UNIVERSITÀ
DI TRENTO

Dipartimento di Ingegneria
e Scienza dell'Informazione

Ingegneria del Software

Grab a Book

Gruppo G23

Anno Accademico 2023/2024

Indice

Scopo del documento

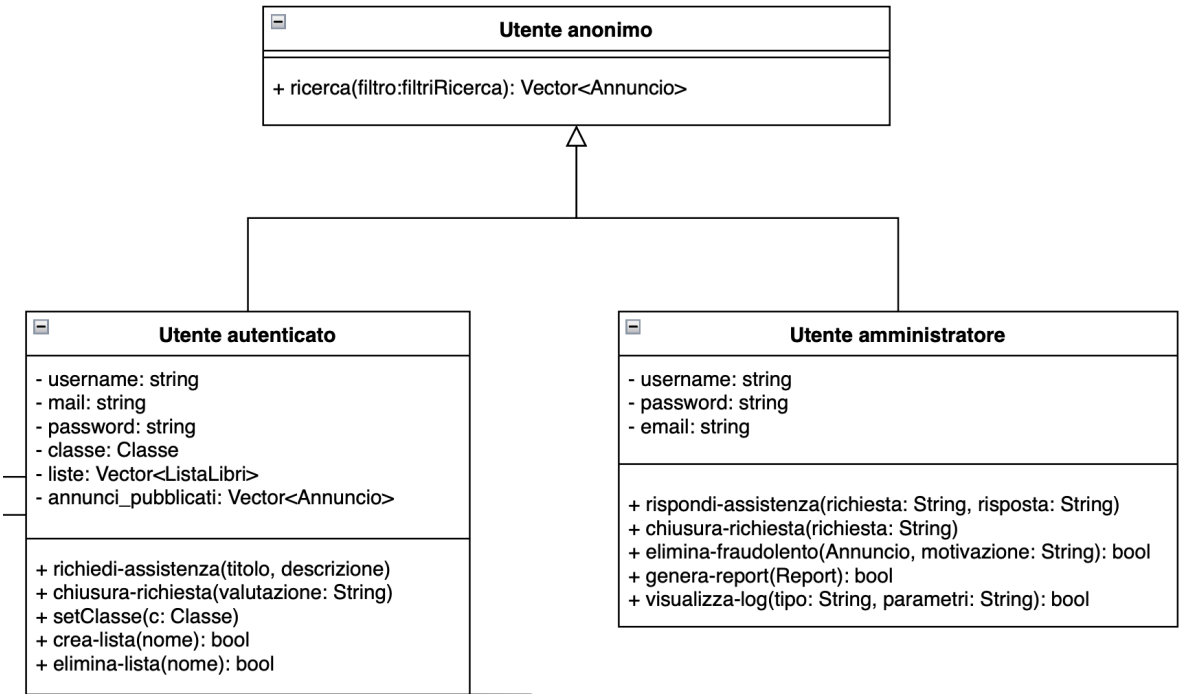
Il presente documento riporta la definizione del progetto Grab a Book usando diagrammi delle classi in Unified Modeling Language (UML) e codice in Object Constraint Language (OCL). Nel documento precedente sono stati presentati il diagramma degli use case, il diagramma di contesto e il diagramma dei componenti.
Ora viene definita l'architettura del sistema dettagliando le classi che dovranno essere implementate e la logica che regola il comportamento del software. Le classi saranno rappresentate con un diagramma delle classi in UML, mentre la logica in OCL. Quest'ultimo viene utilizzato per esprimere dei concetti non esprimibili in nessun altro modo formale nel contesto di UML.

Diagramma delle classi

In questo capitolo vengono presentate le classi previste nell'ambito del progetto Grab A Book. Le classi individuate sono caratterizzate da un nome, una lista di attributi che identificano i dati gestite dalla classe e una lista di metodi che consentono le operazioni previste dalla classe.
Ogni classe poi ha delle associazioni con altre classi, e tramite queste è possibile fornire informazioni su come le classi interagiscono fra loro.
Riportiamo di seguito le classi individuate a partire dai diagrammi di contesto e dei componenti.

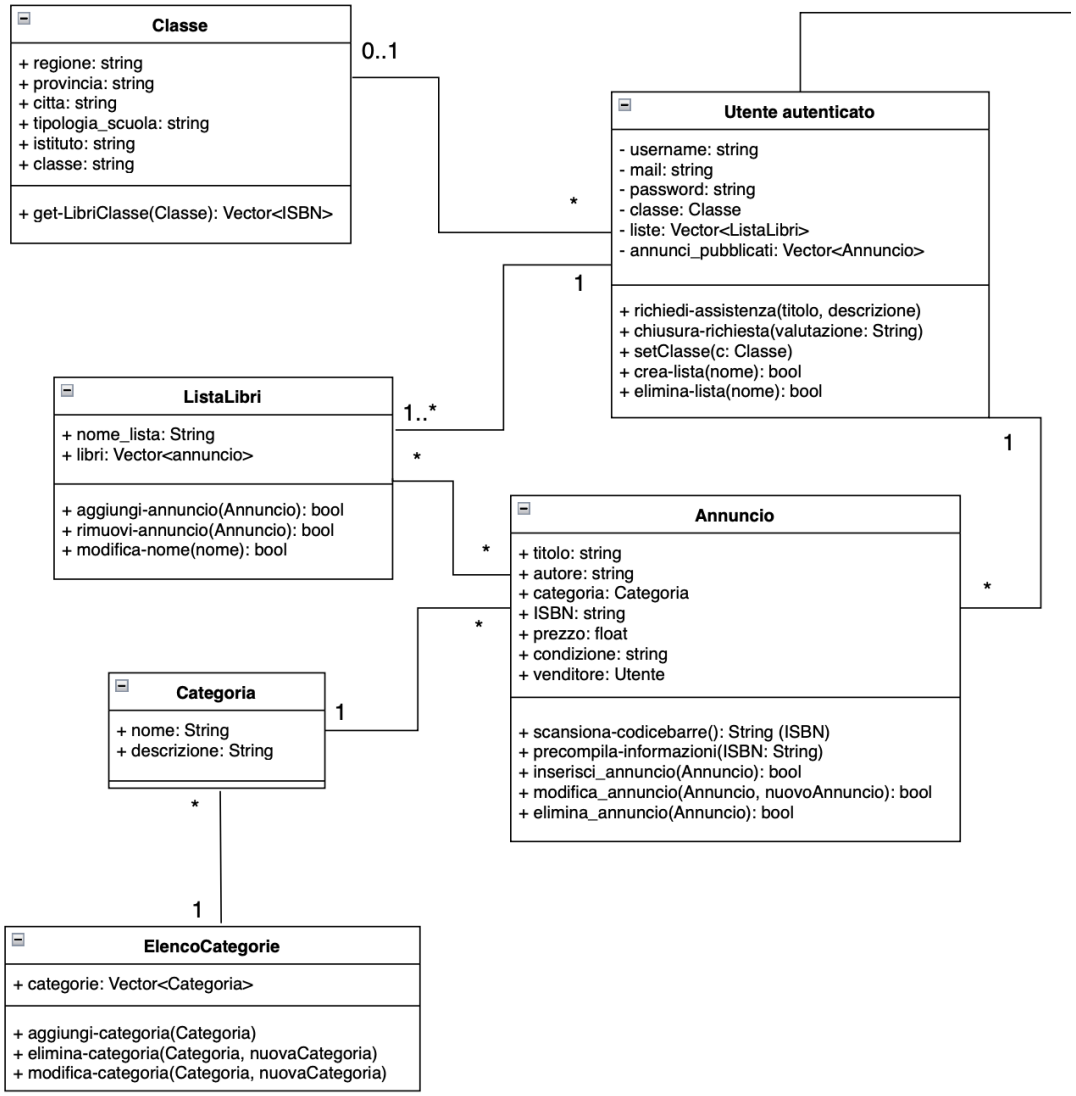
Utenti

Analizzando i 3 diagrammi di contesto proposti nel documento precedente (ma anche i requisiti funzionali) sono state individuate tre tipologie di utenti: Anonimo, Autenticato e Amministratore.
L'utente anonimo è colui che non ha ancora effettuato il login o la registrazione ma può comunque navigare sull'home-page.
Tutte le tipologie di utente hanno in comune la possibilità di effettuare una ricerca attraverso dei filtri specificati (titolo del libro.. categoria), quindi sono state individuate 3 classi, collegate fra loro attraverso la generalizzazione.
Hanno inoltre ruoli e funzioni diverse tra loro.



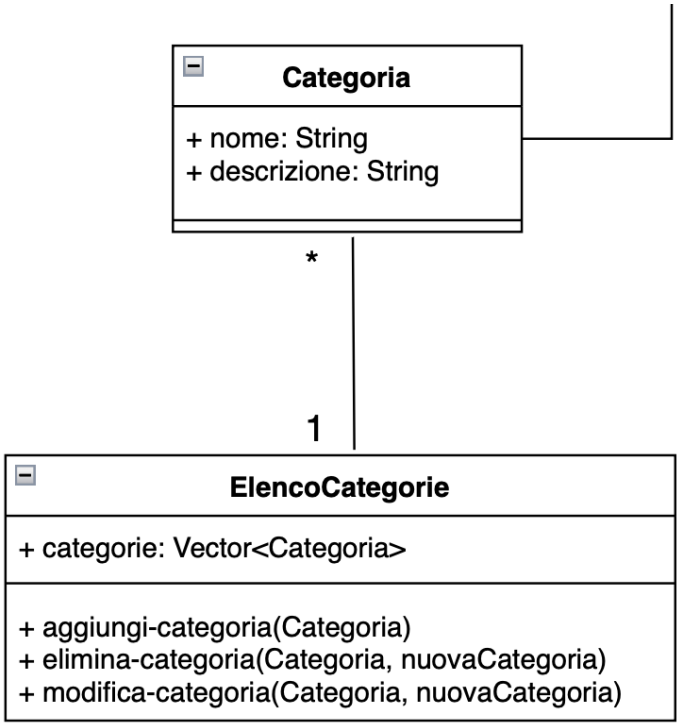
Utente autenticato

L'utente anonimo può ovviamente effettuare il login (descritto in seguito, in Autenticazione) oppure la registrazione. Una volta effettuata una delle precedenti operazioni, l'utente ha a disposizione molte funzionalità, come la richiesta d'assistenza, l'impostazione della classe e la pubblicazione di un annuncio. Può inoltre salvare vari annunci tramite un sistema di liste. Di seguito quindi sono mostrate tutte le classi collegate a questo tipo di Utente, che permettono di mostrare le funzionalità a cui ha accesso.



Categoria ed Elenco Categorie

Nell'aggiunta di un annuncio, tra le varie informazioni da aggiungere è presente la categoria del libro che si sta provando a vendere.
Sono state quindi identificate due classi, Categoria la quale è identificata attraverso il nome e la descrizione, e un ElencoCategoria, che contiene la lista di categorie che potranno essere applicate ad un libro.



Autenticazione

Nel diagramma di contesto analizzato è presente un sistema paritario “Autenticazione Google” ma è anche presente la possibilità di autenticarsi attraverso credenziali a scelta. (email e password)
Per gestire l’autenticazione dei vari utenti, è stata quindi individuata la classe Autenticazione.
Permette agli Utenti Anonimi di effettuare la registrazione attraverso nome utente, e-mail e password oppure attraverso un account Google.
In questo ultimo caso la web-app si interfaccerà con il sistema esterno per la gestione dell’autenticazione.
La classe individuata permette inoltre di verificare le credenziali fornite e effettuare poi il login.
Inoltre permette di recuperare la password collegata ad un certo account e di cancellare il proprio account.

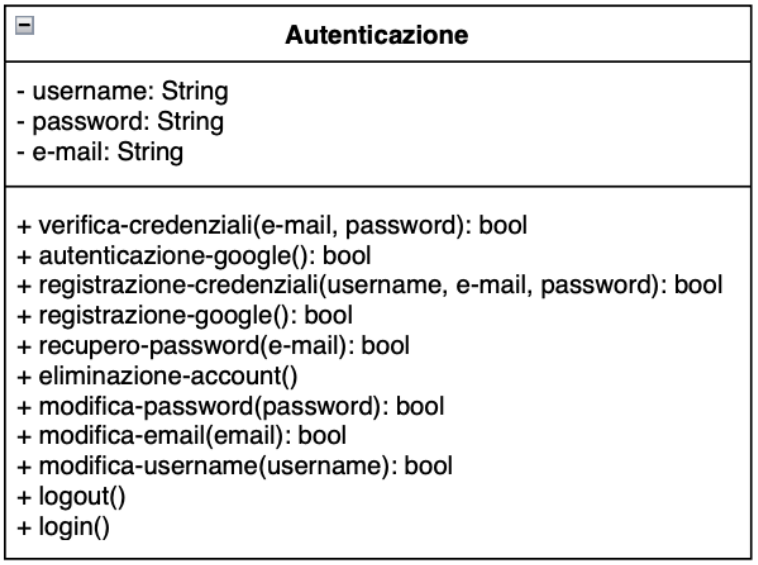
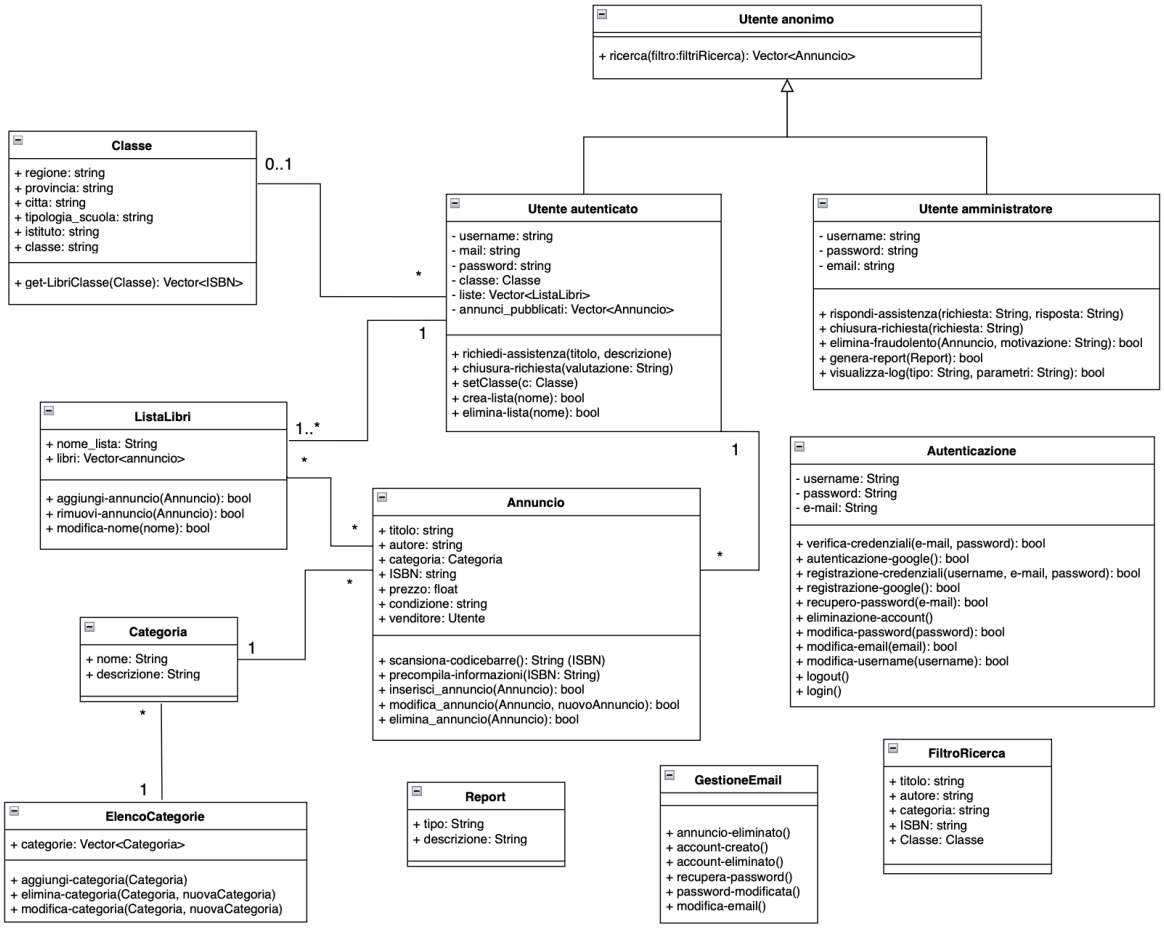


Diagramma delle classi complessivo

Riportiamo qui sotto il diagramma delle classi complessivo individuato.
Oltre alle classi già descritte, sono state inserite classi ausiliarie, come “Report”, “GestioneEmail” e “FiltroRicerca”.
Queste classi servono per descrivere eventuali tipi strutturati usati, ad esempio, negli attributi delle altre classi.

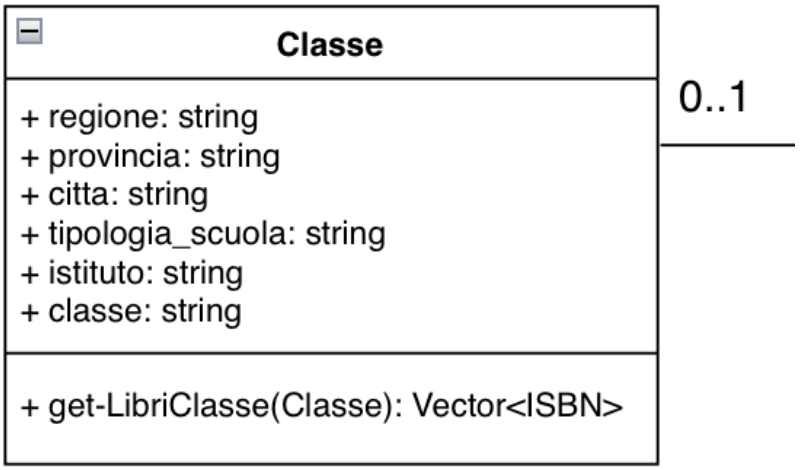


Codice in Object Constraint Language

In questo capitolo è descritta in modo formale la logica prevista nell'ambito di alcune operazioni di alcune classi. Tale logica viene descritta in Object Constraint Language (OCL) perché tali concetti non sono esprimibili in nessun altro modo formale nel contesto di UML.

Get libri classe

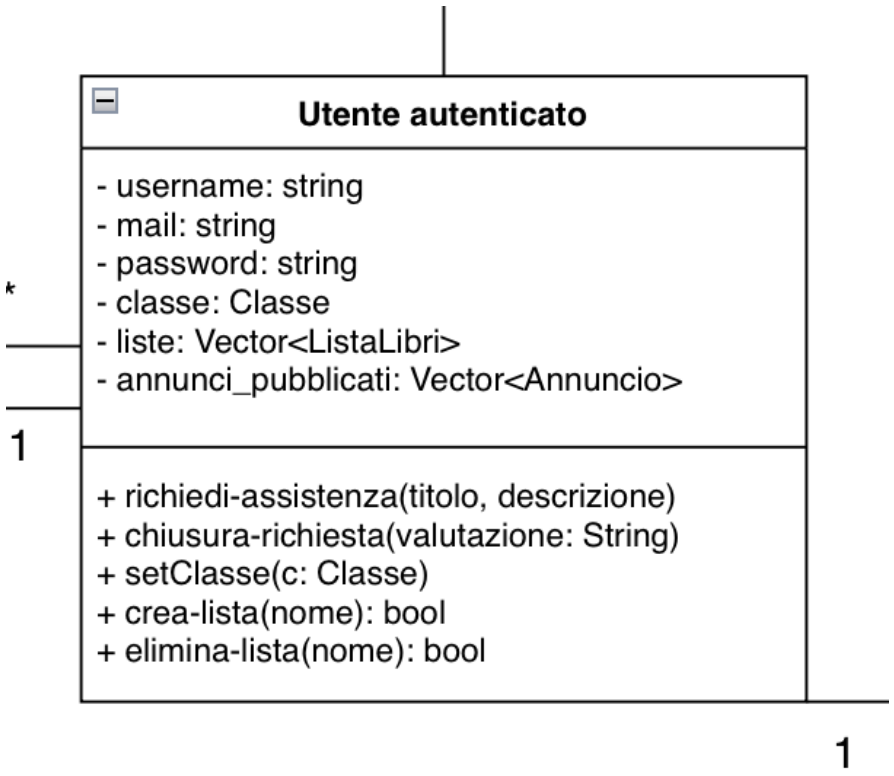
Gli utenti autenticati possono inserire attraverso diversi campi una specifica classe, così da ottenere una lista di ISBN dei libri adottati dalla classe specificata.
Tutti i campi richiesti della Classe devono quindi essere impostati correttamente e non devono essere vuoti. Questa condizione è stata espressa in OCL attraverso una pre-condizione.



context Classe::get-LibriClasse(Classe)
pre: Classe.regione, provincia, città, tipologia_scuola, istituto, classse != null

Imposta classe utente autenticato

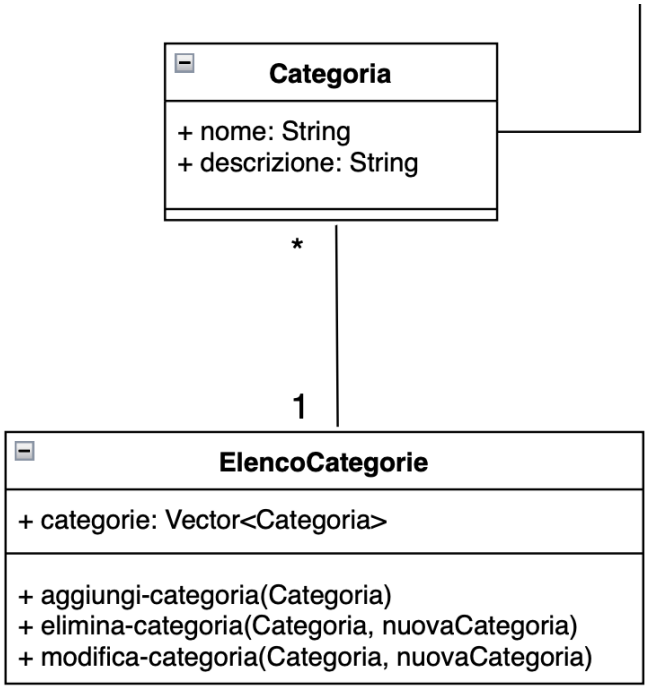
Gli utenti che vogliono inserire la propria classe devono fornire tutti i campi richiesti e non devono quindi essere vuoti come espresso dalla pre condizione OCL.
Dopo l'esecuzione di questo metodo, la classe dell'utente deve essere impostata alla classe richiesta, questa condizione è stata espressa in OCL attraverso una post-condizione.



context Utente autenticato::setClasse(c: Classe)
pre: c: Classe != null
post: Utente autenticato.classe = c: Classe

Modifica o eliminazione di una categoria

Quando viene modificata una categoria dall'elenco di categorie disponibili, viene richiesto anche un attributo aggiuntivo nuovaCategoria, la quale sostituirà la Categoria specificata, con l'effetto che tutti i libri della vecchia categoria saranno presenti nella nuova Categoria.
Similmente, quando viene eliminata una categoria, viene richiesto un attributo aggiuntivo nuovaCategoria, perchè tutti gli annunci presenti nella categoria eliminata verranno trasferiti in quella specificata da nuovaCategoria.
Questa condizione è specificata attraverso una post-condizione in OCL.

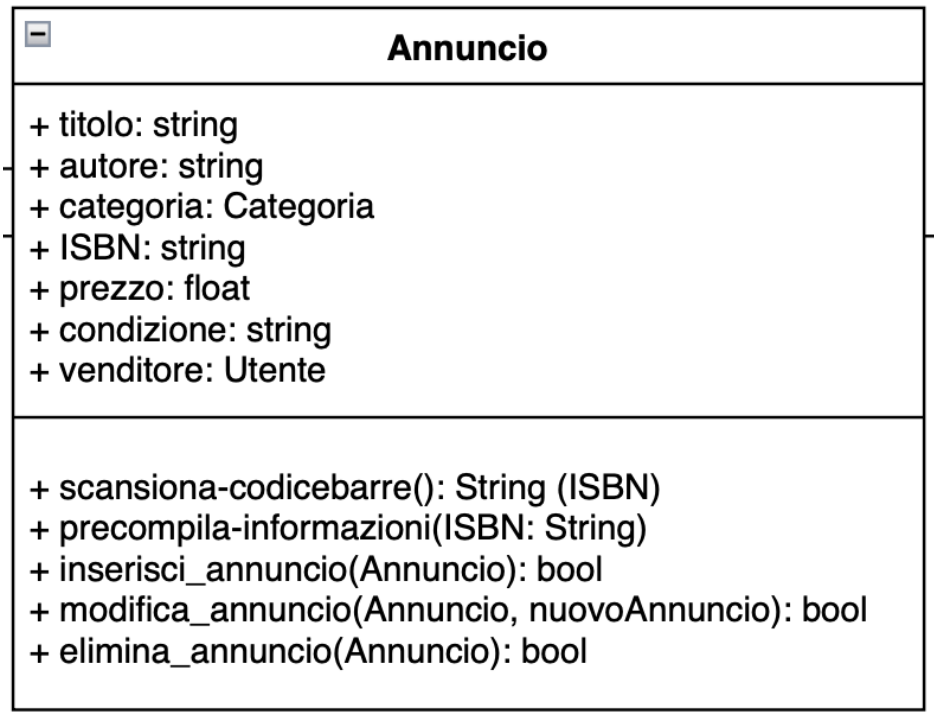


context ElencoCategorie::elimina-categoria(Categoria, nuovaCategoria)
post: Annuncio.categoria = nuovaCategoria

context ElencoCategorie::modifica-categoria(Categoria, nuovaCategoria)
post: Annuncio.categoria = nuovaCategoria

Precompilazione e inserimento di un annuncio

Nella classe annuncio, sono presenti diversi metodi.
Quando viene richiesta la precompilazione delle informazioni nell'aggiunta di un annuncio, viene richiesto un codice ISBN, quindi verranno impostati i campi secondo i dati relativi all'ISBN dato.
Dopo aver eseguito questo metodo, i vari campi titolo, autore, categoria e ISBN non possono essere vuoti (perché precompilati), e questa condizione è espressa in OCL attraverso una post-condizione.
Similmente, nel contesto dell'inserimento dell'annuncio, tutti i campi richiesti devono essere impostati correttamente e non devono essere vuoti. Questa condizione è stata espressa in OCL attraverso una pre-condizione.
Infine, dopo che l'annuncio è stato inserito, il campo Venditore non può essere non impostato, questa condizione è stata espressa in OCL attraverso una post-condizione.



context Annuncio::precompila-informazioni(ISBN)
post: Annuncio.titolo, autore, categoria, ISBN != null

context Annuncio::inserisci-annuncio(Annuncio)
pre: Annuncio.titolo, autore, categoria, ISBN, prezzo, condizione != null
post: Annuncio.venditore != null

Diagramma delle classi con codice OCL

Riportiamo infine il diagramma delle classi con tutte le classi fino ad ora presentate ed il codice OCL individuato.