



Allegato Tecnico

Gruppo SWEight - Progetto Colletta

SWEightGroup@gmail.com

Informazioni sul documento

Versione	1.0.0
Approvatore	Damien Ciagola
Redattori	Enrico Muraro Damien Ciagola Francesco Magarotto
Verificatori	Sebastiano Caccarto Alberto Bacco
Uso	Esterno
Distribuzione	Prof. Vardanega Tullio Prof. Cardin Riccardo Gruppo SWEight

Descrizione

Documento che ha lo scopo di illustrare la Product Baseline, ponendo attenzione alle scelte architettureali e alla copertura di use case e requisiti funzionali

Registro delle modifiche

Versione	Data	Descrizione	Autore	Ruolo
1.0.0	2019-04-05	Approvazione per il rilascio	Damien Ciagola	<i>Responsabile di Progetto</i>
0.4.0	2019-04-05	Verifica del documento	Enrico Muraro	<i>Verificatore</i>
0.4.0	2019-04-03	Stesura sezione §3	Francesco Magarotto	<i>Redattore</i>
0.3.0	2019-04-02	Stesura sezione §4	Damien Ciagola	<i>Redattore</i>
0.2.0	2019-04-02	Stesura sezione §5	Francesco Magarotto	<i>Redattore</i>
0.1.0	2019-03-31	Stesura sezione §1 e §2	Damien Ciagola	<i>Redattore</i>
0.0.1	2019-03-30	Creazione scheletro del documento	Damien Ciagola	<i>Redattore</i>

Indice

1	Introduzione	5
1.1	Scopo del documento	5
1.2	Scopo del prodotto	5
1.3	Glossario	5
2	Installazione ed esecuzione	6
2.1	Maven project	6
2.2	Node.js	6
3	Architettura del prodotto	7
3.1	Design pattern utilizzati	7
3.1.0.1	Object Adapter	7
3.2	MongoDB Database	8
4	Diagrammi dei package	9
4.1	Model	9
4.2	Controller e service	10
4.3	Service e repository	11
5	Diagrammi delle classi e di sequenza	12
5.1	Inserimento di un utente	12
5.2	Login	14

Elenco delle figure

1	Object Adapter	7
2	Exercise insert	8
3	Model	9
4	Controller e Service	10
5	Service e Repository	11
6	Exercise insert	12
7	UML - FrontEnd	13
8	Authorization	14

Elenco delle tabelle

1 Introduzione

1.1 Scopo del documento

Il presente documento ha lo scopo di fornire agli sviluppatori uno specchio informativo sul design strutturale e logico della piattaforma Colletta. Il documento sarà inoltre corredato da diagrammi UML 2.X delle principali scelte prese dal gruppo SWEight e descriverà le tecnologie utilizzate nella realizzazione dell'applicazione.

1.2 Scopo del prodotto

Il prodotto da realizzare consta in un'applicazione web che fornisca uno strumento per creare e svolgere esercizi di analisi grammaticale, e al contempo né raccolga i risultati. I dati raccolti verranno impiegati dagli sviluppatori dell'azienda proponente come strumento per il miglioramento di algoritmi di apprendimento automatico_G. Nello specifico il prodotto verrà utilizzato da tre tipologie di utenti: le/gli insegnanti che si occuperanno della creazione degli esercizi, gli allievi che potranno svolgere gli esercizi e ottenere delle valutazioni e gli sviluppatori che filtreranno i dati secondo alcuni criteri, e infine li scaricheranno.

Il prodotto si interfacerà con un'applicazione di PoS-tagging_G, come FreeLing_G, a cui verrà delegata l'esecuzione dell'analisi grammaticale delle frasi.

1.3 Glossario

Al fine di rendere il documento il più comprensibile possibile e permetterne una rapida fruizione, viene allegato il *Glossario_v3.0.0* in cui sono presenti i termini contraddistinti dal pedice G. Tali termini includono abbreviazioni, acronimi, termini di natura tecnica, oppure sono fonte di ambiguità e pertanto necessitano di una definizione che renda il loro significato inequivocabile. Ogni termine, solo alla prima occorrenza per documento, verrà contrassegnato con la dicitura sopra indicata e rimanderà alla medesima definizione nel *Glossario_v3.0.0*.

2 Installazione ed esecuzione

Il codice relativo alla Product Baseline lo si può trovare al seguente link:

[linkAllaRepo](#).

2.1 Maven project

Una volta fatto il clone della repository o dopo aver scaricato lo zip, posizionarsi nella directory "Mockup-V2/Backend" della repo e utilizzare i seguenti comandi:

```
mvn clean install  
java -jar target/colletta-2019-0.0.1-SNAPSHOT.jar
```

2.2 Node.js

Per lo sviluppo lato server in linguaggio Javascript ci si avvale dell'ultima versione Long Term Support (LTS) di Node.js, che, al momento della stesura di questo documento, è la 10.15.1 LTS. Node.js è reperibile al seguente link:

[Node.js](#).

Il file package.json contiene tutte le configurazioni e dipendenze del progetto. Per installare tutti i moduli necessari è necessario eseguire il seguente comando nella cartella contenente il file package.json:

```
npm install
```

Per eseguire il progetto è necessario usare il comando:

```
npm start
```

3 Architettura del prodotto

3.1 Design pattern utilizzati

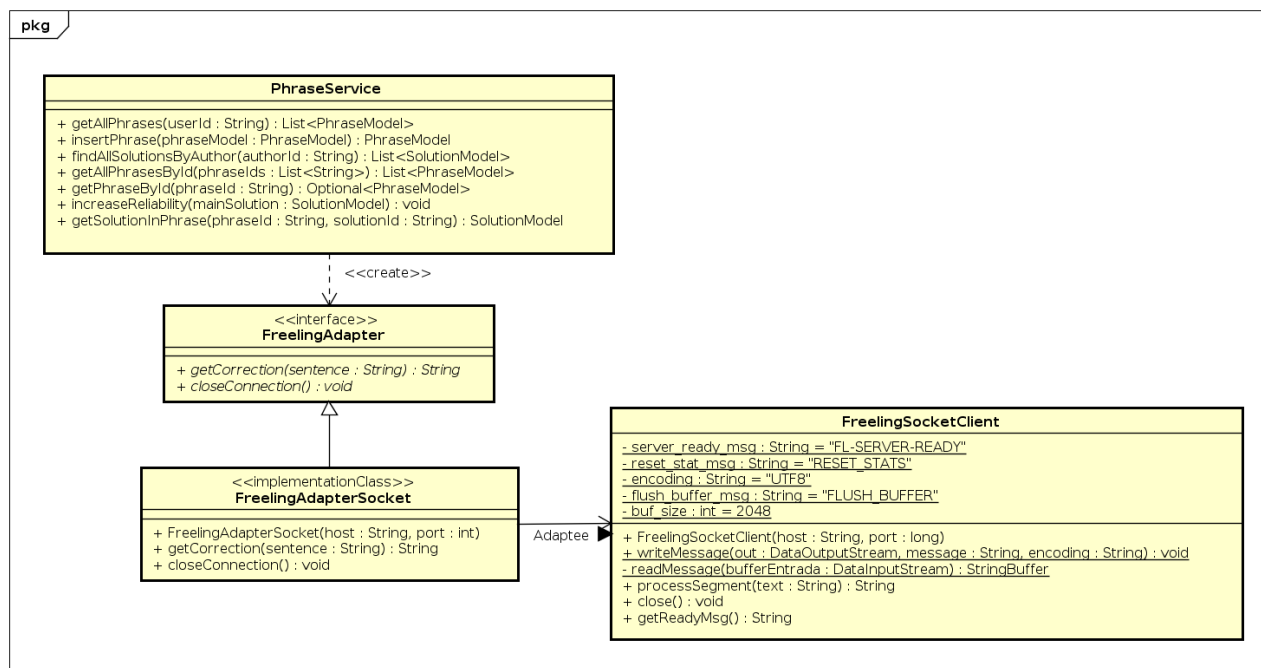


Figura 1: Object Adapter

3.1.0.1 Object Adapter

3.2 MongoDB Database

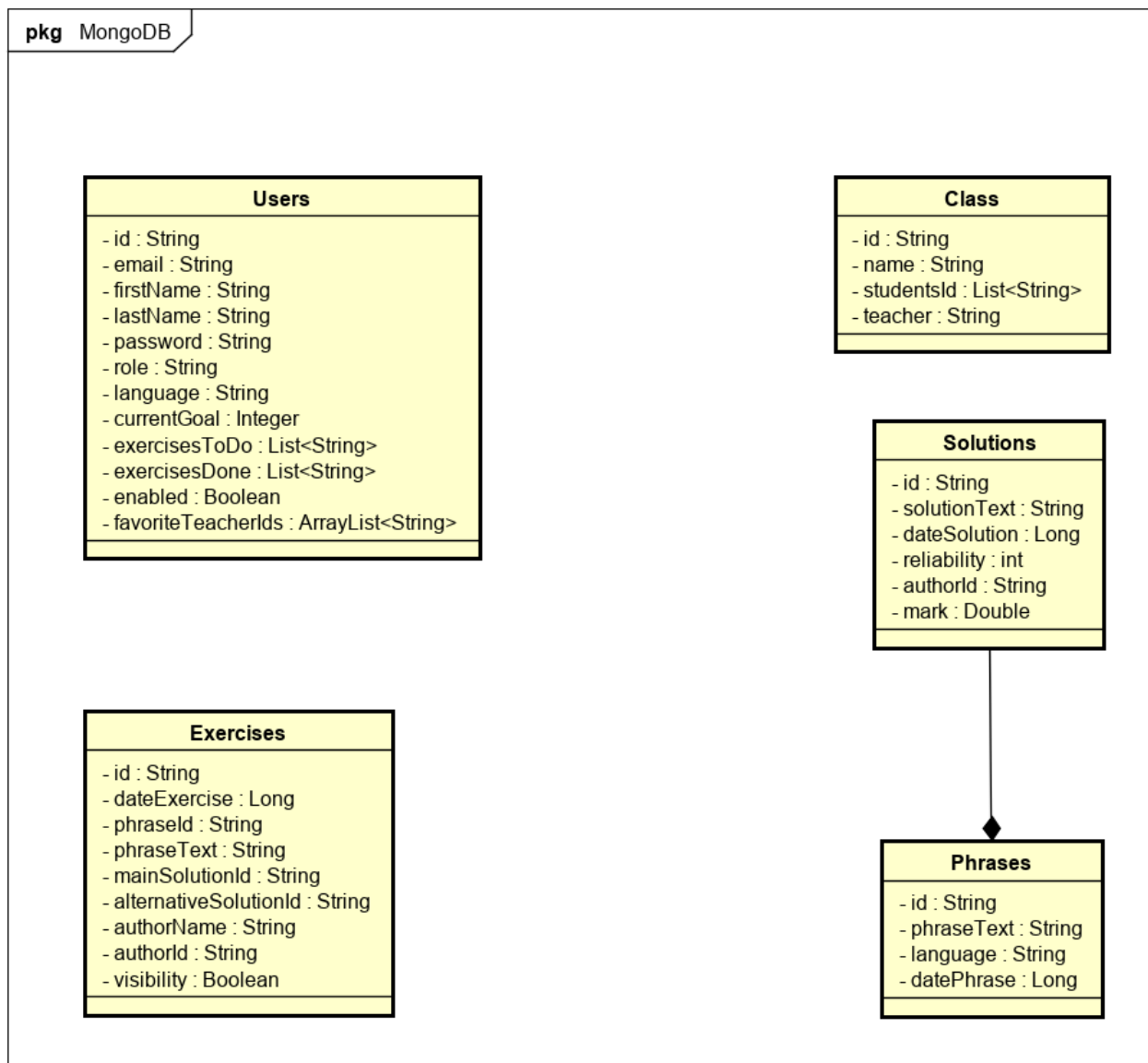


Figura 2: Exercise insert

4 Diagrammi dei package

4.1 Model

Viene di seguito riportato il diagramma delle classi del package model.

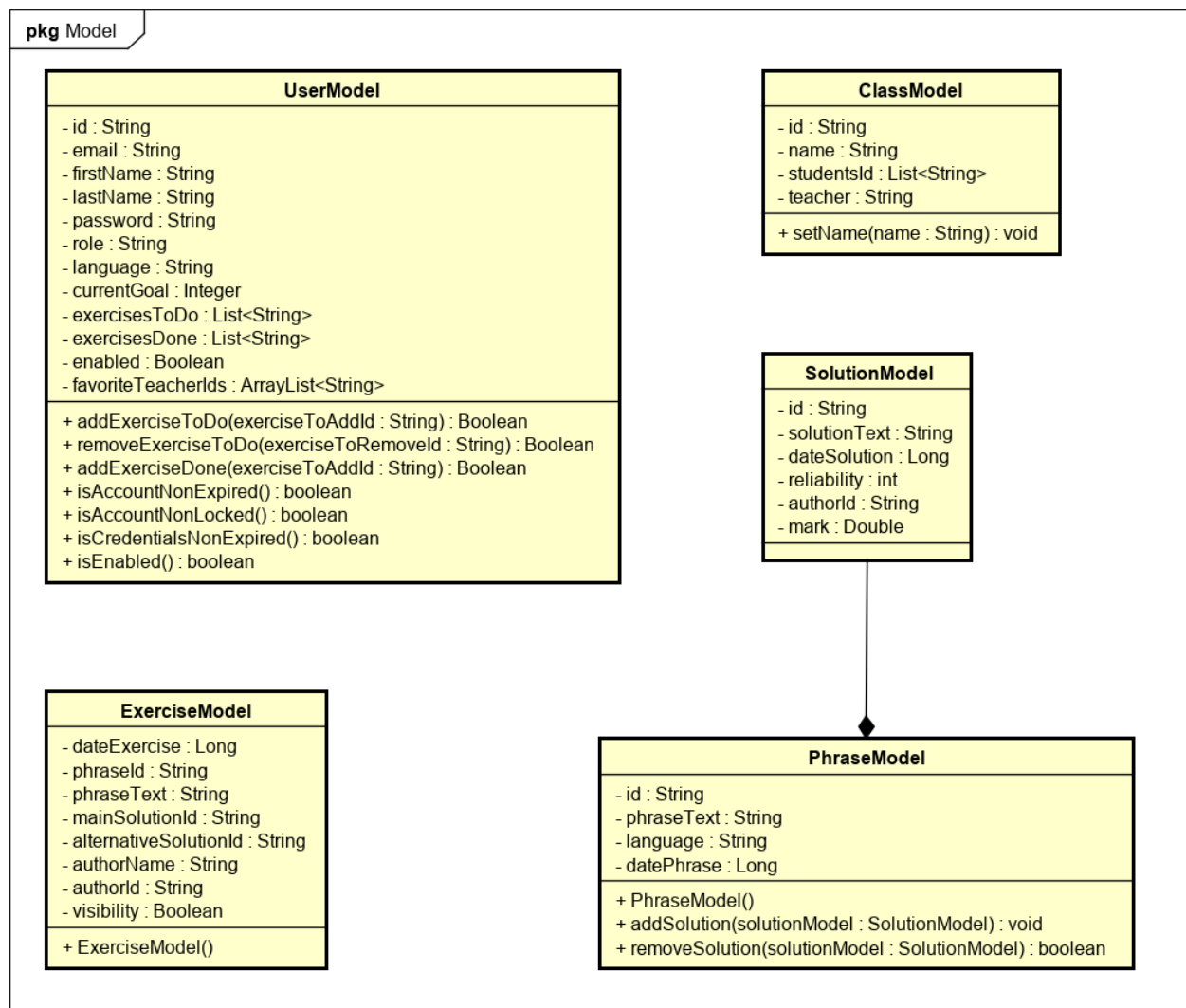


Figura 3: Model

4.2 Controller e service

Viene di seguito riportato il diagramma delle classi di package controller e service.

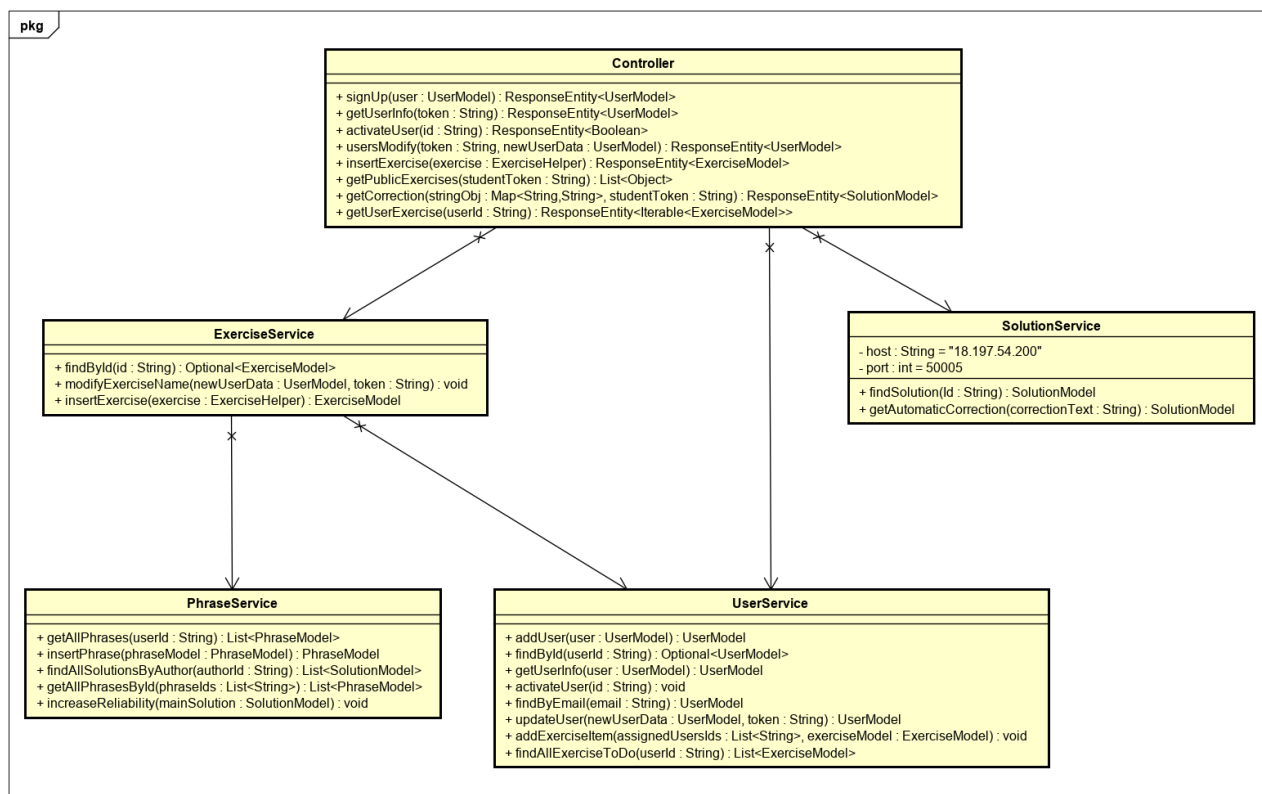


Figura 4: Controller e Service

5 Diagrammi delle classi e di sequenza

5.1 Inserimento di un utente

Il diagramma di sequenza rappresenta l'azione di inserimento di un esercizio nel sistema

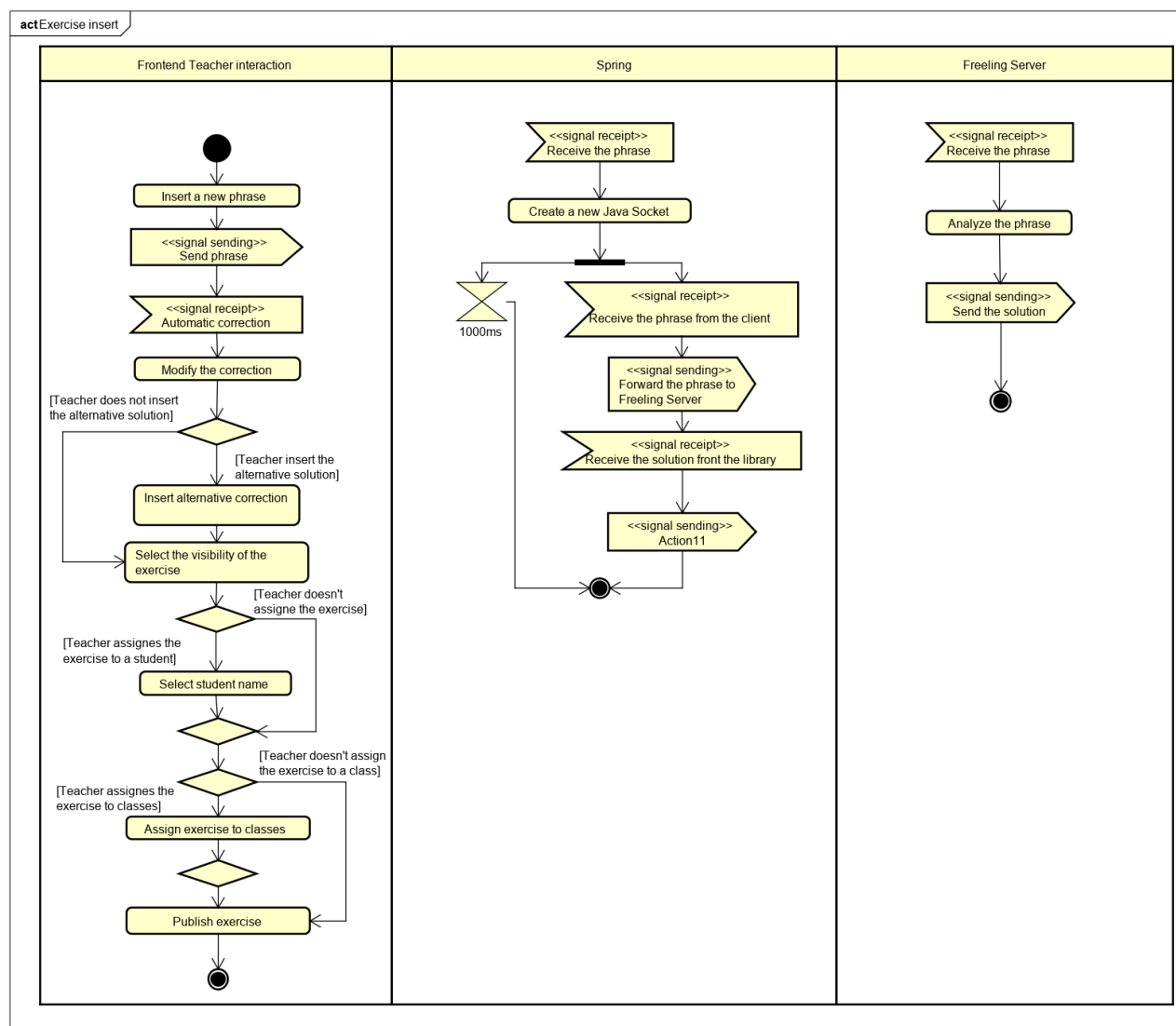


Figura 6: Exercise insert

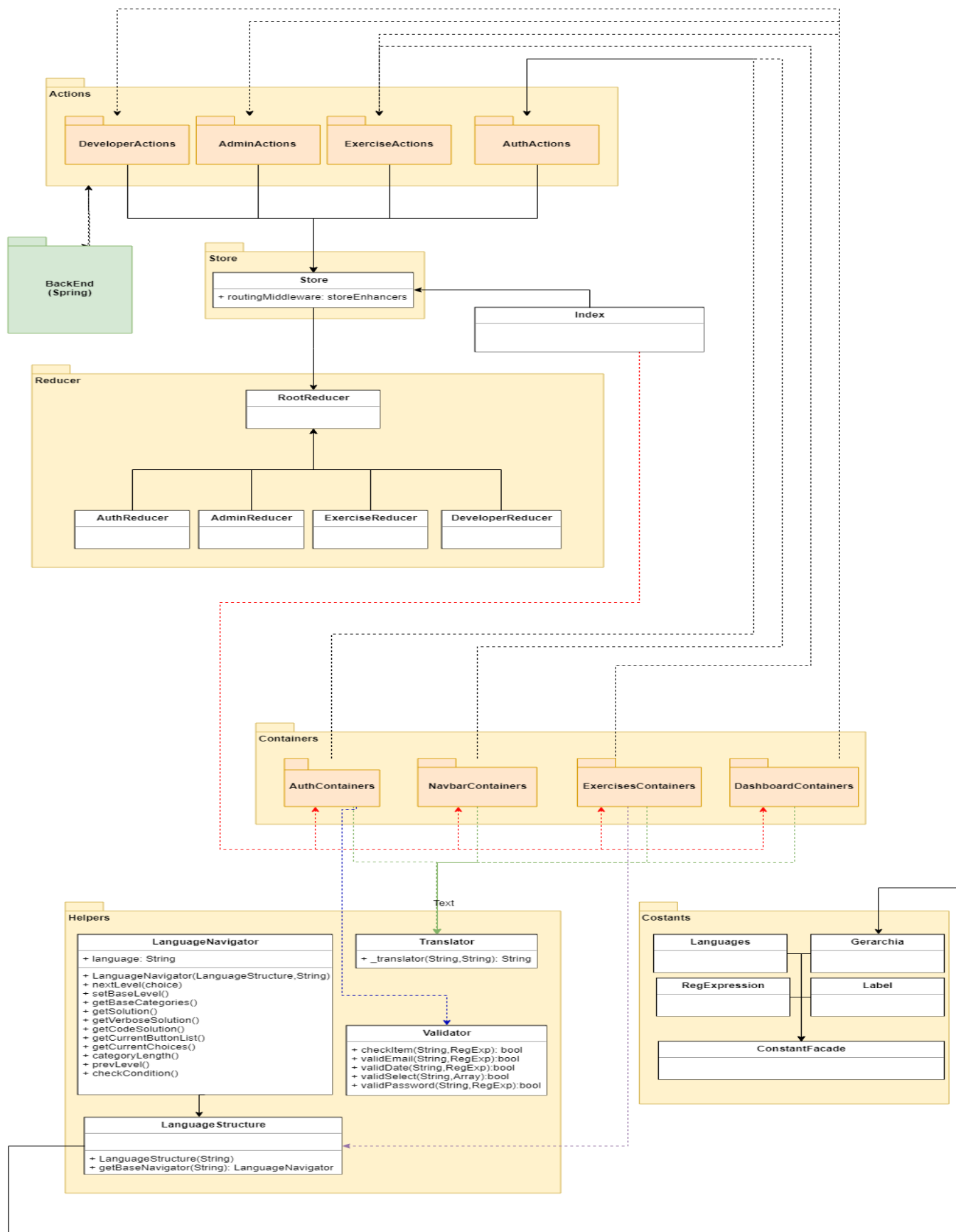


Figura 7: UML - FrontEnd

5.2 Login

Il diagramma di sequenza riportato qui di seguito raffigura il processo di login, durante il quale l'utente che vuole accedere può essere autenticato dal sistema.

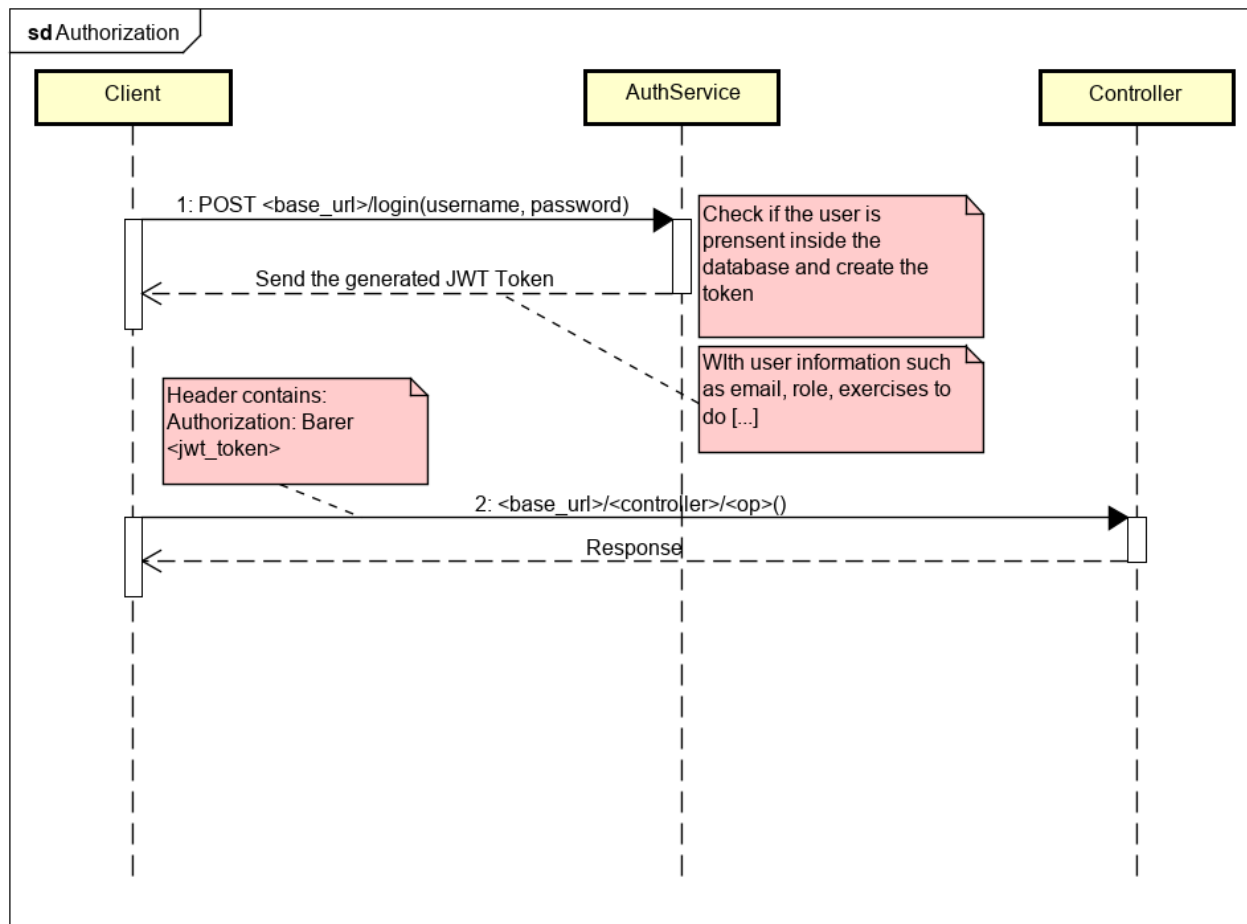


Figura 8: Authorization