Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ

УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированной обработки информации (АОИ)

**BACKEND-ПРИЛОЖЕНИЕ «СПИСОК ДЕЛ»**

Отчет о выполнении лабораторной работы

по дисциплине «Разработка интернет-приложений»

Студент гр. 429-3

\_\_\_\_\_\_\_\_\_\_ Бабец А. А.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2021 г.

Ассистент каф. АОИ

\_\_\_\_\_\_\_\_\_\_ Владимиров М. В.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2021 г.

Томск 2021

**Оглавление**

[1 Введение 3](#_Toc83149604)

[2 Основная часть 4](#_Toc83149605)

[2.1 Описание используемой среды программирования 4](#_Toc83149606)

[2.2 Описание метода решения задачи 4](#_Toc83149607)

[2.3 Описание фрагментов кода 5](#_Toc83149608)

[2.4 Описание функциональных возможностей разработанного приложения. 6](#_Toc83149609)

[3 Заключение 7](#_Toc83149611)

[Приложения 8](#_Toc83149612)

# Введение

Целью лабораторной работы является запуск и проверка работоспособности backend приложения с использованием технологии docker.

Задачи лабораторной работы следующие:

* Запуск и тестирование backend приложения в docker контейнере.

# Основная часть

## Описание стека технологий

Docker — это проект с открытым исходным кодом для автоматизации развертывания приложений в виде переносимых автономных контейнеров, выполняемых в облаке или локальной среде.

Также писать код будем в Visual Studio Code, так как это бесплатная IDE с широким функционалом.

## Описание метода решения задачи

Для контейнеризации необходимо установить Docker и создать подходящий контейнер.

## Описание фрагментов кода

Для создания контейнера используется Dockerfile.

FROM node**:**12**-**slim

WORKDIR **/**dock

COPY package**.**json **/**dock

RUN npm install

COPY **.** **/**dock

CMD **[**"npm"**,** "start"**]**

## Описание функциональных возможностей разработанного приложения

Запустив приложение, мы можем через Postman отправлять GET,POST, PUTT,DELETE запросы на наш сервер, и сервер возвращает json файлы с результатом запросов.

Скриншоты приложения представлен в приложении Б.

# Заключение

В практической работе были достигнуты поставленные задачи. Мы познакомились с ПО Docker.

# Приложения

# Приложение А

# (обязательное)

# Листинг кода

Dockerfile

FROM node**:**12**-**slim

WORKDIR **/**dock

COPY package**.**json **/**dock

RUN npm install

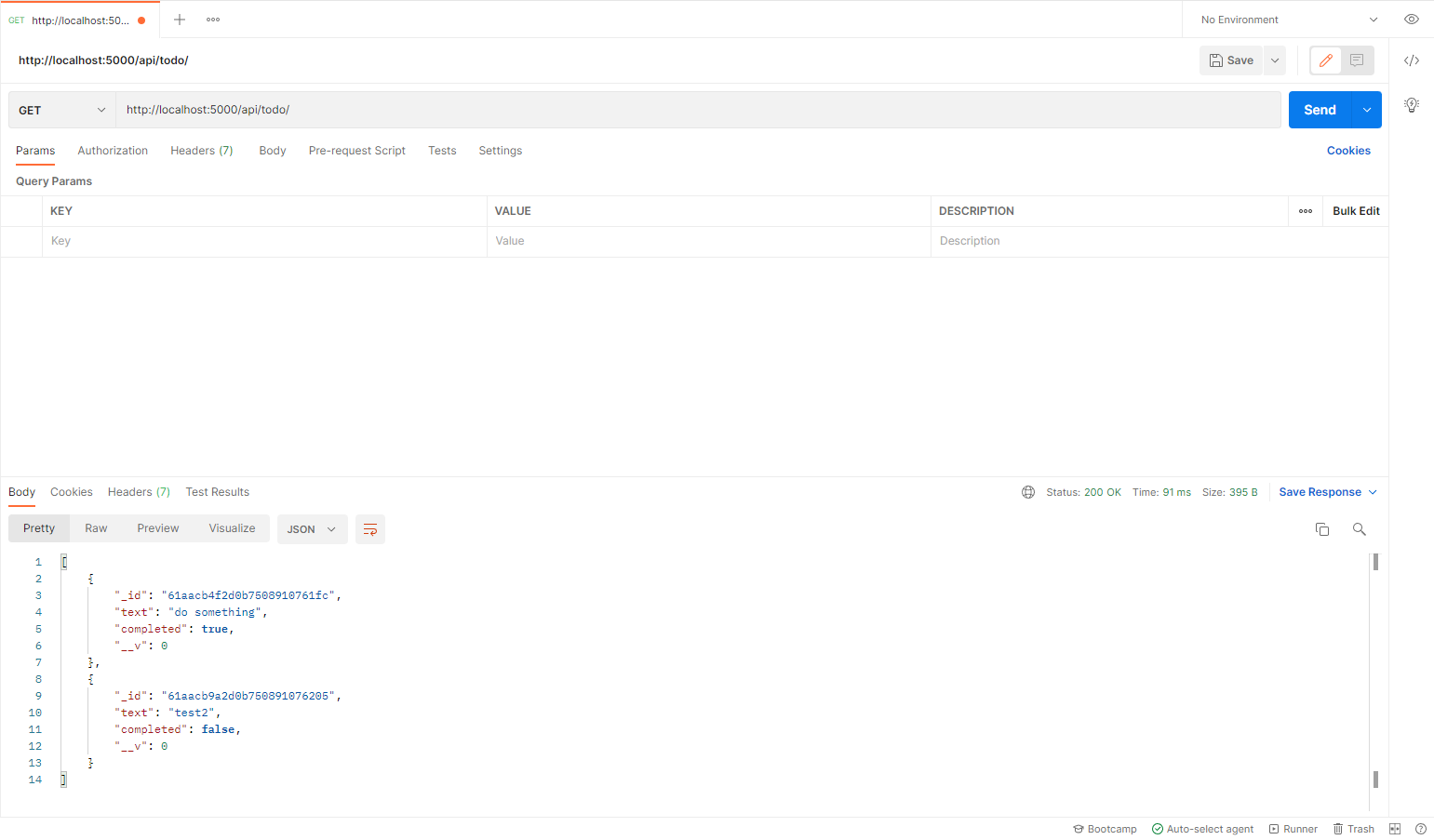
COPY **.** **/**dock

CMD **[**"npm"**,** "start"**]**

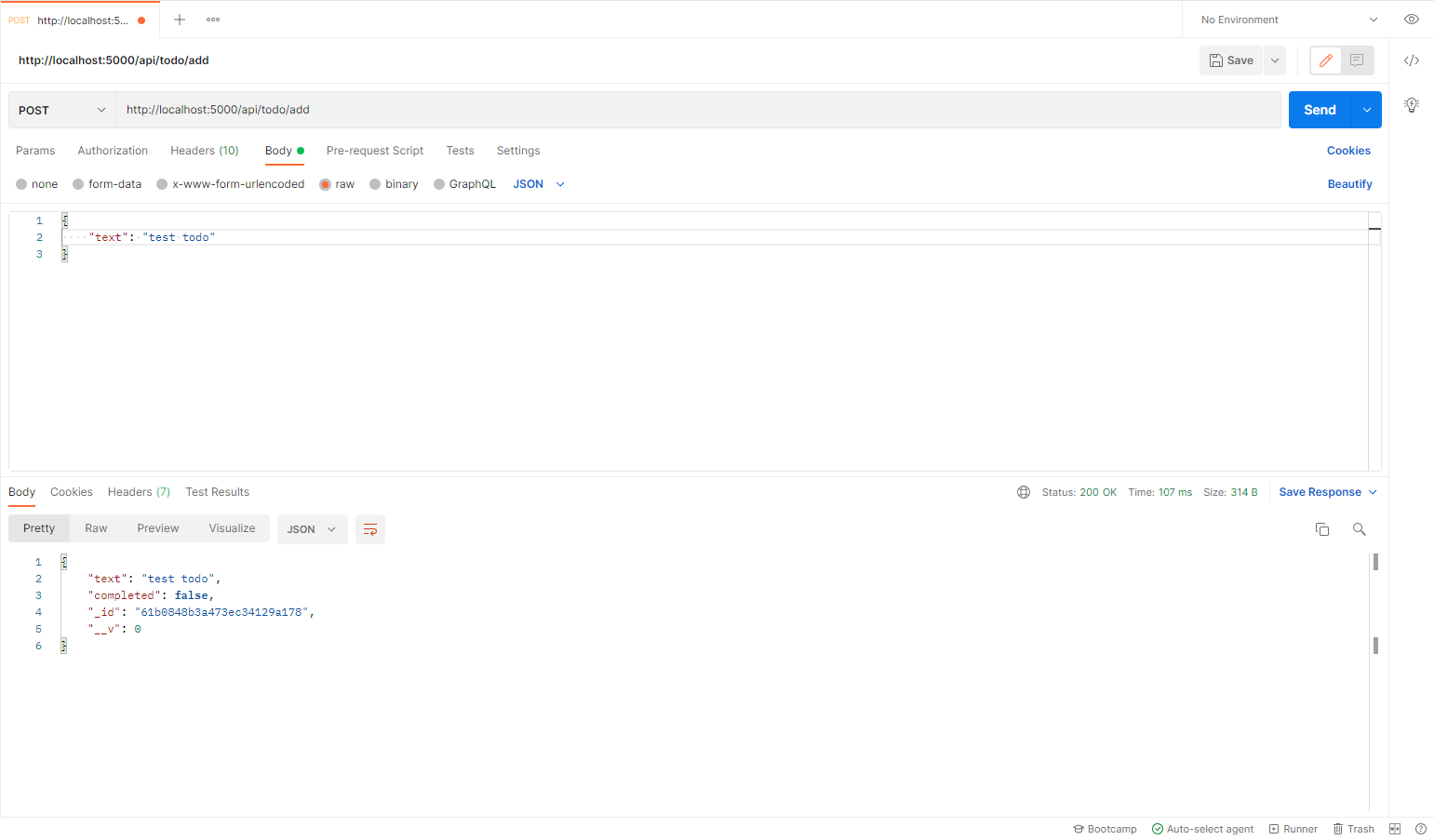
# Приложение Б

# (обязательное)

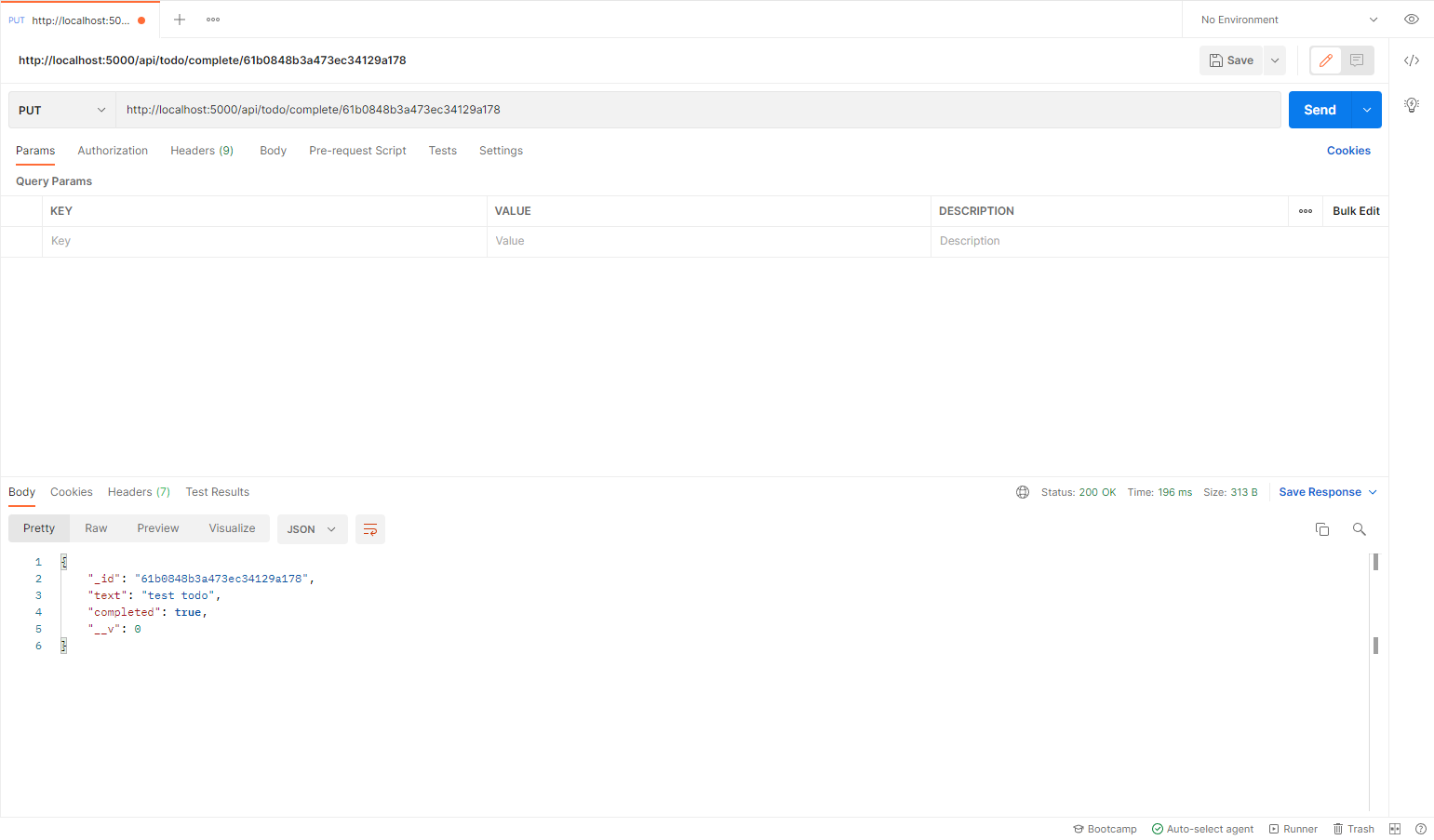
# GET запрос



# POST запрос



# PUT запрос

**DELETE запрос**

