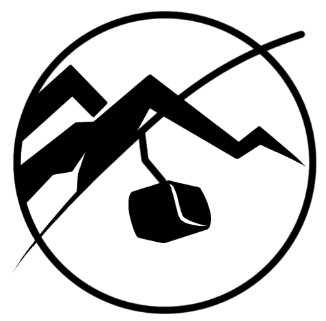
****

**PROGETTO DI INGEGNERIA DEL SOFTWARE**

**SKI ONLINE**

**Analisi dei requisiti**

# Indice dei contenuti

[Indice dei contenuti 2](#_Toc118970639)

[Scopo del documento 2](#_Toc118970640)

[Diagramma delle classi 2](#_Toc118970641)

**Scopo del documento**

Il presente documento riporta la definizione dell’architettura del progetto SKI ONLINE usando diagrammi delle classi in Unified Modeling Language (UML) e codice in Object Constraint Language (OCL). Nel precedente documento è stato presentato il diagramma degli use case, il diagramma di contesto e quello dei componenti. Ora, tenendo conto di questa progettazione, viene definita l’architettura del sistema dettagliando da un lato le classi che dovranno essere implementate a livello di codice e dall’altro la logica che regola il comportamento del software. Le classi vengono rappresentate tramite un digramma delle classi in linguaggio UML. La logica viene descritta in OCL perché tali concetti non sono esprimibili in nessun altro modo formale nel contesto di UML.

# Diagramma delle classi

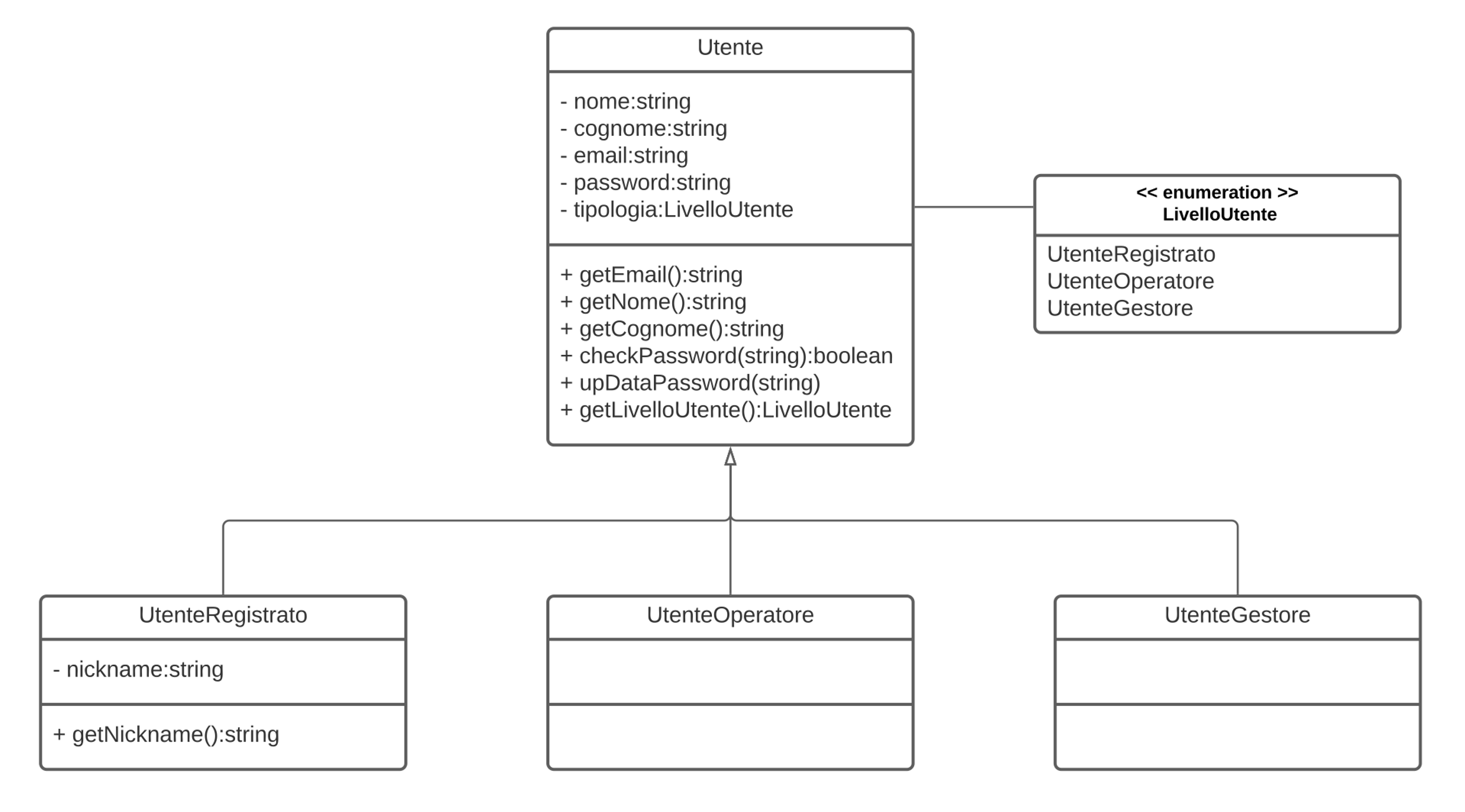
Nel presente capitolo vengono presentate le classi previste nell’ambito del progetto SKI ONLINE. Ogni componente presente nel diagramma dei componenti diventa una o più classi. Tutte le classi individuate sono caratterizzate da un nome, una lista di attributi che identificano i dati gestiti dalla classe e una lista di metodi che definiscono le operazioni previste all’interno della classe. Ogni classe può essere anche associata ad altre classi e, tramite questa associazione, è possibile fornire informazioni su come le classi si relazionano tra loro.

Riportiamo di seguito le classi individuate a partire dai diagrammi di contesto e dei componenti. In questo processo si è proceduto anche nel massimizzare la coesione e minimizzare l’accoppiamento tra classi.

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.

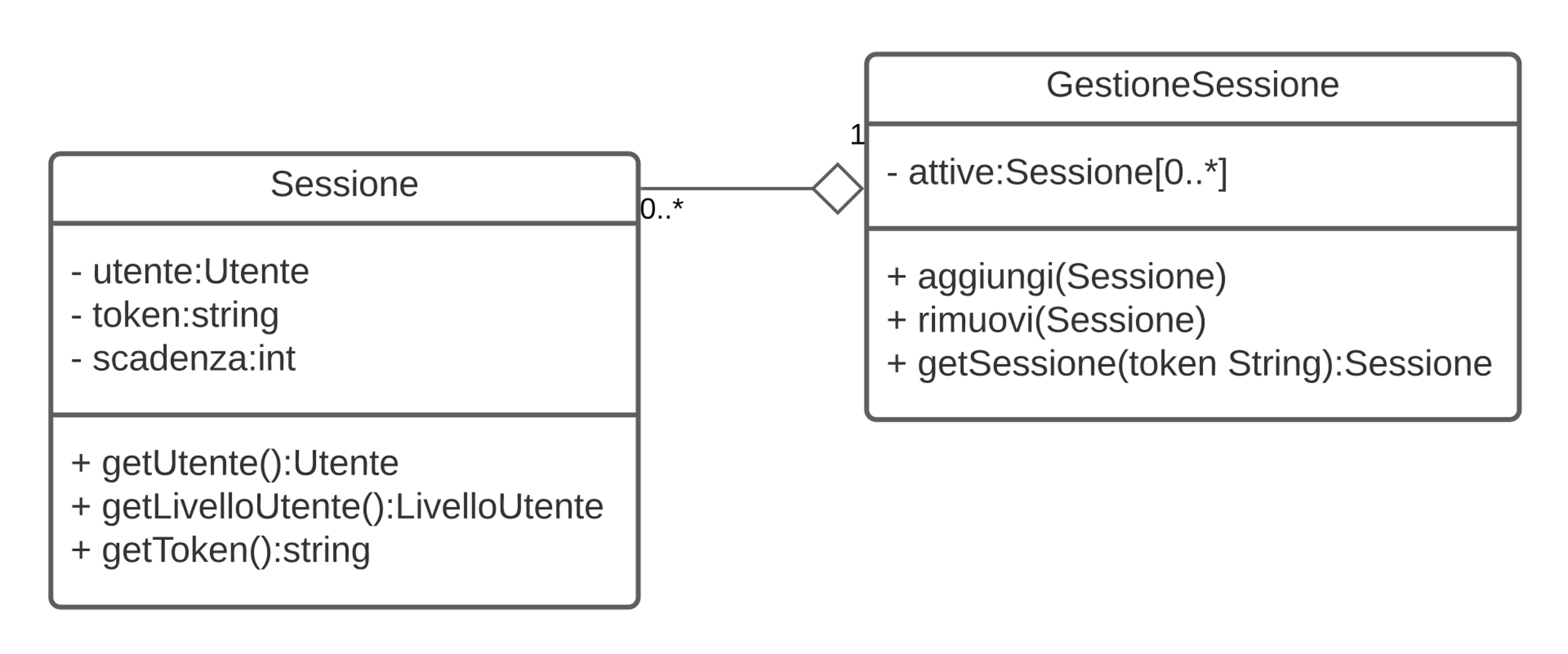
## Utenti del sistema

Analizzando i vari livelli di utenza che sono previsti nel sistema, ovvero: “Utente Registrato”, “Utente Operatore” e “Utente Gestore”, sono state identificate tre classi corrispondenti ai tre livelli di utenza, accomunate da una classe madre comune “Utente”. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



## Gestione sessione

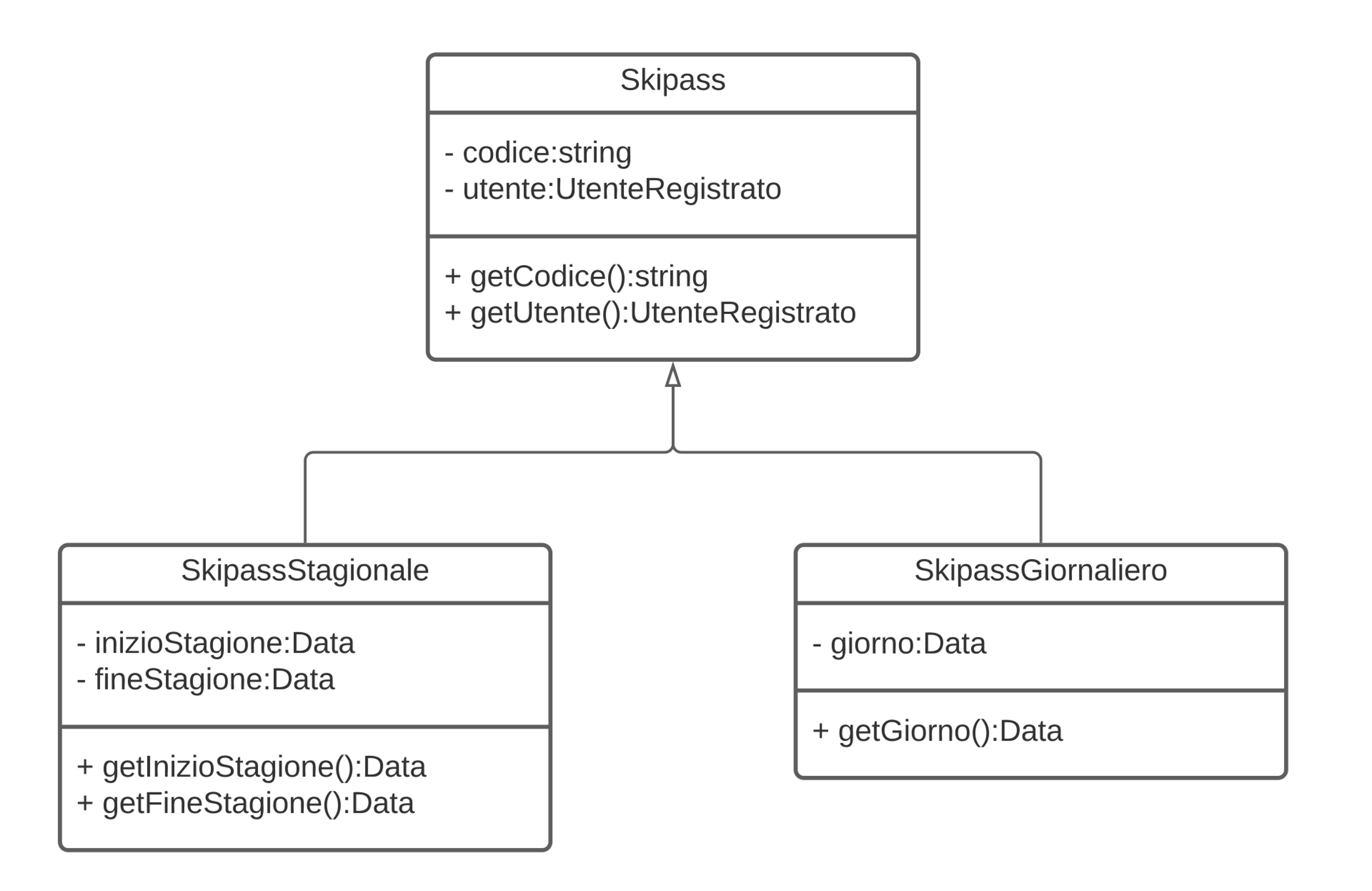
Analizzando la componente “Gestione sessione” questa è stata scomposta in due classi “Sessione” e “Gestione sessione”, la prima si occupa di manipolare i dati di una singola sessione mentre la seconda di raccogliere e gestire tutte le sessioni attualmente attive. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



*Il datatype “Utente” e l’enumeration “LivelloUtente” fanno riferimento a quelli definiti precedentemente*

## Skipass

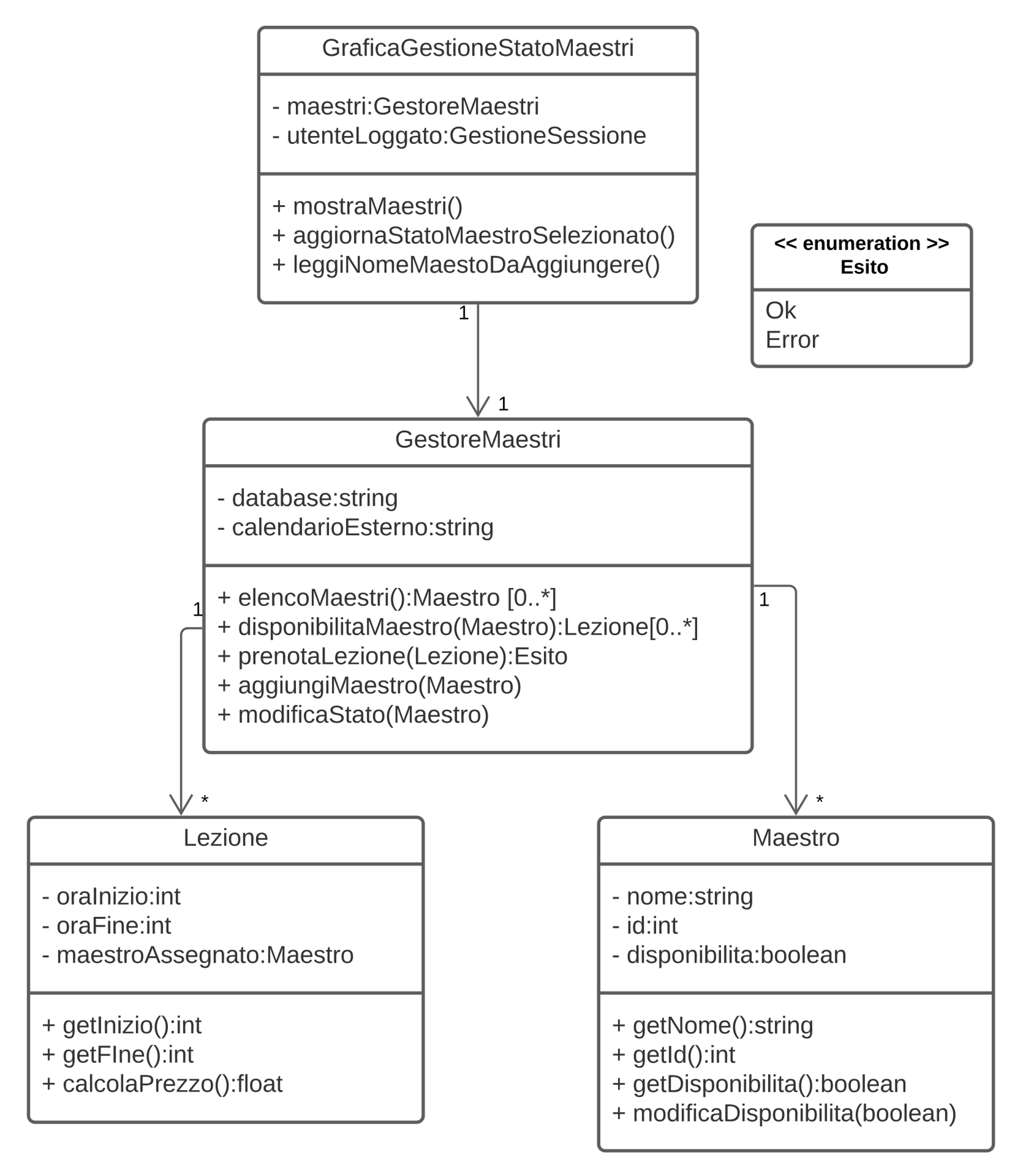
Analizzando le due diverse tipologie di skipass acquistabili, ovvero “Skipass giornaliero” e “Skipass stagionale”, sono state identificate due classi corrispondenti alle due tipologie di skipass, accomunate da una classe madre “Skipass” comune. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



## 

## Gestione maestri

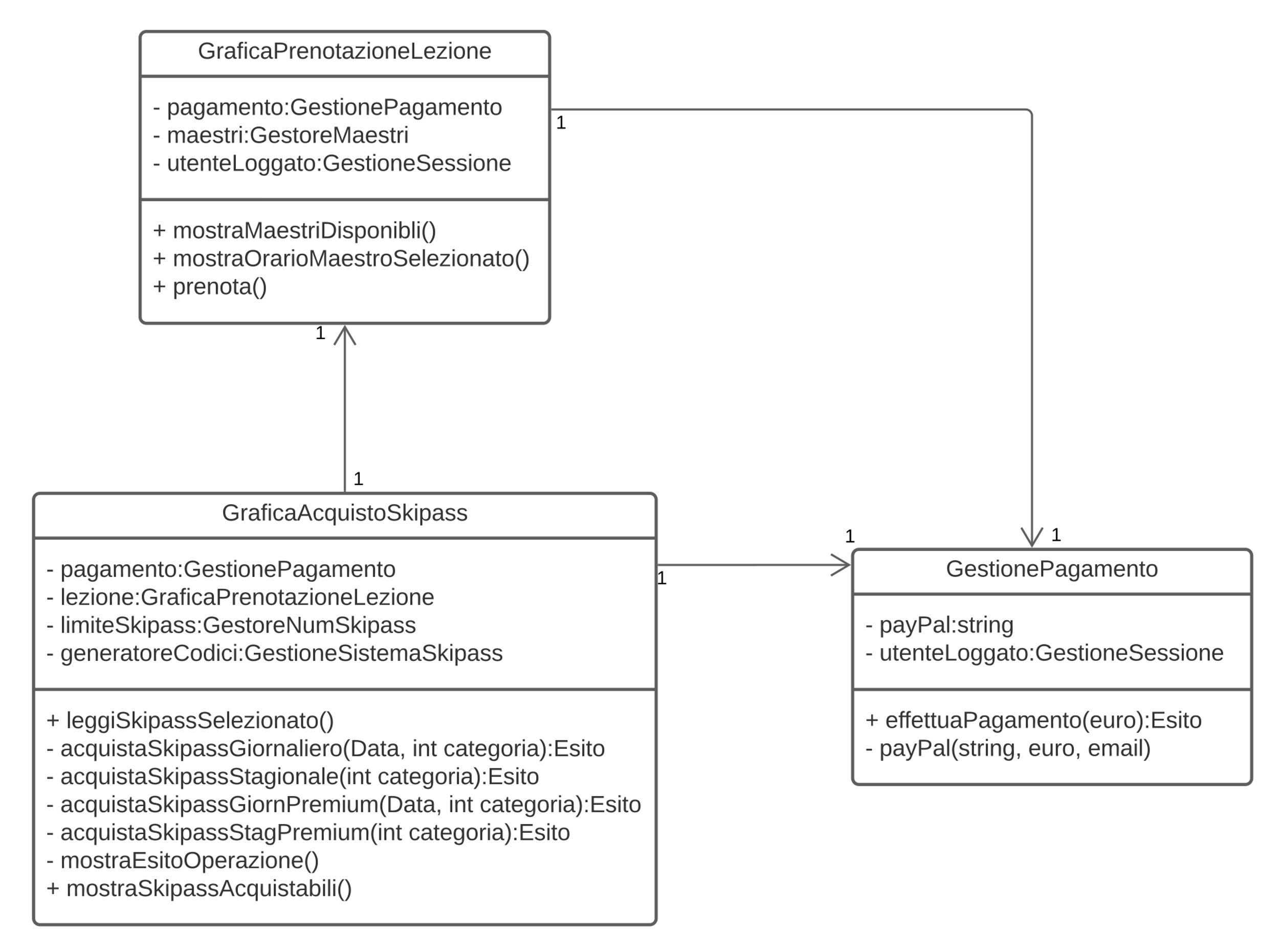
Analizzando le componenti “Gestione maestri” e “Grafica gestione stato maestri”, sono state identificate quattro classi, due direttamente correlate alle due componenti, mentre le rimanenti “Lezione” e “Maestro” derivate dalla tipologia di dati che le prime due utilizzano. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



*Il tipo di dato “Gestione sessione” si riferisce alla classe definita precedentemente*

## Acquisto skipass e prenotazione lezioni

Analizzando le componenti “Grafica acquisto skipass”, “Gestione pagamento” e “Grafica prenotazione lezione”, si è proceduto a identificare tre classi con le medesime funzionalità. Di seguito il dettaglio di queste classi con i propri attributi e metodi.

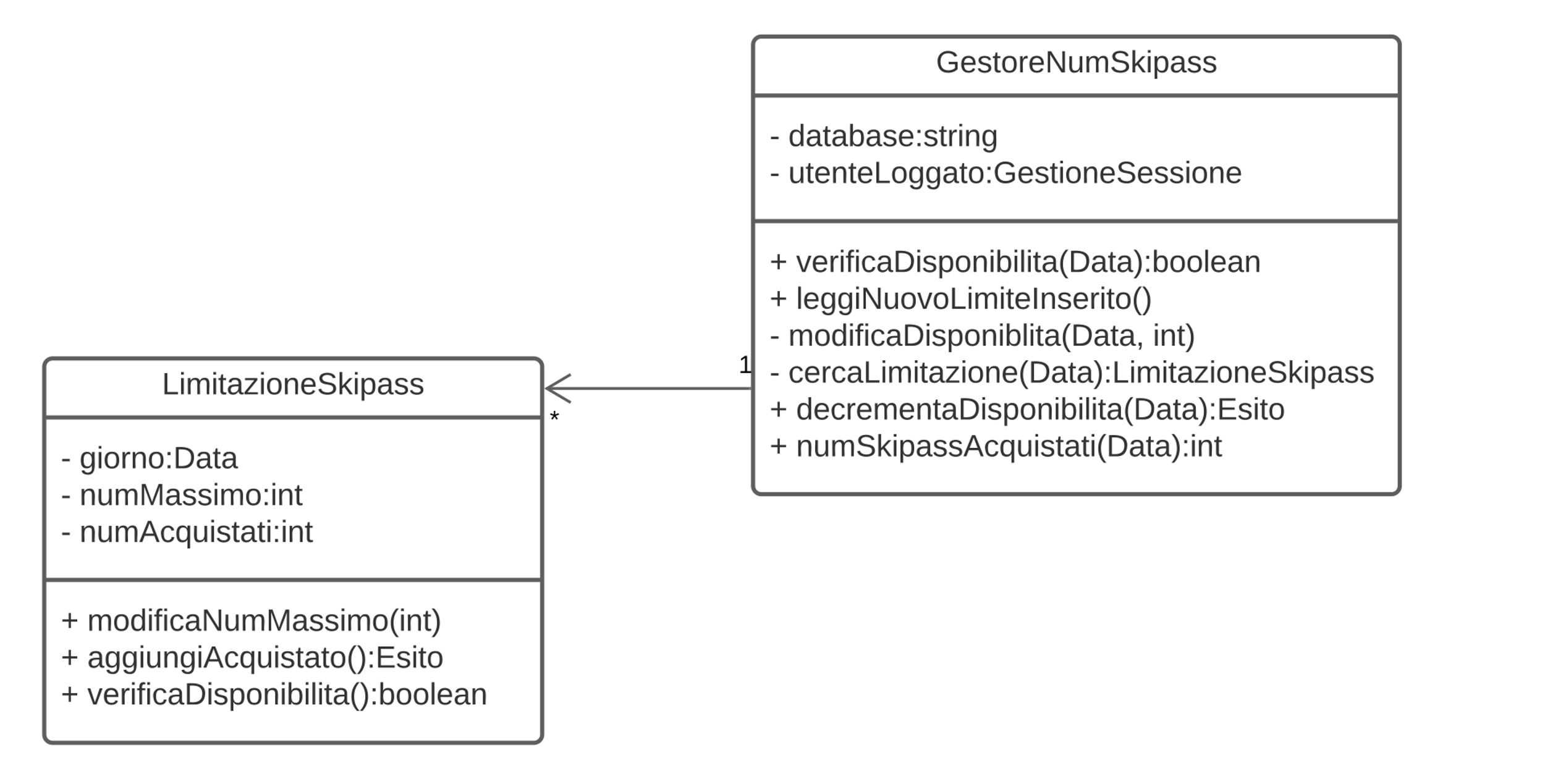


*I tipi di dato “Gestione sessione”, “Gestione maestri” ed “Esito” si riferiscono alle classi (ed enumeration) definite precedentemente*

*I tipi di dato “Gestione numero skipass” e “Gestione sistema skipass” si riferiscono a delle classi che verranno definite successivamente*

## Limitazione numero skipass acquistabili

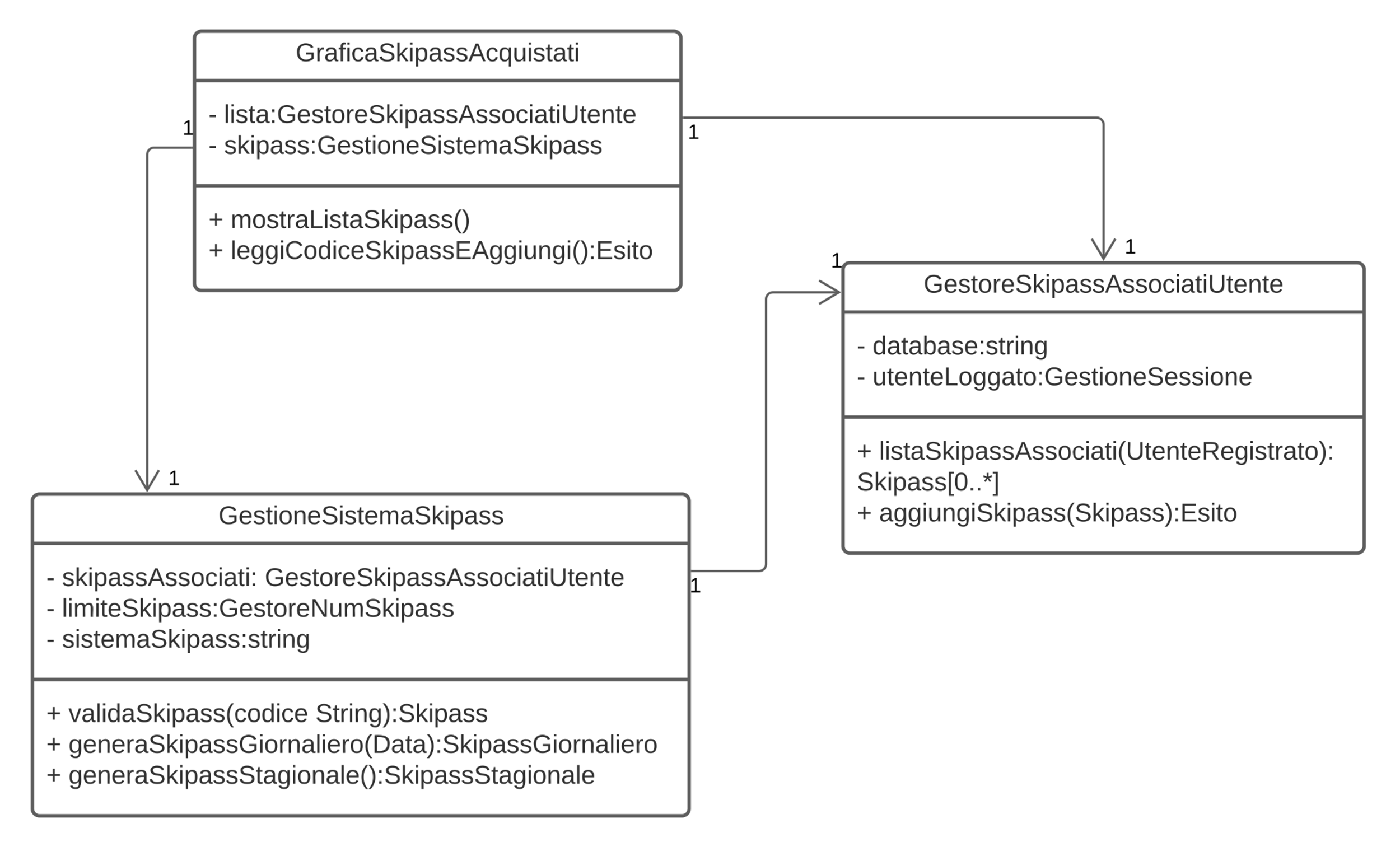
Analizzando la componente “Gestione numero skipass acquistabili” questa è stata scomposta in due classi “GestoreNumeroSkipassAcquistabili” e “LimitazioneSkipass”, la prima si occupa di aggiungere, rimuovere e modificare limitazioni in diverse giornate, mentre la seconda si occupa di manipolare la singola limitazione in una giornata.



*I tipi di dato “Data”, “Gestione sessione” ed “Esito” si riferiscono alle classi definite precedentemente*

## Gestione skipass associati ad una utenza

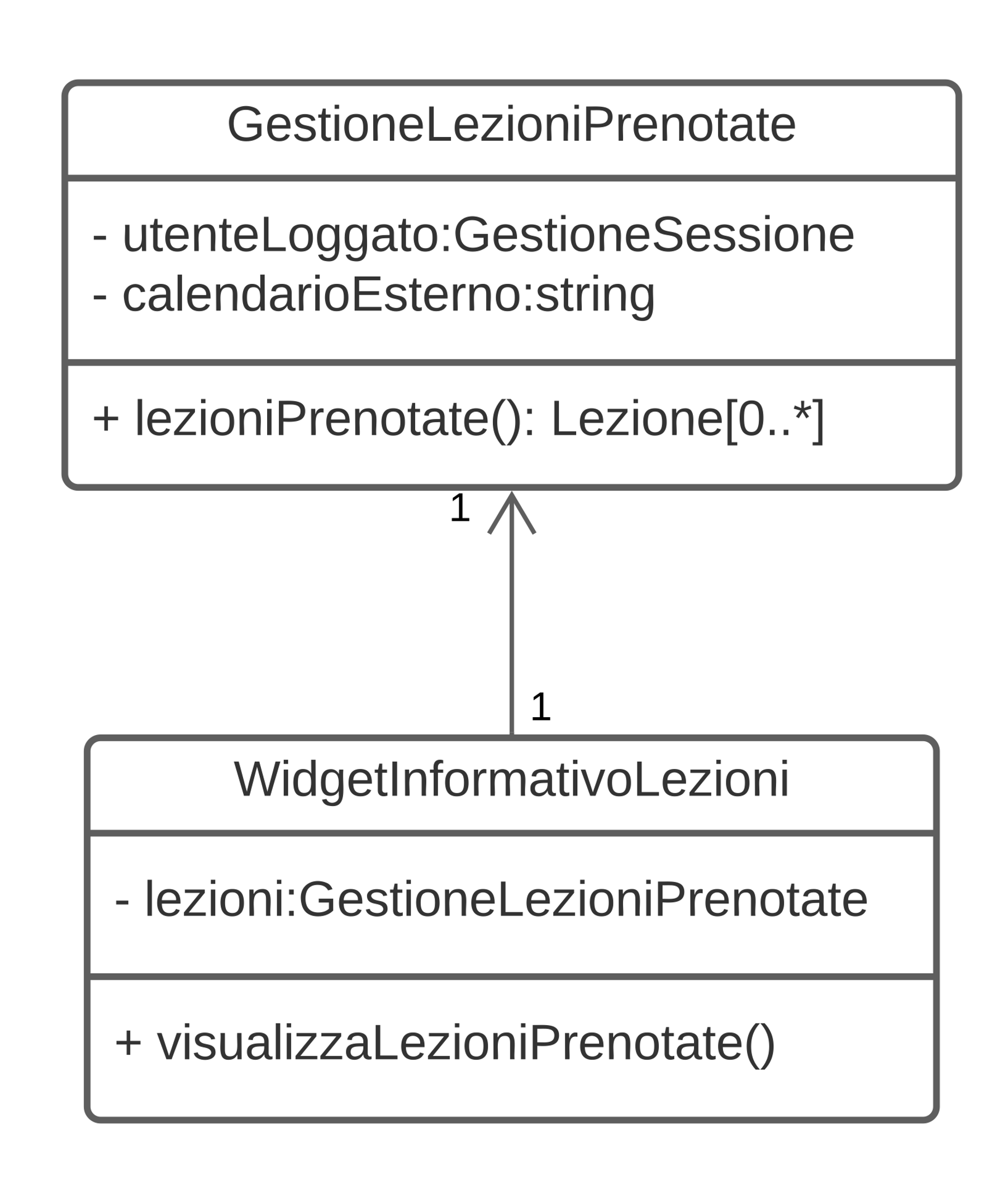
Analizzando le componenti “Gestione sistema skipass preesistente”, “Grafica skipass aquistati” e “Gestione skipass associati ad una utenza” si è proceduto a identificare tre classi con le medesime funzionalità. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



*I tipi di dato “GestoreNumSkipass”, “Gestione sessione”, “Utente Registrato”, “Esito”, “Data”, “Skipass”, “Skipass giornaliero”, “Skipass stagionale” si riferiscono alle classi definite precedentemente*

## Widget informativo lezioni

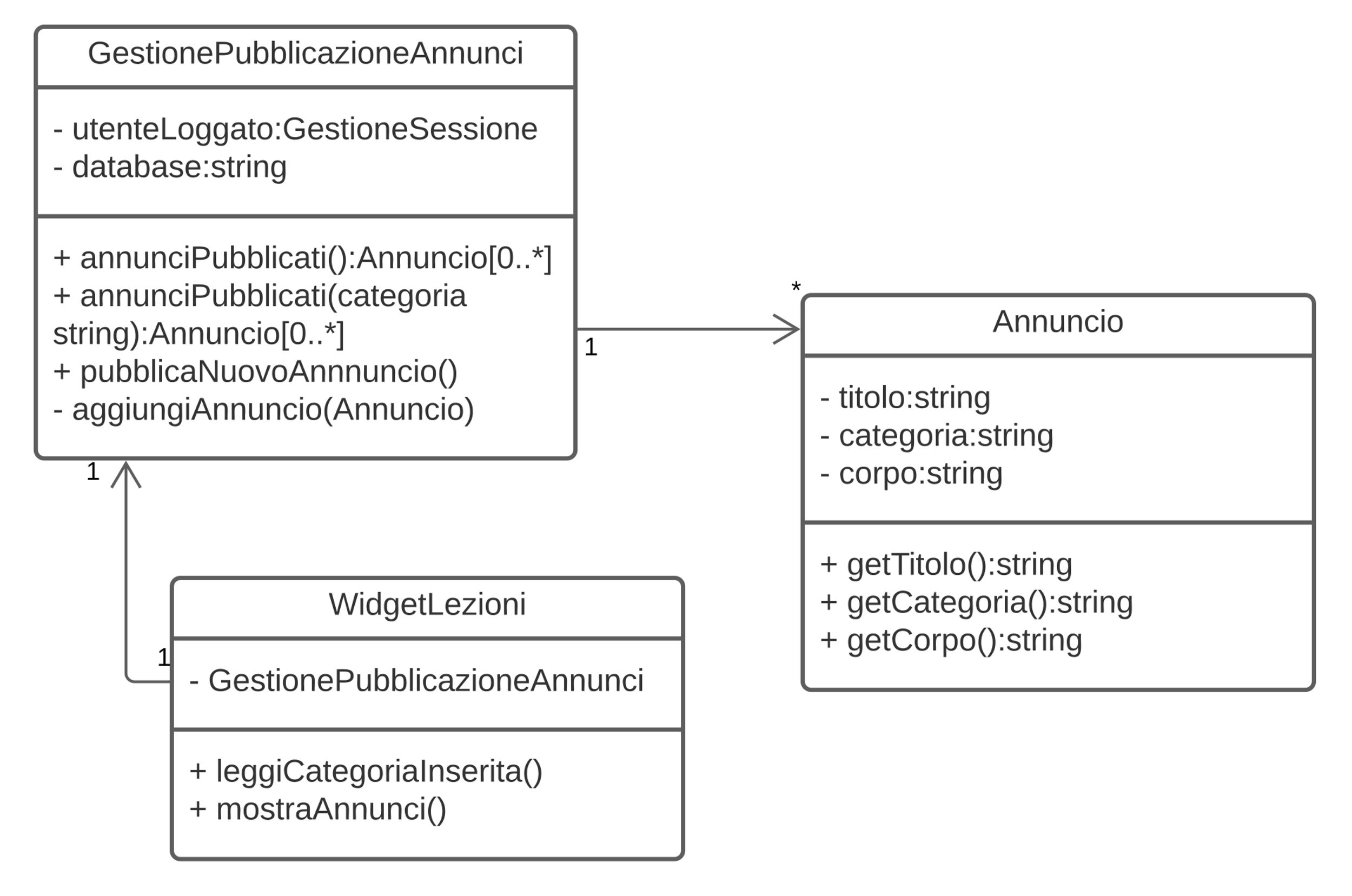
Analizzando le componenti “Widget informativo lezioni” e “Gestione lezioni prenotate” si è proceduto a identificare due classi con le medesime funzionalità. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



*I tipi di dato “Lezione” e “Gestione sessione” si riferiscono alle classi definite precedentemente*

## Widget annunci pubblicati

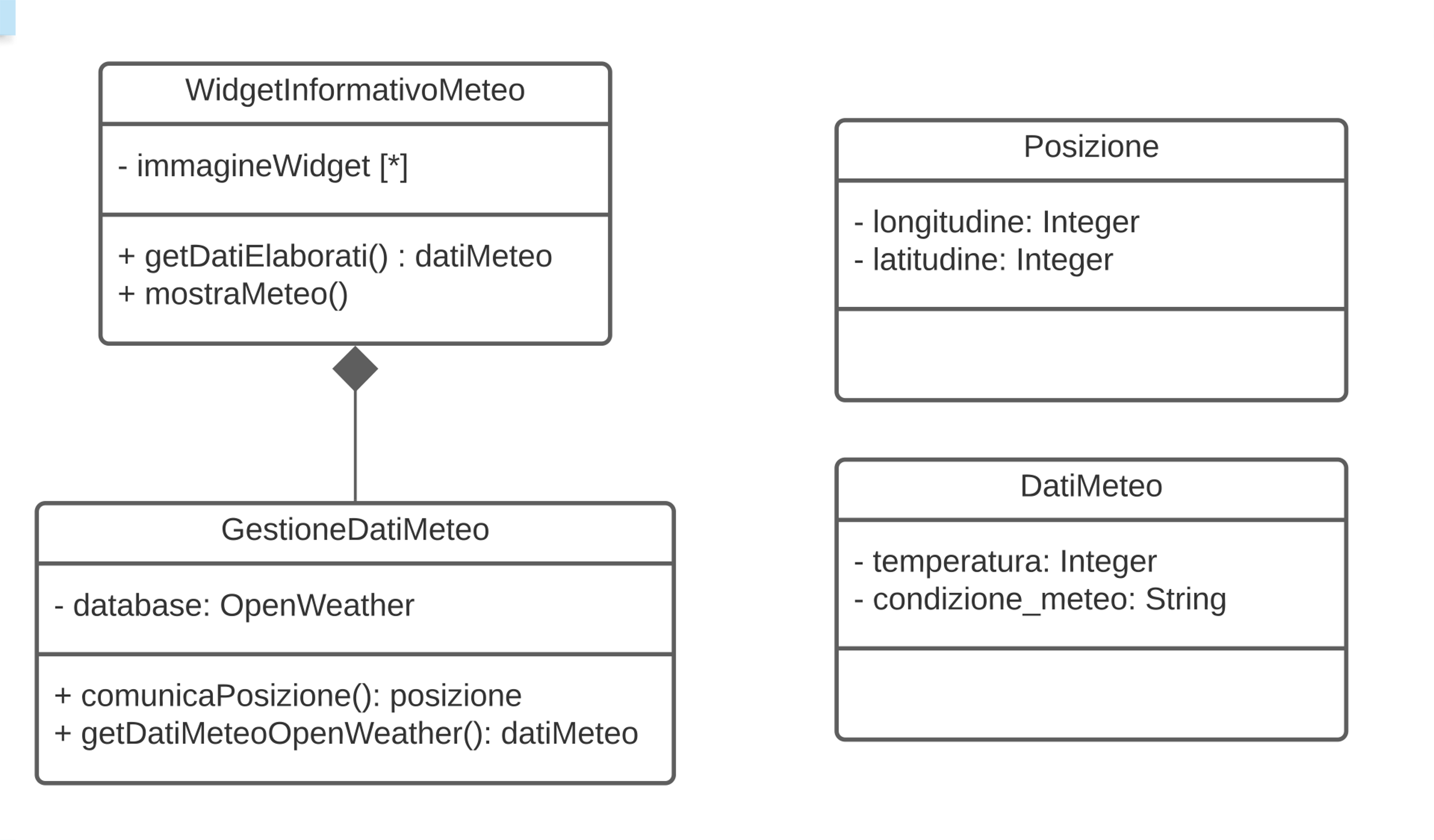
Analizzando le componenti “Widget annunci” e “Gestione pubblicazione annunci” si è proceduto a identificare due classi con le medesime funzionalità. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



*Il tipo di dato “Gestione sessione” si riferisce alla classe definite precedentemente*

## Gestione widget meteorologico

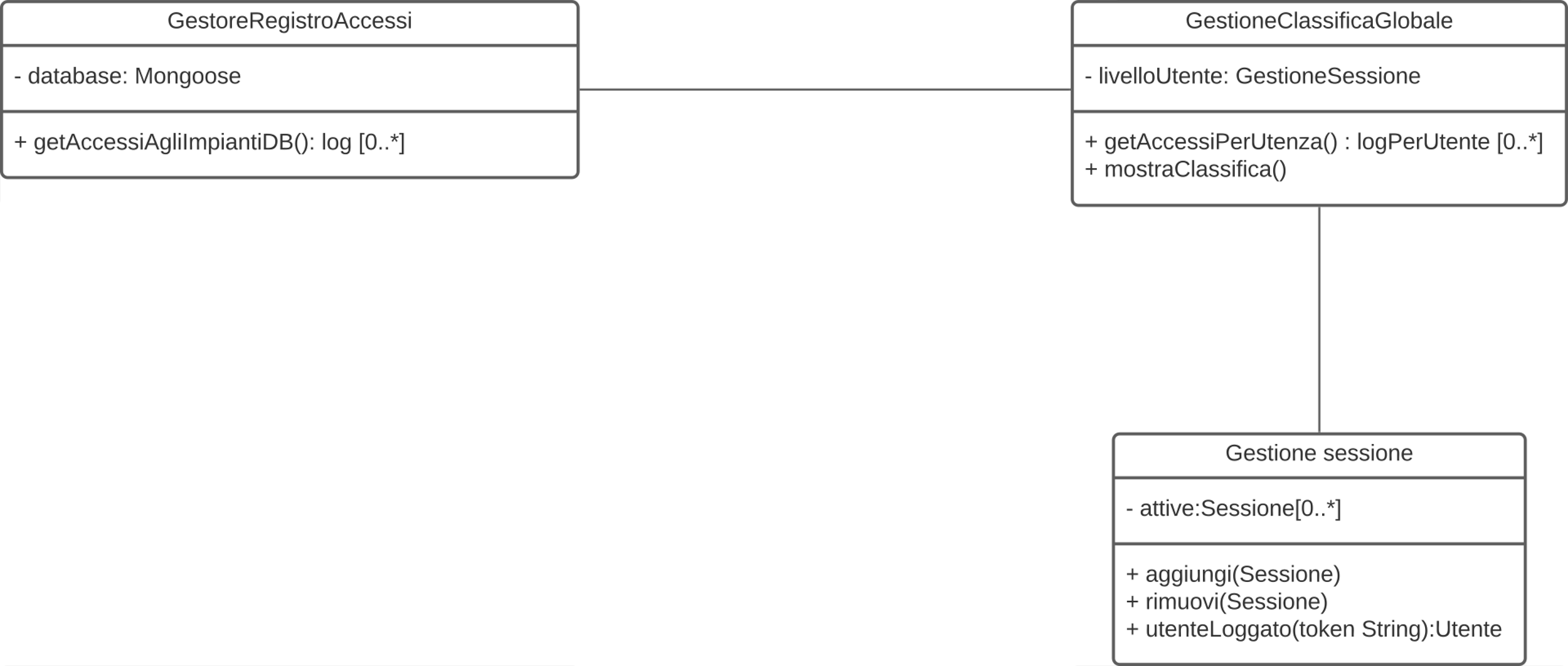
Analizzando le componenti “Widget informativo meteo” e “Gestione dati meteo” queste sono state identificate come classi per la gestione e visualizzazione del widget dedicato alla visione delle condizioni meteorologiche. Di seguito il dettaglio di queste classi con i propri attributi e metodi.



## Gestione stato impianti

## Gestione classifica Globale

Analizzando la componente “Gestione classifica Globale” è stata identificata come classe responsabile della visualizzazione della classifica globale. Definiamo ora la classe e le relative classi ad essa collegate.



## Gestione storico impianti