|  |  |  |
| --- | --- | --- |
|  |  |  |
| 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 Итоговому проекту (Часть 2)**

по дисциплине

«**Технология разработки программных приложений**»

**Тема: «Погодный телеграм-бот»**

|  |  |
| --- | --- |
| **Выполнили:**  Студенты группыИКБО-36-22 | Ярош В.Э, Утенков Ю.Ю,  Захаров А.А |
| **Проверил:** | ассистент Петрова А.А. |

МОСКВА 2024 г.

**СОДЕРЖАНИЕ**

[1 Написание файла README.md 3](#_Toc165735338)

[2 Выбор технологий и разработка архитектуры 3](#_Toc165735339)

[3 Настройка системы сборки 5](#_Toc165735340)

[4 Система сборки для сборки документации 7](#_Toc165735341)

[ВЫВОД 8](#_Toc165735342)

# **1 Написание файла README.md**

Был написан файл README.md с описанием зависимостей и инструкцией по запуску, что показано на Рисунке 1.

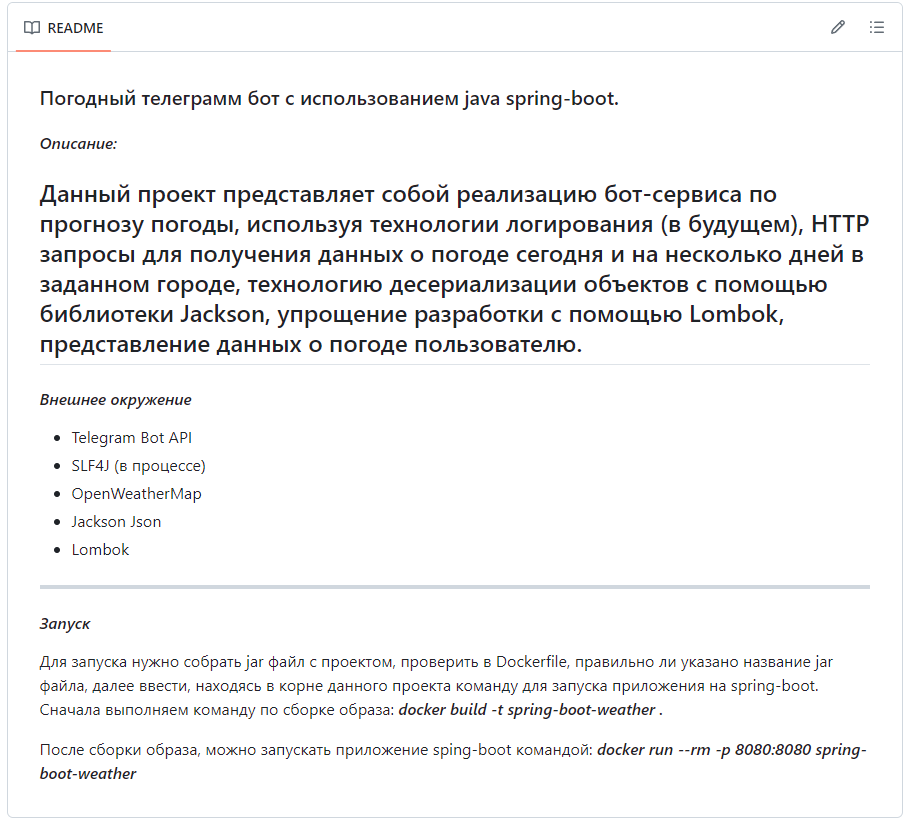


Рисунок 1 – Файл README.md

# **2 Выбор технологий и разработка архитектуры**

Для разработки приложения телеграм-бот на Java, были выбраны следующие технологии:

Java: Это основной язык программирования для создания приложения. Он предоставляет гибкость, безопасность и производительность, необходимые для разработки сложного и надежного приложения.

Telegram Bot API: Для взаимодействия с Telegram платформой и создания бота я бы использовал Telegram Bot API. Это позволяет создавать ботов и обрабатывать сообщения от пользователей.

Java Spring Boot: Для реализации бота как веб-приложения на языке Java я бы выбрал Spring Boot. Spring Boot обеспечит простоту настройки и развертывания приложения, а также предоставит мощные инструменты для разработки REST.

OpenWeatherMap API: Для получения информации о погоде по запросу пользователя я бы воспользовался OpenWeatherMap API или аналогичным сервисом. Это API предоставляет доступ к актуальным метеорологическим данным, которые могут быть интегрированы в боте.

HTTP клиент (например, Apache HttpClient): Для отправки HTTP запросов к OpenWeatherMap API я бы использовал HTTP клиент на Java, такой как Apache HttpClient. Эти инструменты облегчают взаимодействие с внешними веб-сервисами.

Dependency Injection Framework (например, Spring DI): Для управления зависимостями и инверсии управления в приложении я бы использовал Dependency Injection фреймворк, например, встроенный в Spring Framework.

JUnit или другие фреймворки для тестирования: Для тестирования функционала бота я бы написал модульные тесты с использованием JUnit или других фреймворков для тестирования Java приложений.

Логирование (например, SLF4J) [в процессе разработки]: Для логирования работы бота я бы воспользовался библиотеками логирования, такими как SLF4J с реализацией Logback, чтобы обеспечить контроль и отслеживание работы приложения.

Docker - это платформа для автоматизации развертывания, тестирования и развертывания приложений в средах с различными операционными системами и архитектурами. Он позволяет создавать образы контейнеров, которые содержат все необходимые компоненты для работы приложения, и управлять ими с помощью оркестраторов, таких как Docker Compose.

Maven - это система сборки позволяет автоматизировать процесс компиляции, тестирования и упаковки приложений. Maven поддерживает многомодульные проекты, позволяя разработчикам организовывать свои проекты в модули и управлять зависимостями между ними.

Была разработана архитектура приложения, что показано на Рисунке 2.

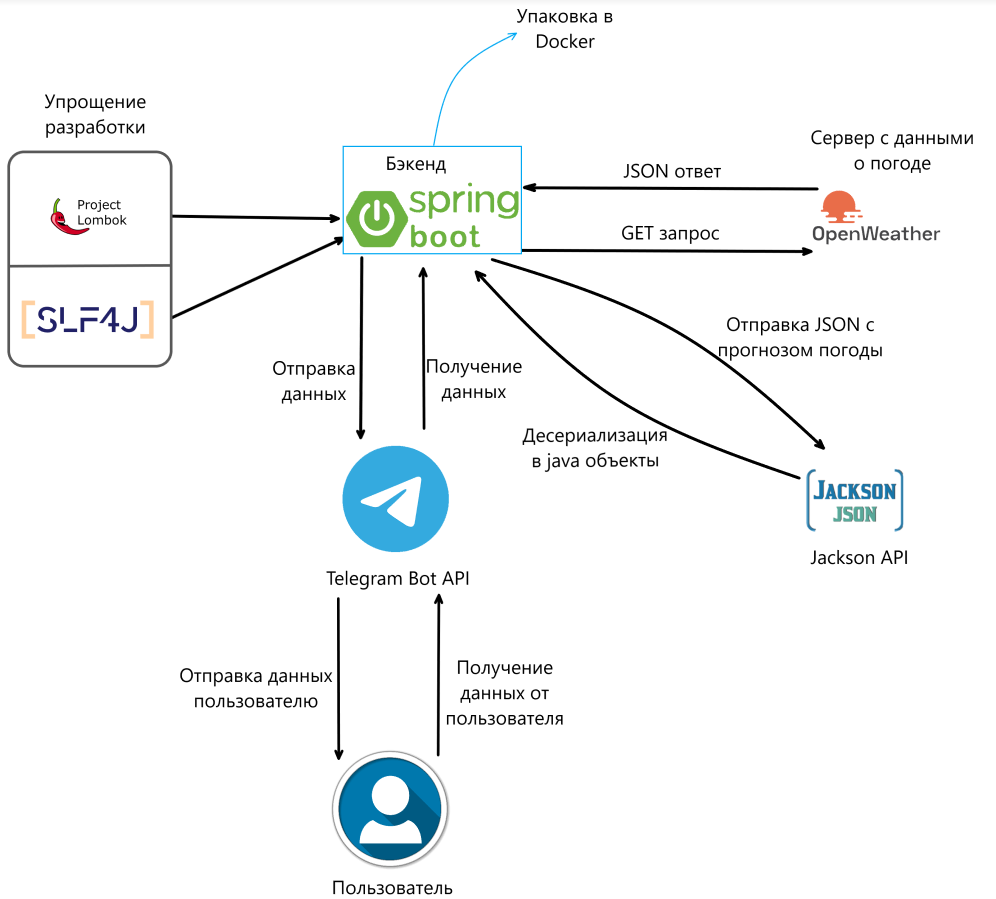


Рисунок 2 – Архитектура приложения

# **3 Настройка системы сборки**

Для сборки приложения был написан файл pom.xml (Рисунок 3-4).

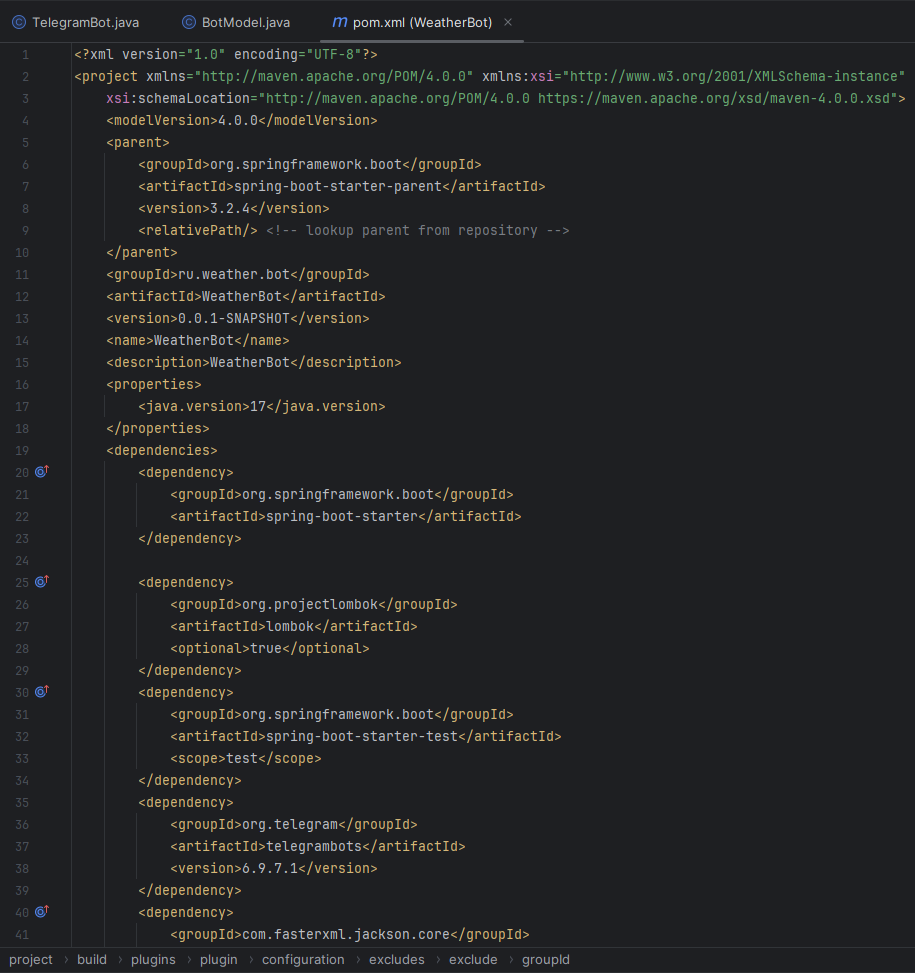


Рисунок 3 – Файл pom.xml

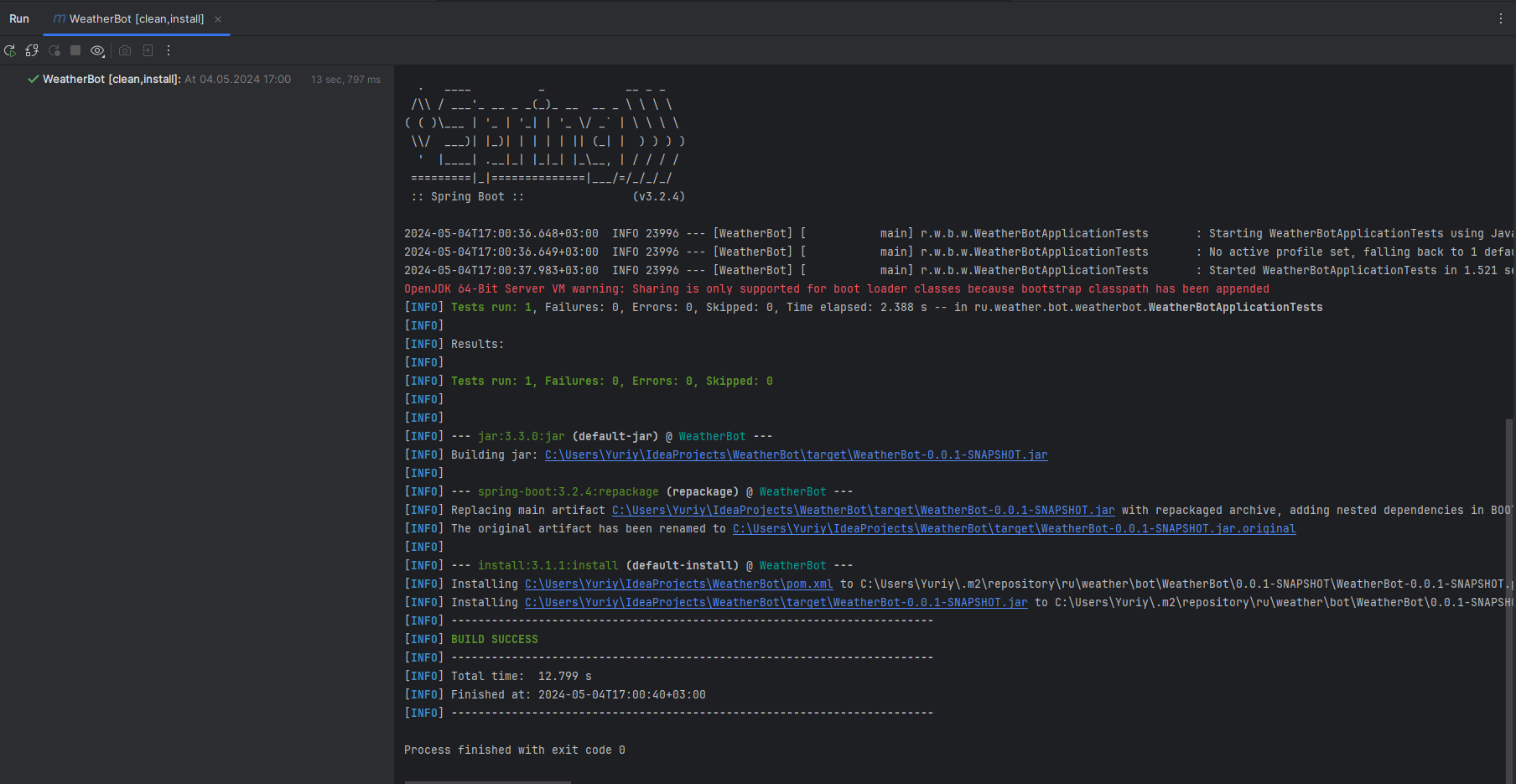


Рисунок 4 – Сборка проекта

# **4 Система сборки для сборки документации**

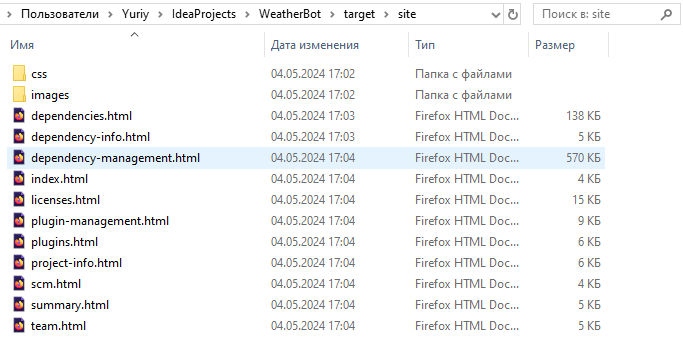
Была выполнена команда site.maven на сборку документации, сгенерированная документация показана на Рисунке 5.

Рисунок 5 – Сгенерированная документация

Документация в браузере показана на Рисунке 6.

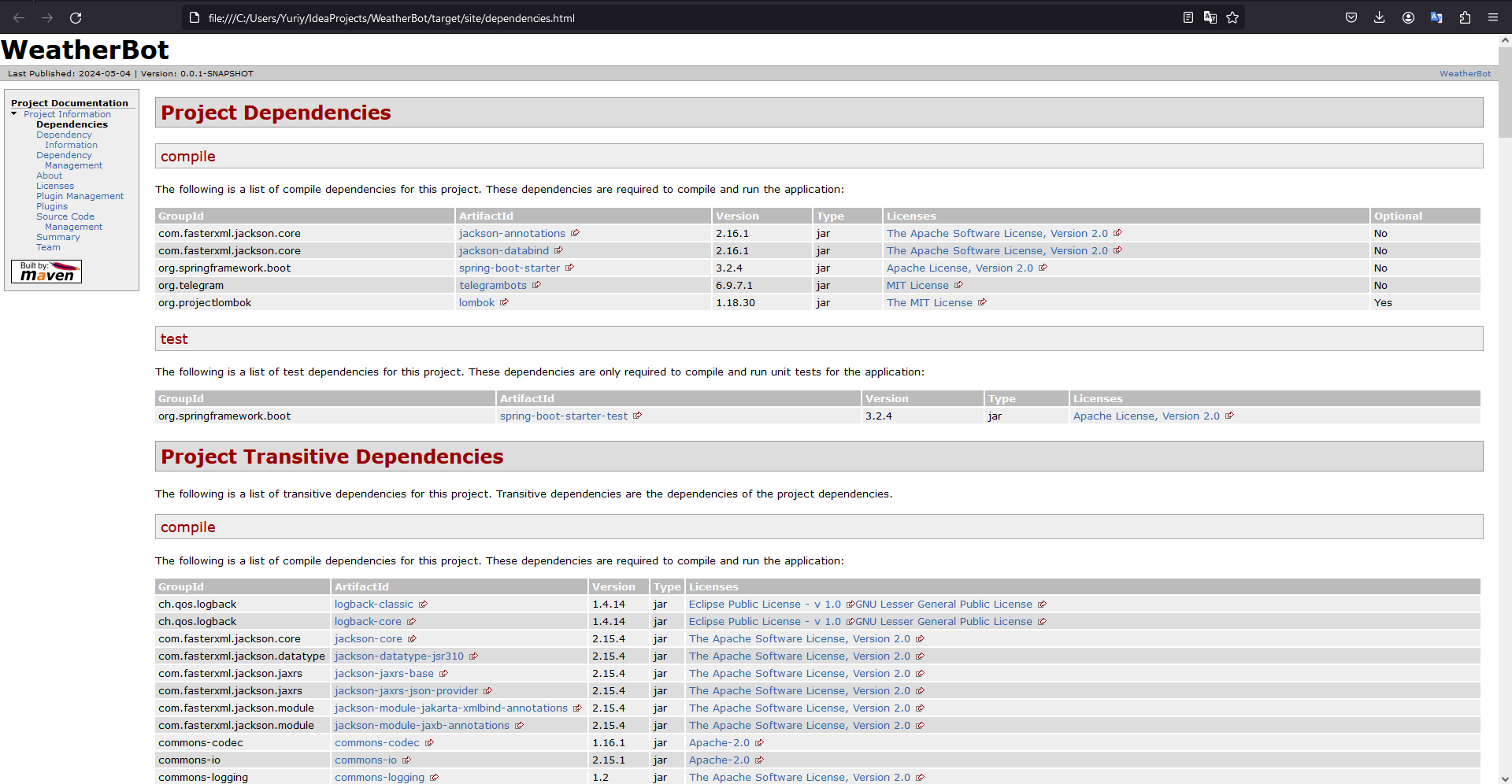


Рисунок 6 – Документация в браузере

# **ВЫВОД**

В ходе данной части проекта удалось ознакомиться с Maven, все задания выполнены в полном объеме.