

Asciiidoc

Содержание

1. Зависимости сервера	1
2. Зависимости клиента	2
3. Использование	2
4. Документы	3

Github: <https://github.com/Ddudde/DipvLom>

Публикации в HTML и PDF форматах находятся в /project/server/docs/asciiDoc/build

Основное описание проекта здесь. Также есть дополнительные:

Также в /project/server/docs/start-web.bat можно стартовать Python http server для просмотра документации

- Swagger:
- JavaDoc: <http://localhost:9001/javaDoc/>
- AsciiDoc: <http://localhost:9001/asciiDoc/build/ReadMe.html>

Сервер:

Находится в /project/server

<http://localhost:8080>

Java, Spring, REST + Server Sent Events

Клиент:

Находится в /project/client

<http://localhost:3000/DipvLom>

Github Pages: <https://ddudde.github.io/DipvLom/>

JS, React, Redux

1. Зависимости сервера



БД H2 для тестирования, а PostgreSQL для остального.

IntelliJ IDEA в качестве IDE

BD: H2

http://localhost:8080/console_db

1. H2
2. PostgreSQL
3. Gradle
4. Spring Boot

5. Spring Boot Test
6. Spring Security
7. JavaMail API
8. JavaFaker
9. JUnit
10. Mockito
11. AsciiDoctor Gradle Plugin
12. Launch4j
13. Server-Sent Events
14. Hibernate
15. Lombok
16. Spring Data JPA

2. Зависимости клиента



WebStorm в качестве IDE

1. Github Pages
2. Single Page Application
3. Progressive Web App
4. React
5. Redux
6. React Router
7. Firebase Cloud Messaging

3. Использование

Запускаем сервер

И запускаем клиент по одному из адресов: <http://localhost:3000/DipvLom>
, <https://ddudde.github.io/DipvLom/>

Выбираем роль, допустим администратора

И авторизируемся

Логин: nm12

Пароль: 1111

Радуетесь!

4. Документы

JavaDoc и AsciiDoc генерируются при помощи Gradle команды:

```
gradle javadoc  
gradle asciidoctor
```

todo - что нужно доделать

beenDo - что уже сделано