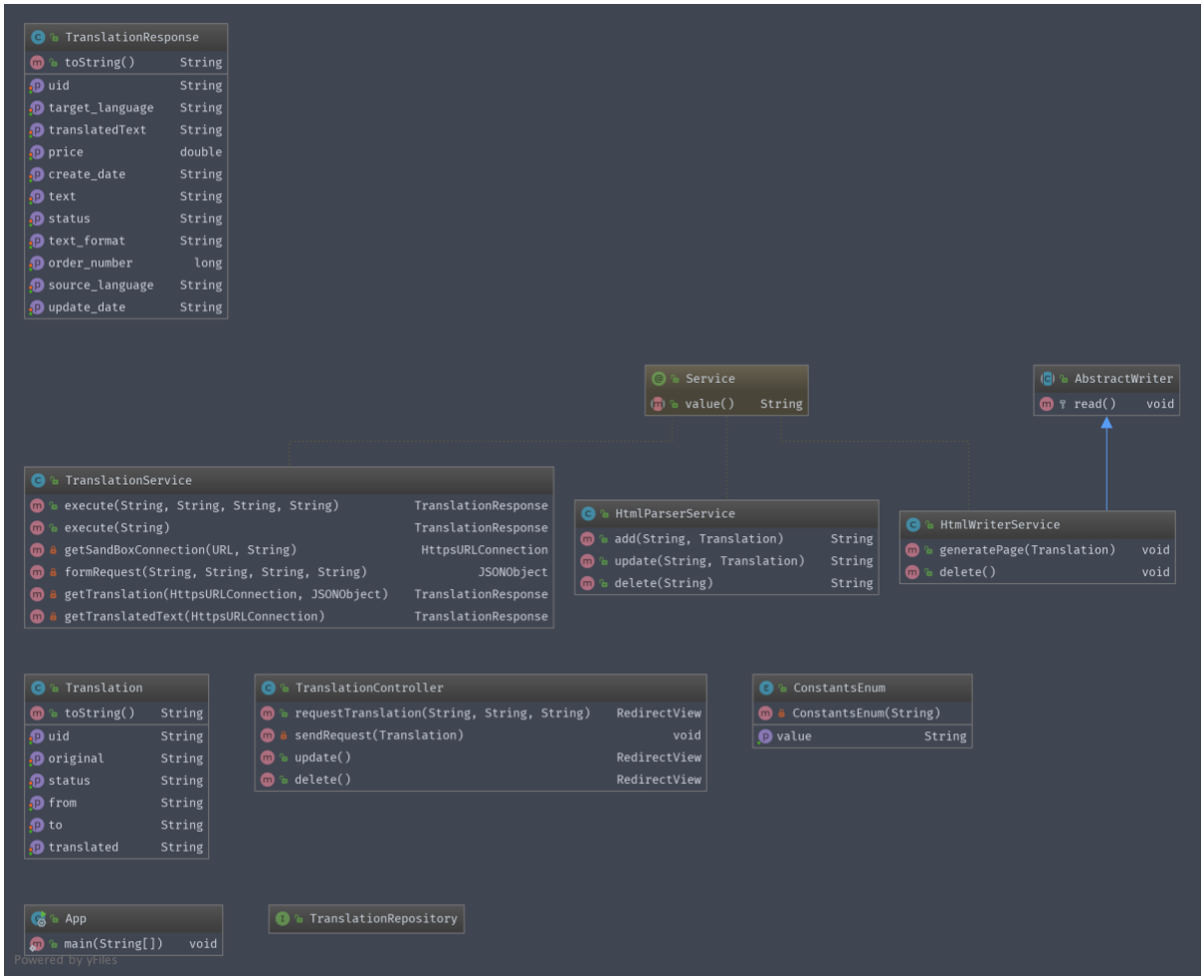
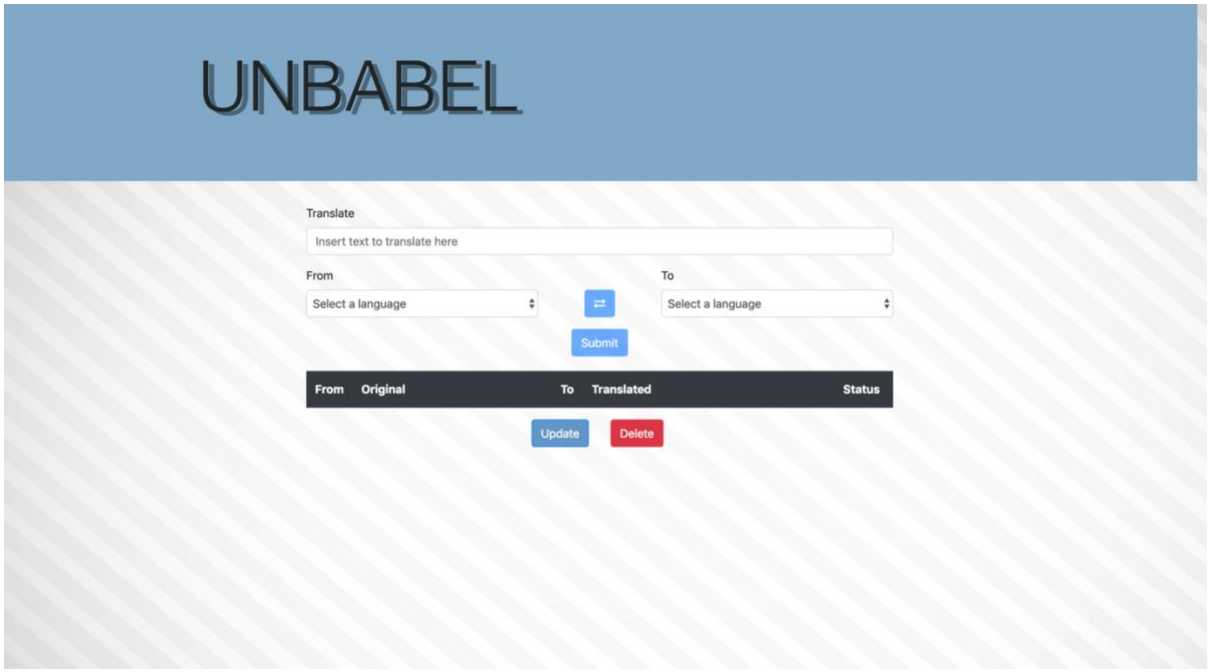
Java Coding Challenge

Data Model



Technical Scope

To start the application run the following command: ***mvn spring-boot:run*** and access the page at ***localhost:8080***.
This is a single page web application built with spring boot.



The user interface has an input text field, where the user should add the text to translate, and two picklist fields, where the user should pick the languages to translate from and to. There's also a button that automatically switches the languages from one field to the other.

The switch button is disabled until both languages are picked and is only enabled if the language pair is available in unbabel sandbox api.

Once the fields are properly filled the user should click the submit button.

The ***submit button*** will send a **post** request to the Translation controller with the request parameters previouslyadded by the user. Firstly, since the request took some time to process and I wanted to page to load fast, a model object *Translation* receives the request parameters and the html is rewritten in the *HtmlWriterService* with a new row added to the table, then an executor sends the request to the api and once it obtains a response (with the request uid) the *TranslationRequest* object is added to the table *translation_response* in the database.

To check if the status has been altered the user should click the update button.

Once the ***update button*** is clicked, a new **get** request, with the translation *uid* is sent to the api. If there were any changes to the object's TranslationResponse status, we update it in the database and create a new model object *Translation*, parse the html and alter the corresponding row.

When clicked, the ***delete button*** will delete every row in the html table, but all the translations are kept in the table *translation_response.*

Use Cases

This application could be used to translated user generated content, articles, emails, notes, etc.