e dandole connotati maggiormente delineati.

# Caso d’uso d’avviamento: UC1

Vediamo adesso brevemente il caso d’uso UC1, Registrazione Autore, che possiamo considerare un caso d’uso d’avviamento, dato che ci consente di avere dei dati iniziali di alcuni autori registrati al sistema che possono effettuare delle sottomissioni. Tramite questi dati possiamo anche effettuare dei test della nostra implementazione tramite software di questa iterazione.

## 4.1 Scenario principale di successo

Riproponiamo innanzitutto lo scenario principale di successo, effettuando alcuni aggiornamenti sulla base delle precedenti analisi OOA e OOD. Si è scelto di utilizzare l’e-mail come identificativo univoco, quindi non è necessario avere uno username extra.

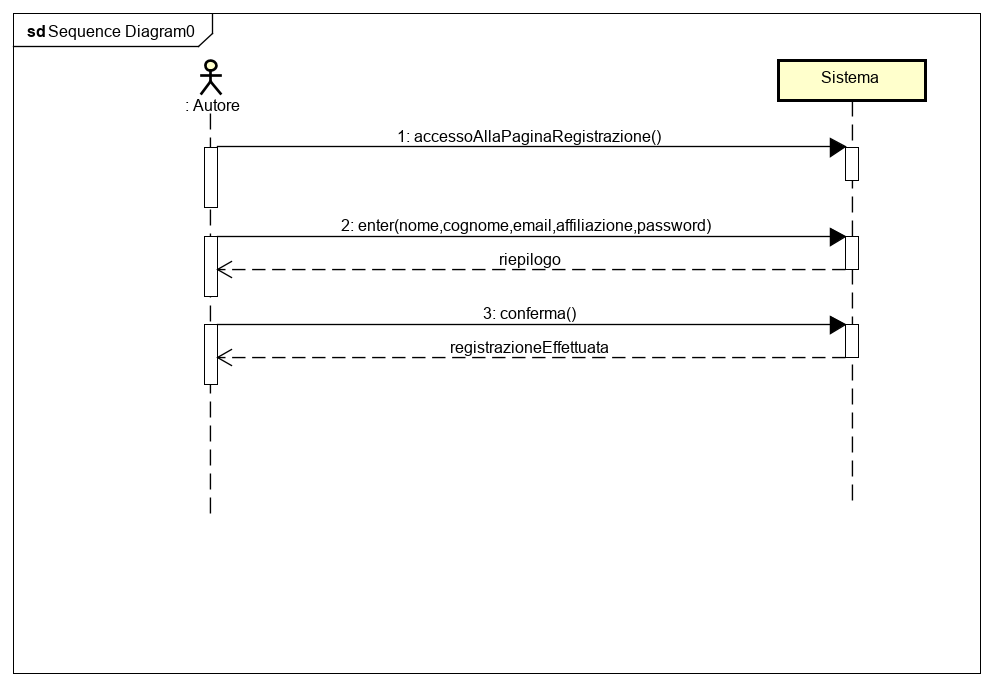
|  |  |
| --- | --- |
| **Scenario principale di successo** | 1. L’autore accede al sistema; 2. L’autore (non ancora registrato) accede alla pagina di registrazione; 3. L’autore inserisce i propri dati all’interno dei vari form richiesti (nome, cognome, indirizzo e-mail, affiliazione);   4. L’autore 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. |

## 4.2 Modello di dominio

Il Modello di Dominio non ha bisogno di variazioni rispetto a quello già elaborato nel corso di questa iterazione. Le uniche classi concettuali che prendono parte alla registrazione dell’autore sono il sistema TSR e l’Autore, già presenti nel modello sopra riportato.

## 4.3 Diagrammi di sequenza di sistema

Analizziamo l’SSD del nostro scenario principale di successo, che coinvolge come attori l’Autore e il Sistema.



## 4.4 Regole di business

Nel principale scenario di successo abbiamo parlato in maniera generica di vincoli imposti per la validità di una password. Definiamo qui una regola di business a tal proposito. Nella cartella Iterazione1\_02 è possibile trovare il documento di Ideazione\_2.0 aggiornato con questa regola ulteriore, non precedentemente definita.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Regola | Modificabilità | Sorgente |
| R5 | Una password è valida se rispetta i seguenti vincoli: lunghezza minima 8 caratteri; almeno una lettera maiuscola e una minuscola; almeno un numero. | Non si prevedono modifiche. | Regola interna del sistema. |

## 4.5 Contratto delle operazioni

Definiamo un contratto per l’operazione di registrazione.

**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. |

## 4.6 Diagramma delle classi

Sono stati aggiunti i metodi controllaPassword() e registrazioneAutore() alla classe TSR. Il diagramma si può trovare nel file “DCD\_it1\_1.1.png” nella cartella Immagini dell’iterazione corrente (Iterazione1\_02).

# Implementazione

È fornito il codice relativo all’implementazione dei due casi d’uso UC1 e UC3.