|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  «МИРЭА **–** Российский технологический университет»  РТУ МИРЭА | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Отчет по практической работе №3**

по дисциплине «Технологии разработки программных приложений»

**Тема практической работы:** «Системы сборки»

|  |  |  |
| --- | --- | --- |
| **Выполнил:**  Студент группыИКБО-42-23 | | Голев С.С. |
| **Проверил:** | Доцент кафедры МОСИТ,  кандидат технических наук, доцент Жматов Д.В. | | |

# СОДЕРЖАНИЕ

[ЗАДАНИЕ 3](#_Toc196356286)

[ХОД РАБОТЫ 4](#_Toc196356287)

[КОНТРОЛЬНЫЕ ВОПРОСЫ 9](#_Toc196356288)

[ВЫВОД 10](#_Toc196356289)

# ЗАДАНИЕ

Найти отсутствующую зависимость и указать ее в соответствующем блоке в build.gradle, чтобы проект снова начал собираться

В некоторых классах поправить имя пакета

Собрать документацию проекта, найти в ней запросы состояния и сущности по идентификатору

Собрать jar со всеми зависимостями (так называемый UberJar), после чего запустить приложение. По умолчанию, сервер стартует на порту 8080.

Запросить состояние запущенного сервера (GET запрос по адресу http://localhost:8080)

Запросить сущность по идентификатору (GET запрос по адресу: http://localhost:8080/сущность/идентификатор)

Идентификатором будут 3 последних цифры в серийном номере вашего студенческого билета.

В задаче shadowJar добавить к jar-файлу вашу фамилию

Выполнить задачу checkstyleMain. Посмотреть сгенерированный отчет. Устранить ошибки оформления кода.

Персональный вариант: репозиторий: https://github.com/rtu-mirea/trpp-second-2, сущность ru.mirea.entity.Student

# ХОД РАБОТЫ

Для нахождения зависимостей просмотрим файлы, на наличие отсутствующих зависимостей. В файле StudentController.java была найдена недостающая зависимость.

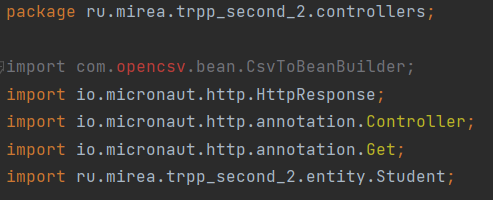


Рисунок 1 – Зависимости контроллера

Добавим эту зависимость в проект.

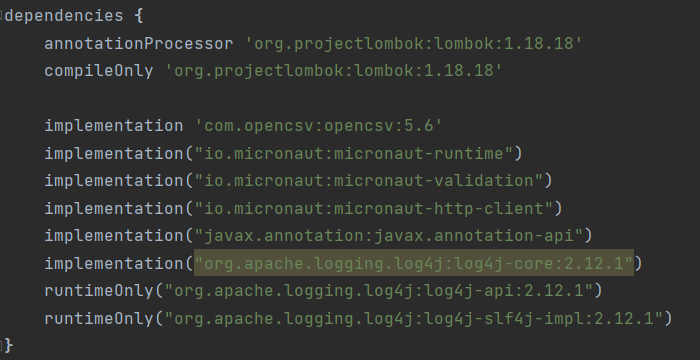


Рисунок 2 – Зависимости проекта

Проверим имена пакетов на корректность.

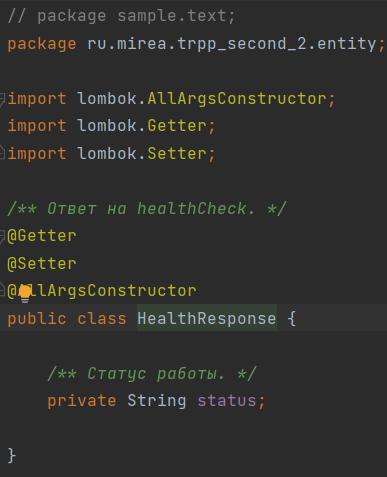


Рисунок 3 – Изменённое имя пакета

Соберём документация проекта.

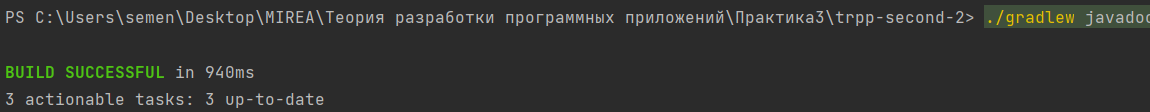


Рисунок 4 – Сбор документации

Запросы состояний сервера:

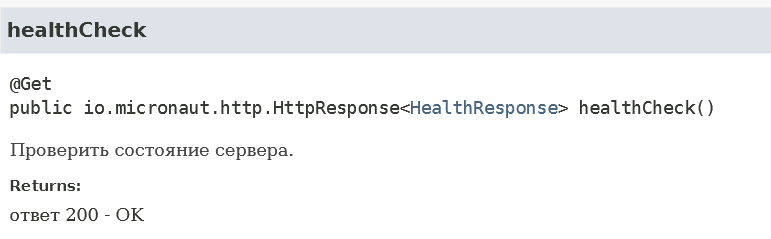


Рисунок 5 – Запрос состояния сервера.

Соберём jar-файл и запустим пролижение.

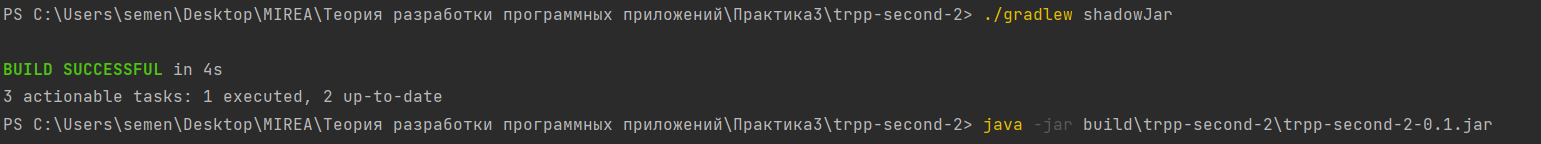


Рисунок 6 – Сбор jar-файла и запуск приложения

Запросим состояние сервера по порту 8080.

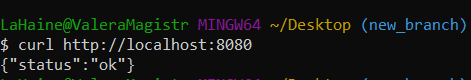


Рисунок 7 – Состояние сервера

Запросим состояние сущности.

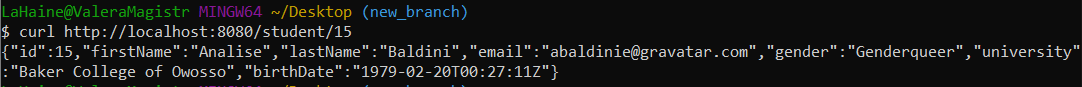


Рисунок 8 – Состояние сущности

Выполним задачу checkstyleMain.

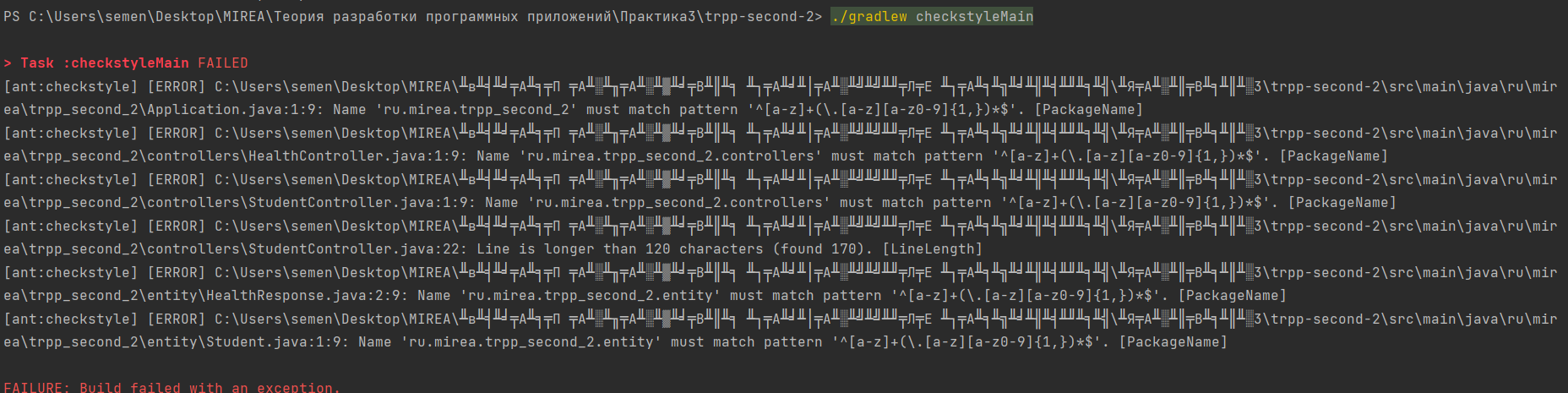


Рисунок 9 – Найденные ошибки

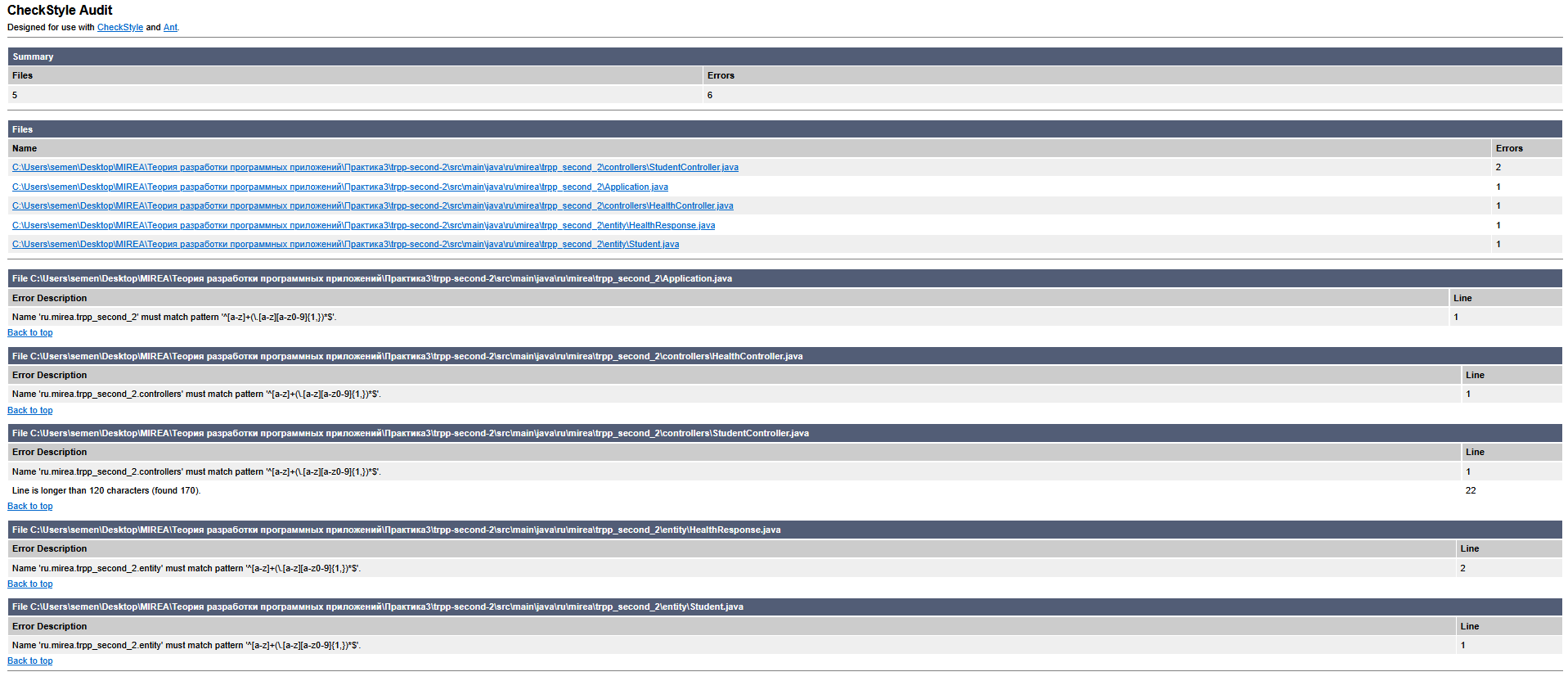


Рисунок 10 – Файл с описанием ошибок

Исправим данные ошибки.

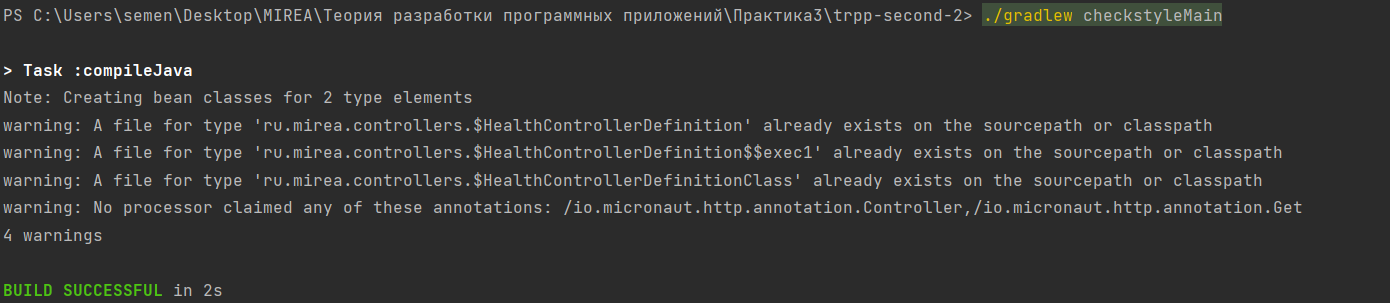


Рисунок 11 – Исправление всех ошибок

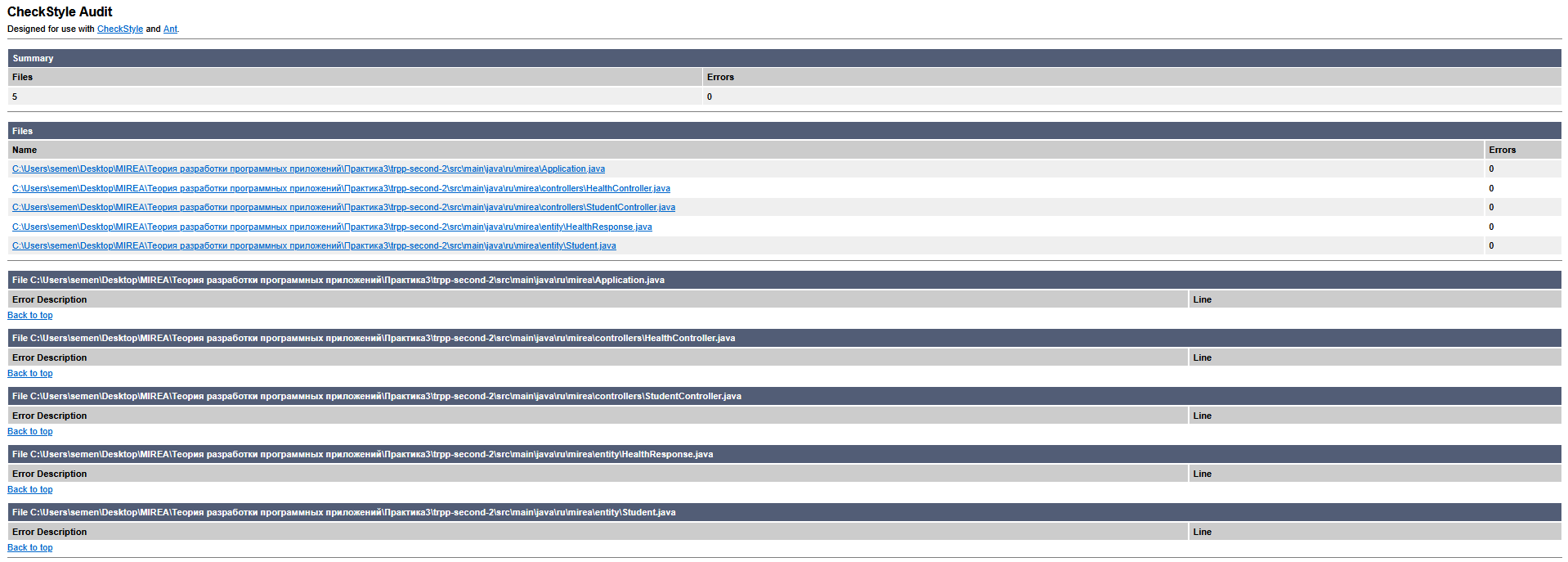


Рисунок 11 – Файл с описанием ошибок

# КОНТРОЛЬНЫЕ ВОПРОСЫ

2. Что такое система сборки? – инструмент, который автоматизирует процессы компиляции, тестирования, упаковки и развёртывания программного обеспечения.

3. Что такое репозиторий? – это хранилище, где размещаются: код проекта, артефакты зависимостей.

4. Как указать зависимости проекта? – В файле конфигурации системы сборки используется блок dependencies, где зависимости указываются через ключевое слово implementation**.**

17. Что такое lombok? – библиотека для Java, которая автоматически генерирует шаблонный код через аннотации.

18. Что такое postman? – инструмент для тестирования API. Отправляет HTTP запросы, проверяет ответы сервера, сохраняется коллекцию запросов.

19. Что такое аннотация в Java? – метка, которая добавляет метаданные к классам, методам или полям.

# ВЫВОД

В ходе выполнения практической работы были освоены ключевые аспекты работы с Gradle, Gradle доказал свою эффективность как инструмент для автоматизации жизненного цикла разработки. Применение Gradle в будущих проектах позволит сократить время на рутинные операции и сосредоточиться на логике приложения.