**Приложение**

1. Фронт-енд был реализован с использованием, css-фреймворка bulma.io.
2. Были использованы собственные стили (файл style.css).
3. Для редактора кода был использован ace code editor (темы solarized-dark, solarized-light).
4. Также в качестве textarea был использован webcomponent paper-textarea (polymer). Остальной код был написан на чистом js (внутри тэгов script, файл script.js).
5. В качестве обработчика post-запроса с фронт-енда – сервер на kotlin с использованием фреймворка vertx. Был использован gradle build tool. Посредством gradle plugin shadow (http://imperceptiblethoughts.com/shadow/) создается fat-jar. Используется порт 8080. Хост – localhost.
6. Используется Apache FreeMarker template engine (файл index.ftl).
7. Обработанный запрос с фронт-енда сериализуется в json (для этих целей создается клиент) и отправляется на другой сервер (по идее реализованный другой командой, но нами на всякий случай реализован фейковый сервер на golang). Порт - 8000, хост – localhost.
8. В свою очередь фейковый сервер возвращает json строку. Эта строка возвращается на фронт-енд, где в свою очередь обрабатывается.
9. Все форматы данных для сериализации и десериализации находятся в файлах Code.kt, CompilationResult.kt. Проверено было посредством класса (io.vertx.core.json.Json (методы Json.decodeValue(…) и Json.encode(…)))
10. Все статичные файлы сохранены в папке “resources/static”
11. Сервер сохранен в формате .jar. Фейковый сервер в формате .exe
12. Везде используется протокол http. “Content-type” в случае отправки данных с фронт-енда –“application/x-www-form-urlencoded”, в остальных случаях – “application/json”
13. Ошибки выделяются цветом: Error: #F44336; Warnings: #FF6F00;
14. На всякий пожарный исходные коды сохранены в репозиториях github студента М1-16.