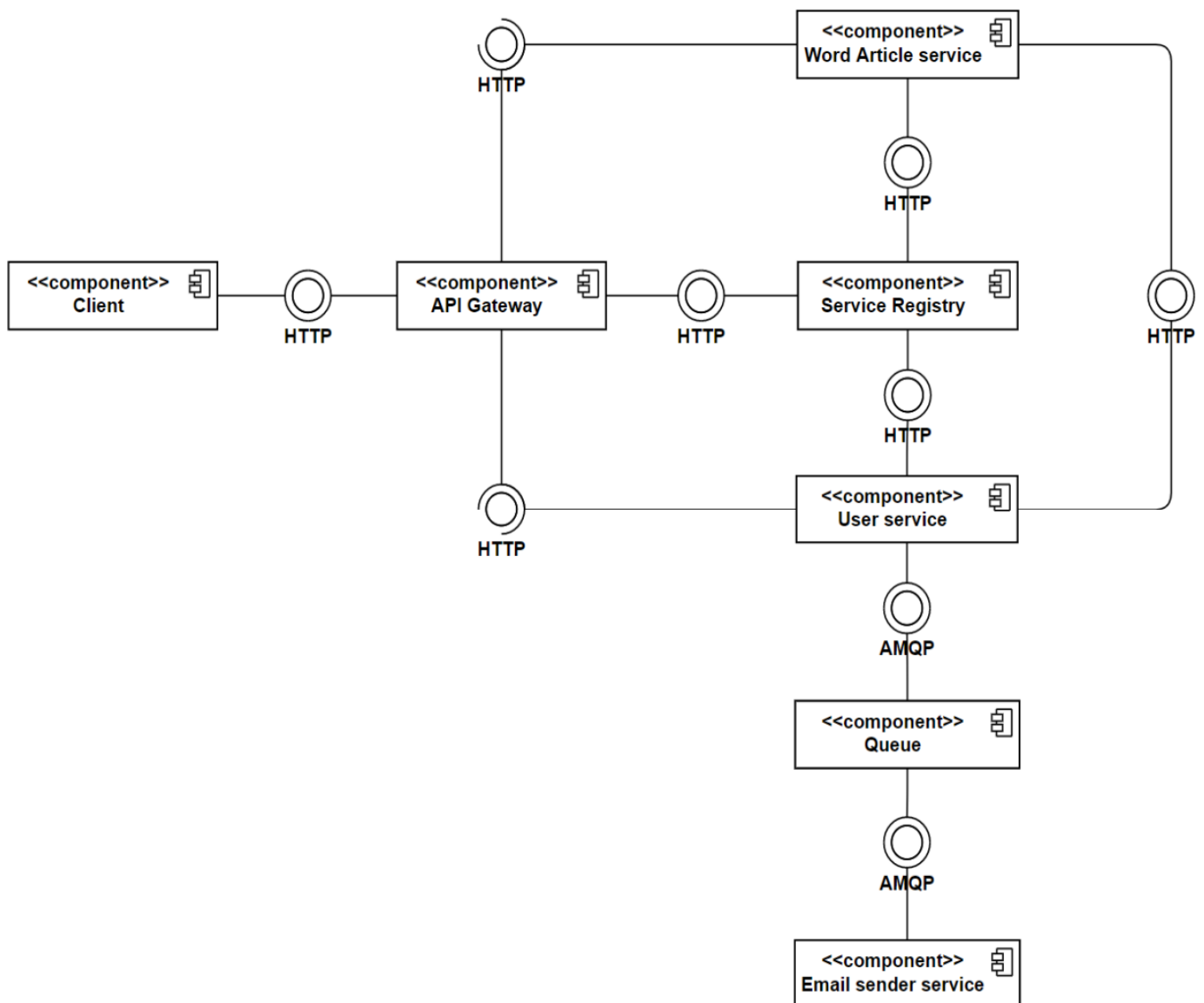


EDS

Скрипт для клонування всіх проектів

```
git clone git@gitlab.com:Moretti7/diploma-user-service.git
git clone git@gitlab.com:Moretti7/diploma-service-registry.git
git clone git@gitlab.com:Moretti7/diploma-email-sender.git
git clone git@gitlab.com:Moretti7/diploma-dictionary.git
git clone git@gitlab.com:Moretti7/diploma-client.git
git clone git@gitlab.com:Moretti7/diploma-api-gateway.git
```

Архітектура



Client – веб-клієнт, створений на React

API Gateway – Java Spring Boot додаток, який використовується production версією клієнта для викликів сервісів.

`Word article service` – Java Spring Boot додаток для роботи зі словниковими статтями, в якості бази даних використовується MongoDB.

`Service Registry` – Java Spring Boot додаток для спілкування інших додатків між собою. `Word Article service` надсилає запити до `User service`, використовуючи `Service registry` для отримання адреси сервісу.

`User service` – Java Spring Boot додаток для додання та видалення адміністраторів, а також для створення та валідації токенів доступу, в якості бази даних використовується MongoDB.

`Queue` – RabbitMQ, куда відправляються email з паролями адміністраторів.

`Email sender service` – Java Spring Boot додаток, який отримує email від `Queue` та відправляє сам email.

Docker Compose для RabbitMQ та MongoDB

```
version: "3"

services:
  mongo:
    image: mongo:5.0.3
    container_name: mongo
    ports:
      - 27017:27017
    volumes:
      - mongo-data:/data/db
  rabbit-mq:
    image: rabbitmq
    container_name: rabbitmq
    ports:
      - 5672:5672
    volumes:
      - rabbitmq-data:/var/lib/rabbitmq

volumes:
  mongo-data:
  rabbitmq-data:
```

Notes

1. Java додатки побудовані на Java 8, при використанні іншої версії можуть бути помилки, наприклад на 11 не працює `User service`.
2. Для локального запуску Java додатків треба вказувати `dev` для `spring.profiles.active`.
3. Проекти містять `README.MD` з переліком необхідних проперті/environment variables та команду для запуску.

4. Рекомендований порядок запуску додатків:

1. MongoDB, RabbitMQ
2. diploma-service-registry
3. diploma-api-gateway
4. diploma-email-sender, diploma-user-service, diploma-dictionary
5. diploma-client

Email sender service (diploma-email-sender)

Для роботи додатку необхідно надати значення для `email.sender.mail` та `email.sender.password`. Його єдина задача – відправка email з паролями.

User service (diploma-user-service)

Сервіс використовується для реєстрації адміністраторів системи, як звичайних, так і `root` адміністраторів, які можуть реєструвати інших адміністраторів. Іншою задачею сервісу є видача, оновлення та валідація токенів доступу.

При старті додаток бере значення з проперті `root-admins` та визначає, яких адміністраторів з `root` правами треба додати, а яких видалити. Валідні значення: comma separated email.

Наприклад, якщо в системи зареєстровані адміністратори `a@mail.com` та `b@mail.com`, а `root-admins` містить значення `a@mail.com,c@mail.com`, то `b@mail.com` буде видалений, а `c@mail.com` буде доданий до системи.

При реєстрації будь-якого адміністратора йому на пошту відправляється пароль, за яким він може увійти в систему. Але єдине, що може зробити адміністратор – змінити пароль. Після зміни пароля, йому будуть доступні інші функції. Тобто перелік дій наступний:

1. Реєстрація адміністратора.
2. Відправка email з паролем.
3. Логін.
4. Зміна пароля.

Word Article service (diploma-dictionary)

Використовується для управління словниковими статтями. Неаутентифікований користувач може шукати та переглядати статті, адміністратори можуть створювати статті, оновлювати, видаляти та завантажувати їх з `docx` файлу.

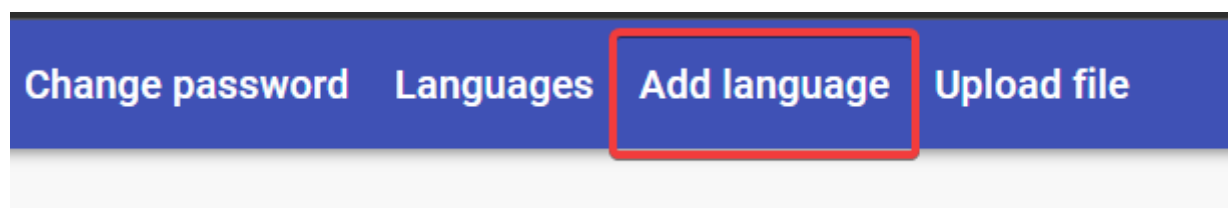
Endpoint, на який відправляється `POST` запит для створення словникової статті містить в собі мови словника, наприклад для українсько-німецького словника шлях запиту буде наступний: `/ukrainian/german`. Назви мов використовуються для визначення назви MongoDB колекції. Для українсько-німецького словника колекція буде називатись `ukrainian_german_articles`. В коді назва колекції визначається наступним чином

`MongoCollectionUtils.toCollectionName(languages)`. Також для кожного слова визначаються його n-грами – перетворення слова на групи, що складаються з літер цього слова. Наприклад: `абітурієнт` буде перетворено на `абі біт іту тур урі ріє ієн ент`. N-грами використовуються при пошуку словникових статей з використанням [текстового індексу MongoDB](#). Створення текстового індексу можна знайти в `WordArticleMongoRepository.save()`. Також для кожної статті визначає "головне слово" – перше слово першої мови, воно використовується для сортування словникових статей за алфавітом.

Ендпоінт для завантаження файлу визначений в `WordArticlesUploadController`.

Diploma Client

Для роботи необхідно додати мінімум дві мови.



Для створення мови необхідно вказати її назву та літери алфавіту через кому.

Language Ukrainian

Alphabet

А, Б, В, Г, Г', Д, Е, Є, Ж, З, И, І, Ї, Й, К, Л, М, Н, О, П, Р, С, Т, У, Ф, Х, Ц, Ч, Ш, Щ, Ъ, Ю, Я

SAVE

Після додання двох мов повинен з'явитись компонент для вибору мов словника

Dictionary: Ukrainian ↺ German ▼

А Б В Г Г Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ю Я
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü SS

Для імпорту статей з `docx` файлу є спеціальний пункт меню

Languages Add language **Upload file**

Dictionary: Ukrainian ↺ German ▼

А Б В Г Г Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ю Я
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü SS

No file selected

SELECT FILE

UPLOAD FILE

Обираєш файл та завантажуєш його. Парсинг файлу проходить в рамках HTTP запиту та займає певний час, тому треба почекати. Коли файл буде успішно оброблений у правому верхньому куту з'явиться відповідне повідомлення та кнопка `UPLOAD FILE` стане знову доступна.

Dictionary: Ukrainian ↺ German ▼

А Б В Г Г Д Е Є Ж З И І Ї Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ю Я

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü SS

Selected file: Slovnyk_04_10.docx

SELECT FILE

UPLOAD FILE



Додання адміністратора відбувається шляхом вказання його email.

Create article Administrators **Add administrator** Change password Languages Add language Upload file

Email

CREATE

Add administrator Change password Languages Add language Upload file Logout





Admin successfully created, please wait email with password

Email test@gmail.com

CREATE



Search...



Diploma admin password

to: <test@gmail.com> a few seconds ago

From: <04b12abfa1c46e>

To: <test@gmail.com>

Show Headers

HTML

HTML Source

Text

Raw

Your password SPZy1FIz4T61E7uzY9IW

Видалення адміністраторів відбувається на сторінці адміністраторів

Create article Administrators Add administrator Change password Languages Add language Upload file

Dictionary: Ukrainian German

А Б В Г Г Д Е Є Ж З И І Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ю Я

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü SS

abc@gmail.com

DELETE

test@gmail.com

DELETE

Для оновлення або видалення словникової статті, на сторінці з переліком словникових статей треба натиснути на статтю й тоді з'явиться кнопки для відповідних дій

⚠ АБІТУРІЄНТ

⚠ Abiturient

⚠ АБИТУРІЄНТ

ч (-а)

той, хто вступає до вищого або середнього спеціального навчального закладу: *заява абітурієнта*.

⚠ Abiturient

m (-en, -en)

випускник середньої школи: *die Abiturienten feiern den Schulabschluss*.

GO TO ARTICLE

EDIT

REMOVE

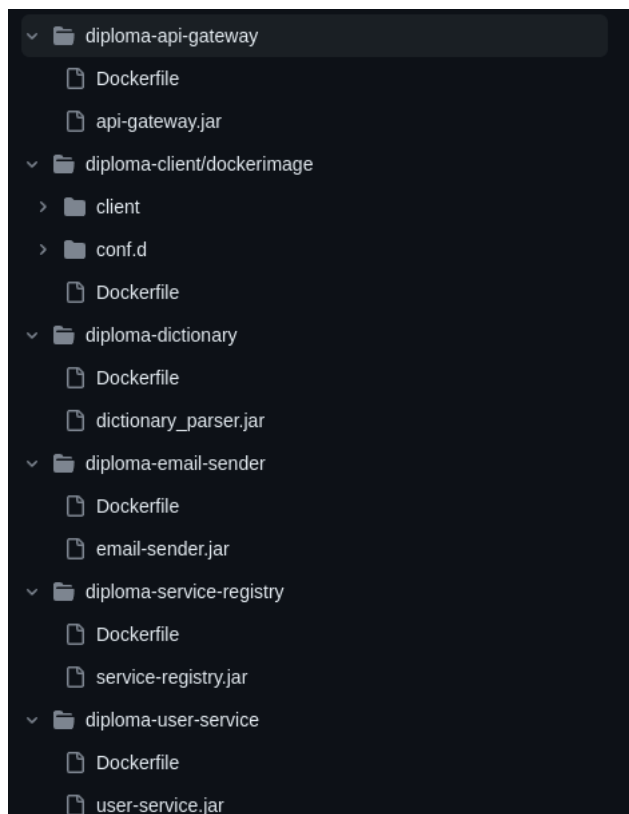
Файл з кодом, для відправки http запитів знаходиться у файлі `api.ts`.

Доповнення:

Можна заклонувати оновлений проект з додатковими файлами та описом одним паком:

```
git clone https://github.com/vlad-lisnenko/diploma_eds.git
```

Для деплою проекту на сервер(на прикладі ZOMRO) потрібно заклонити свій проект на сервер, можна не увесь, а самі файли.jar(збілдені проекти) та власне dockerfile.



Після цього виконуємо команди для docker-compose в терміналі на сервері.

```
docker-compose build
```

```
docker-compose run
```

Щоб конфігурувати з'єднання серверу з нашим клієнт додатком, потрібно вказати порт через яке буде відбуватися прослуховування.

```
client:
  build:
    context: ../diploma-client/dockerimage
  container_name: client
  ports:
    - "5953:80"
    - "443:443"
```

В нашому випадку порт сервера(зовнішній порт): 5953

Вхід на сайт виконуємо через ір сервера та порт

Приклад: <http://212.8.244.225:5953/>

Щоб відновити бекап бази даних як для запуску локально так і на сервері виконуємо наступні кроки:

1. В проєкті, міститься файл бекапу edsdump-mongo-5.0.3.gz коли контейнер бази mongo запущено, через термінал перемістити цей файл до контейнеру

```
docker cp edsdump-mongo-5.0.3.gz mongo:/
```
2. Виконати команду для відновлення даних

```
docker exec -it mongo mongorestore --archive=edsdump-mongo-5.0.3.gz
```