МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**Расчетно-графическое задание**

по дисциплине «Технологии Web-программирования»

Выполнил: ст. группы ВТ-42

Черных Артём Вячеславович

Проверил: Картамышев С.В.

Белгород 2021 г.

**Содержание**

[HTML. Разработка макетов и верстка шаблонов 3](#_Toc89715087)

[Клиентское программирование. Vue js 5](#_Toc89715088)

[Серверное программирование. API 7](#_Toc89715089)

[Проектирование Базы Данных 11](#_Toc89715090)

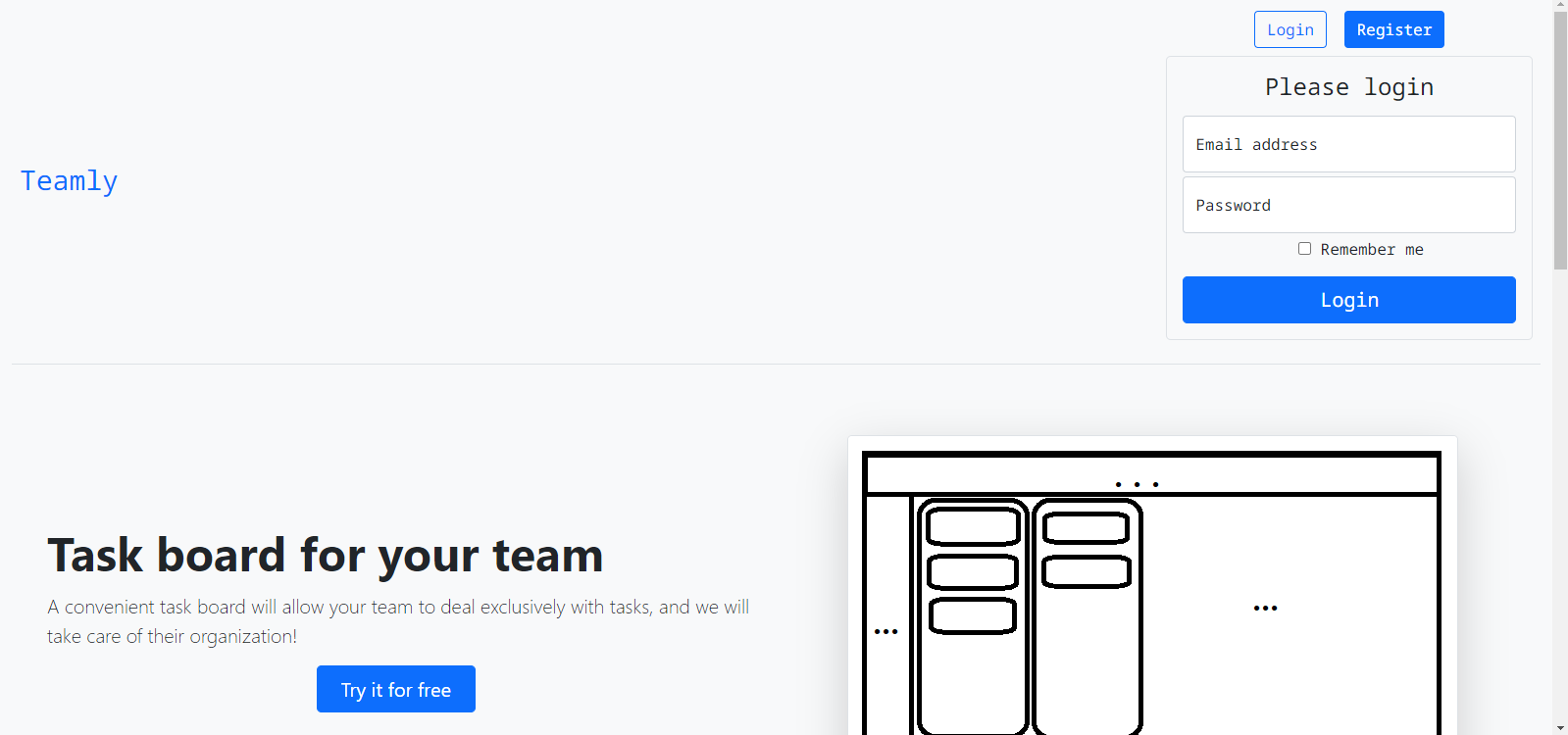
[Серверное программирование. REST API 13](#_Toc89715091)

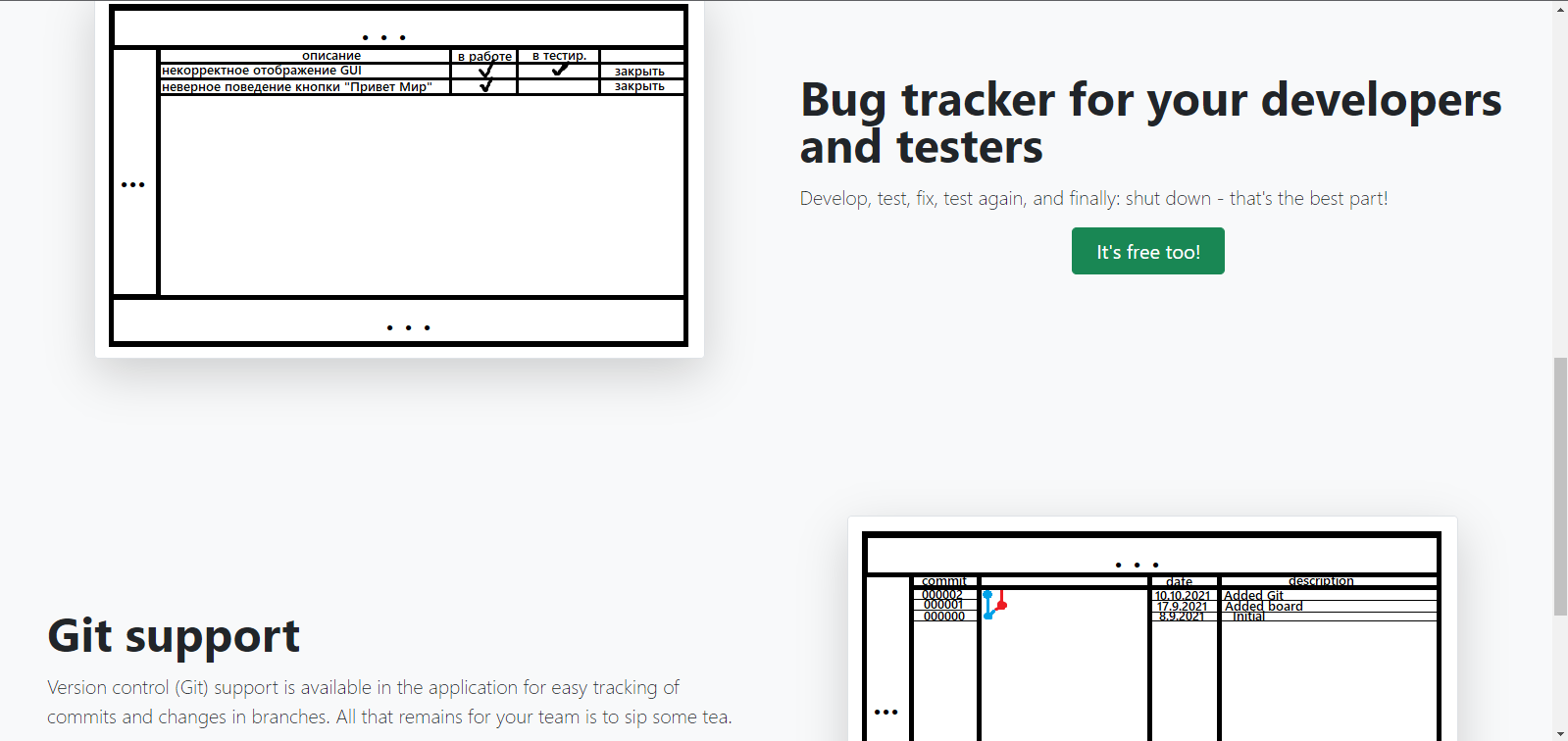
[Взаимодействие frontend с backend 18](#_Toc89715092)

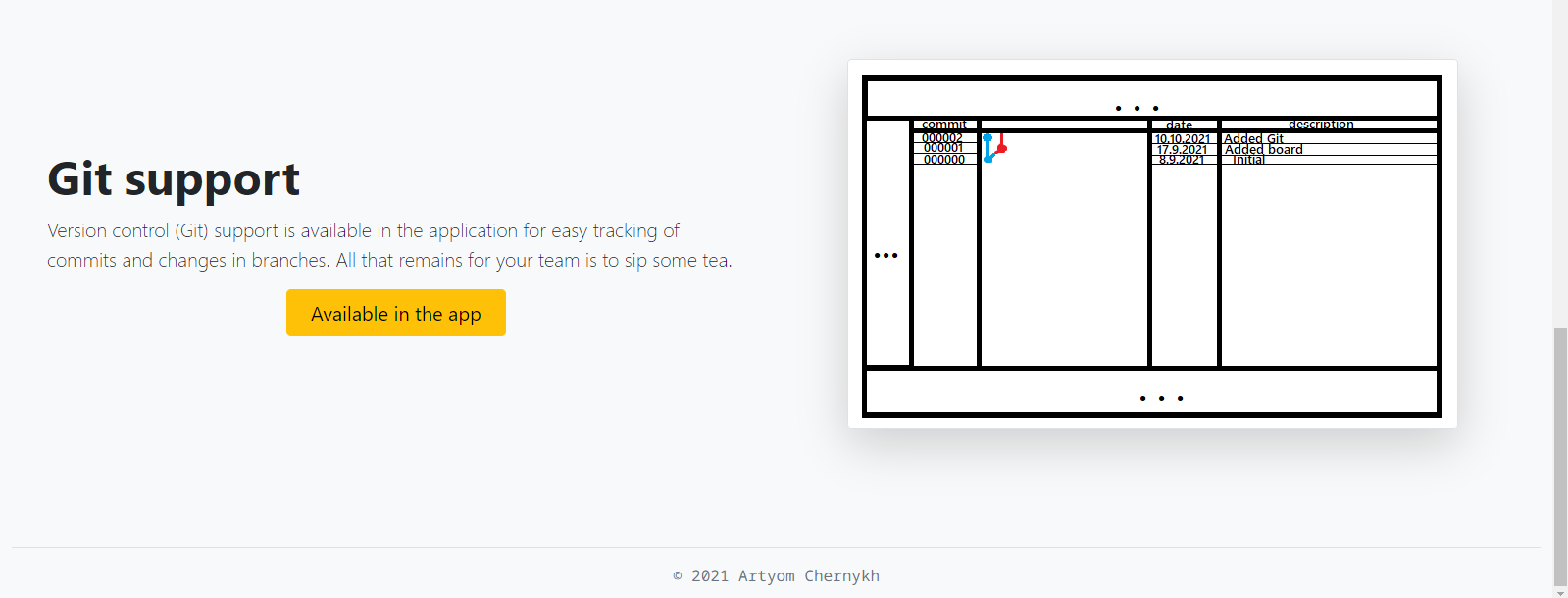
# HTML. Разработка макетов и верстка шаблонов

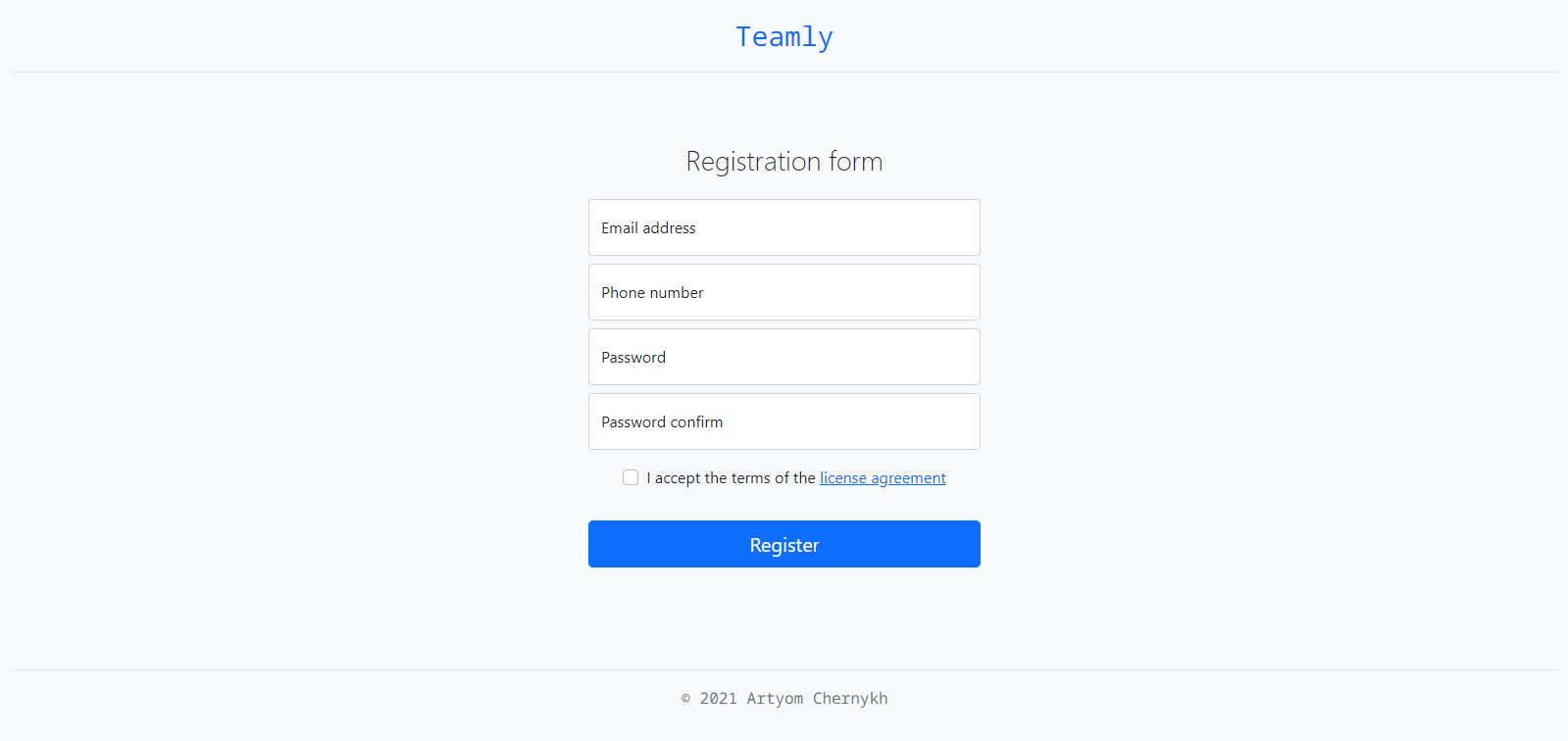
Целью ставилось изучение основ работы протокола передачи гипертекста HTTP. Изучение основ разметки страниц с помощью языка HTML. Изучение основ вёрстки страниц, адаптированных под различные устройства. Изучение основ работы задания стилей для разметки страницы с помощью языка CSS. Разработка макетов и верстка страницы для web-приложения.

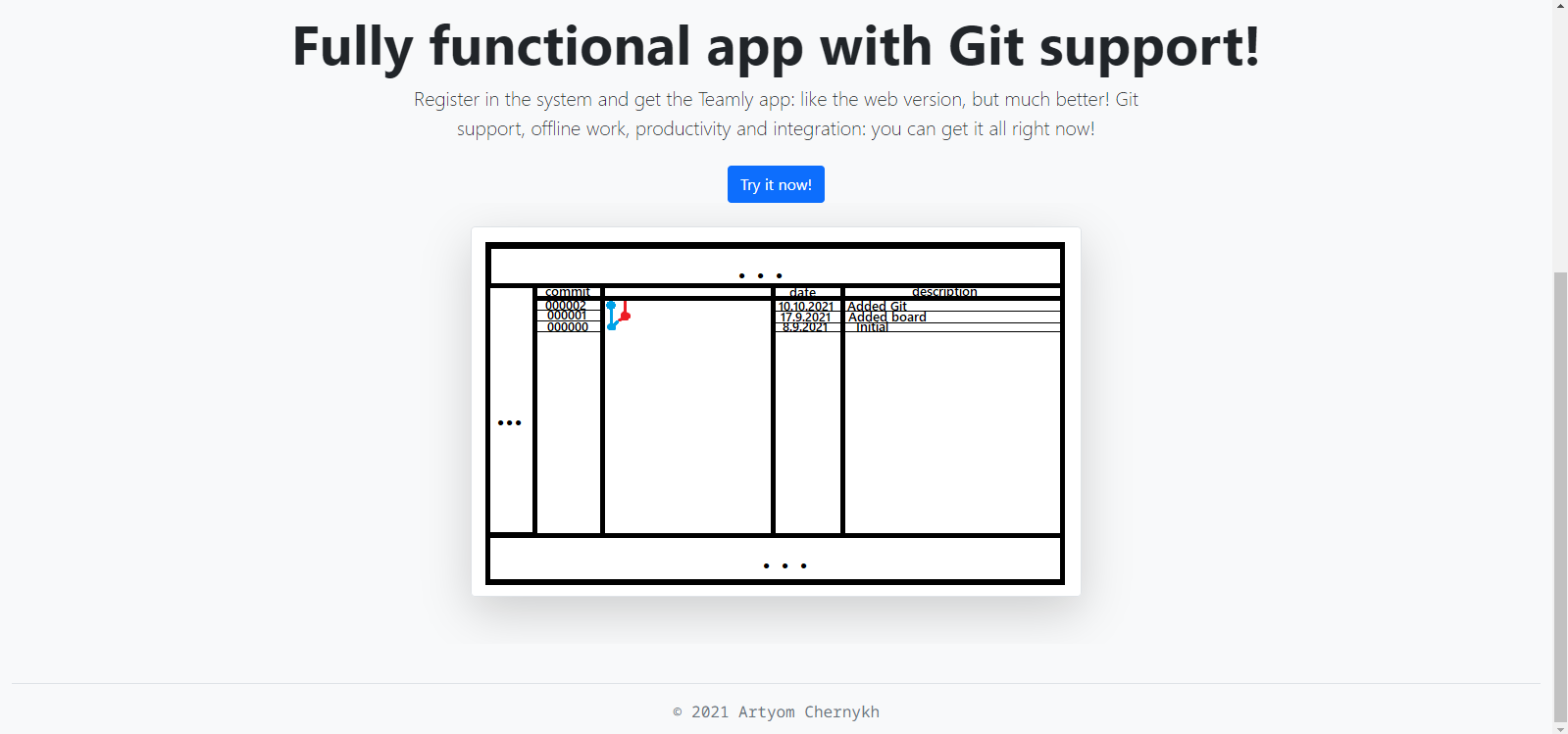
В качестве фреймворка был выбран Bootstrap версии 5.





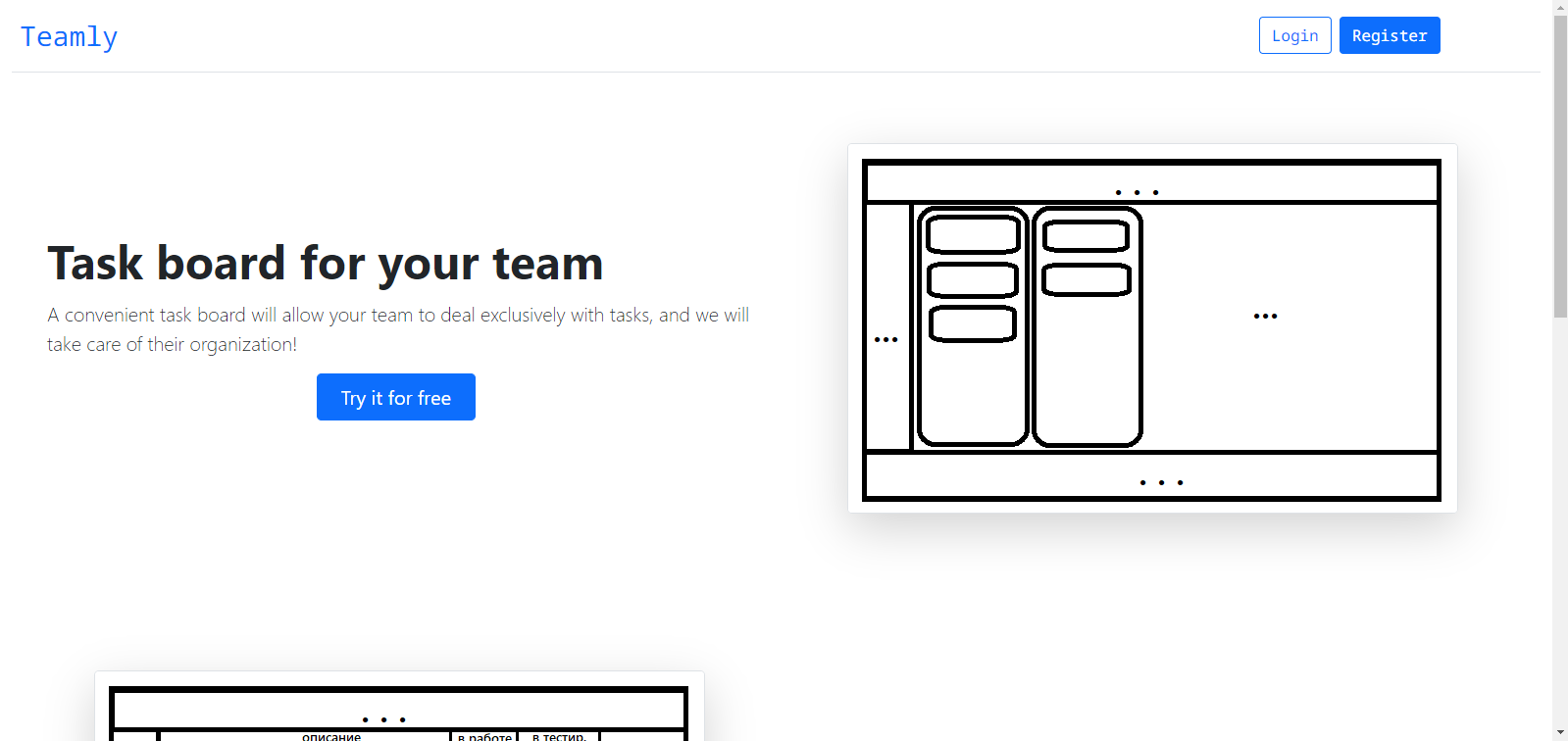


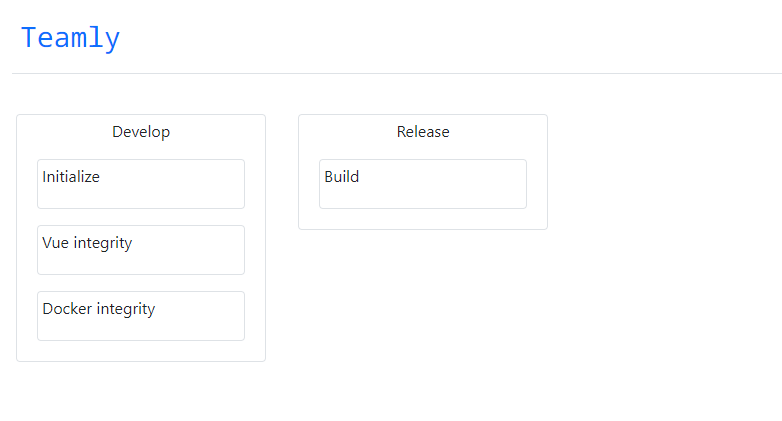


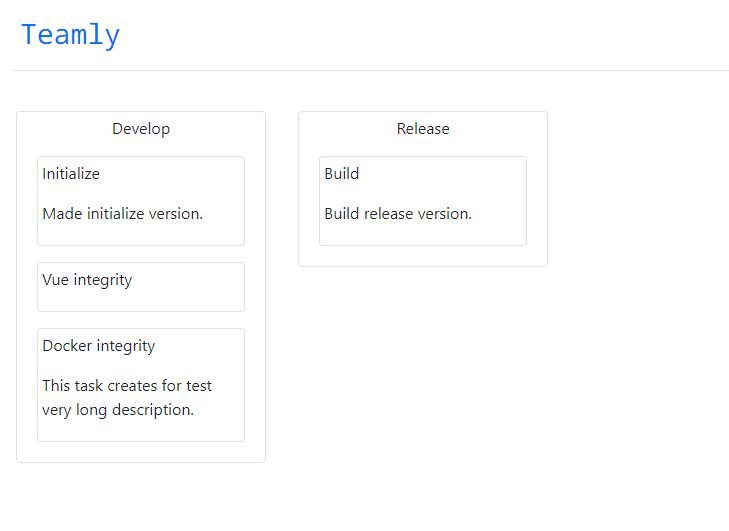


# Клиентское программирование. Vue js

Целью ставилось ознакомление с основами разработки на языке JavaScript. Ознакомление с основами разработки frontend приложения, в частности с использованием клиентских фреймворков (Vue js). Перенос вёрстки в отдельные компоненты.





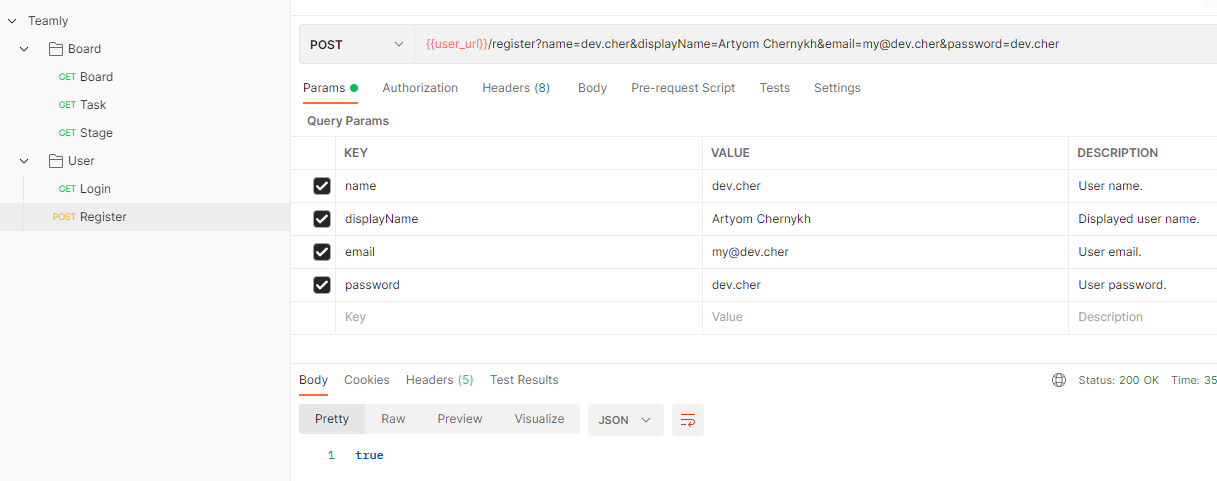


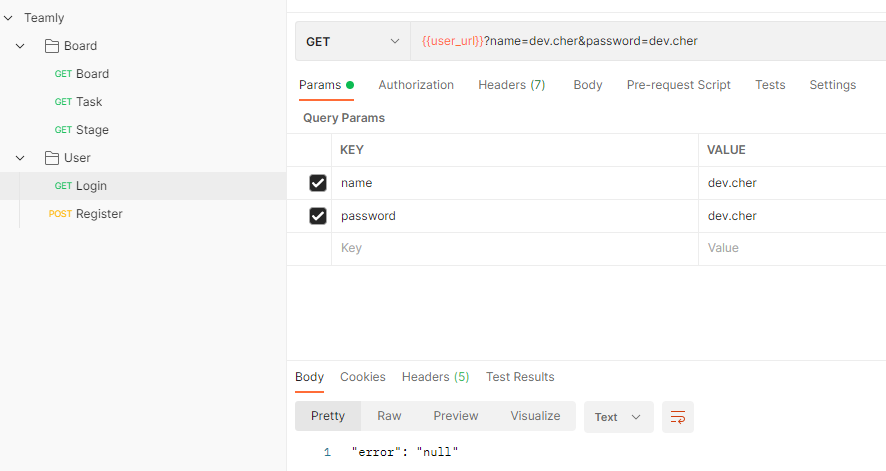
# Серверное программирование. API

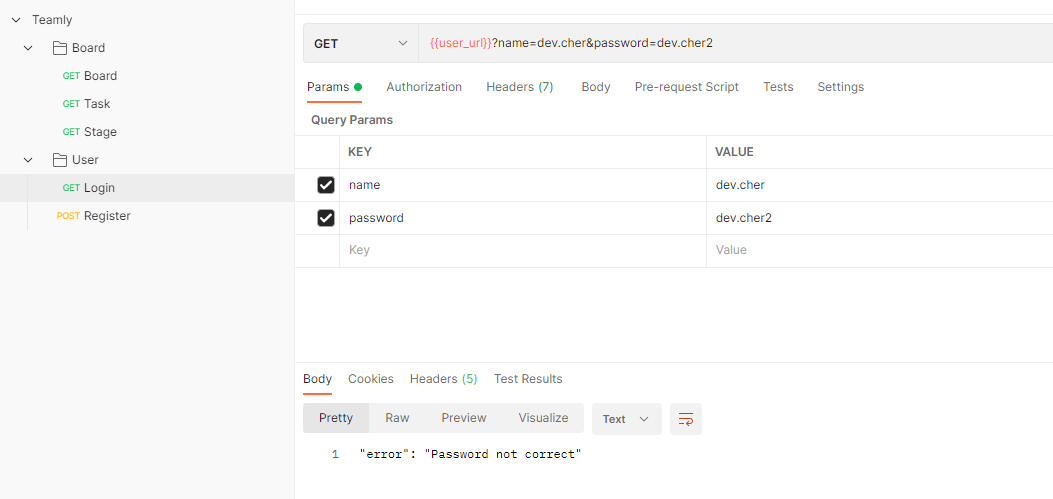
Целью ставилось ознакомление с основами разработки на языке Java. Ознакомление с основами разработки backend приложения, в частности с использованием серверных фреймворков (Spring). Составление API системы.

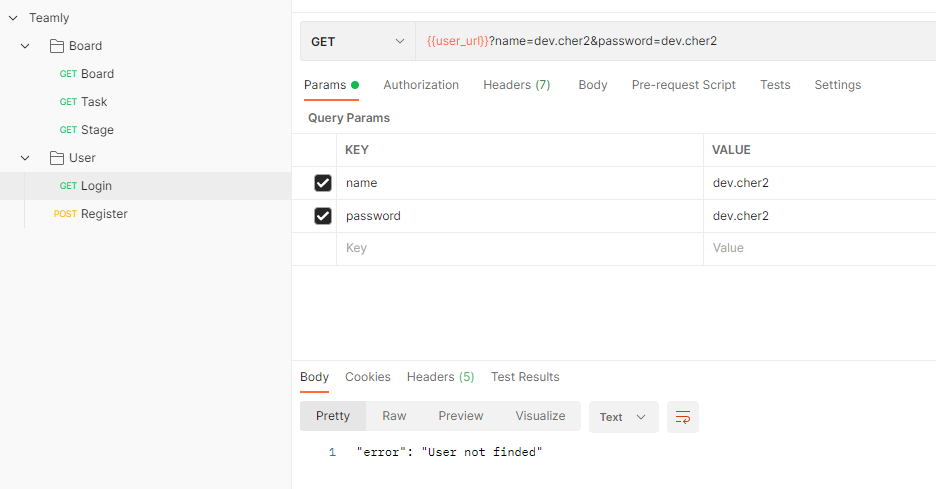
Для написания backend-части системы был выбран язык Java и фреймворк Spring. Также было составлено API для получения данных доски (в целом и по отдельности) и операции с пользователем (войти, регистрация).

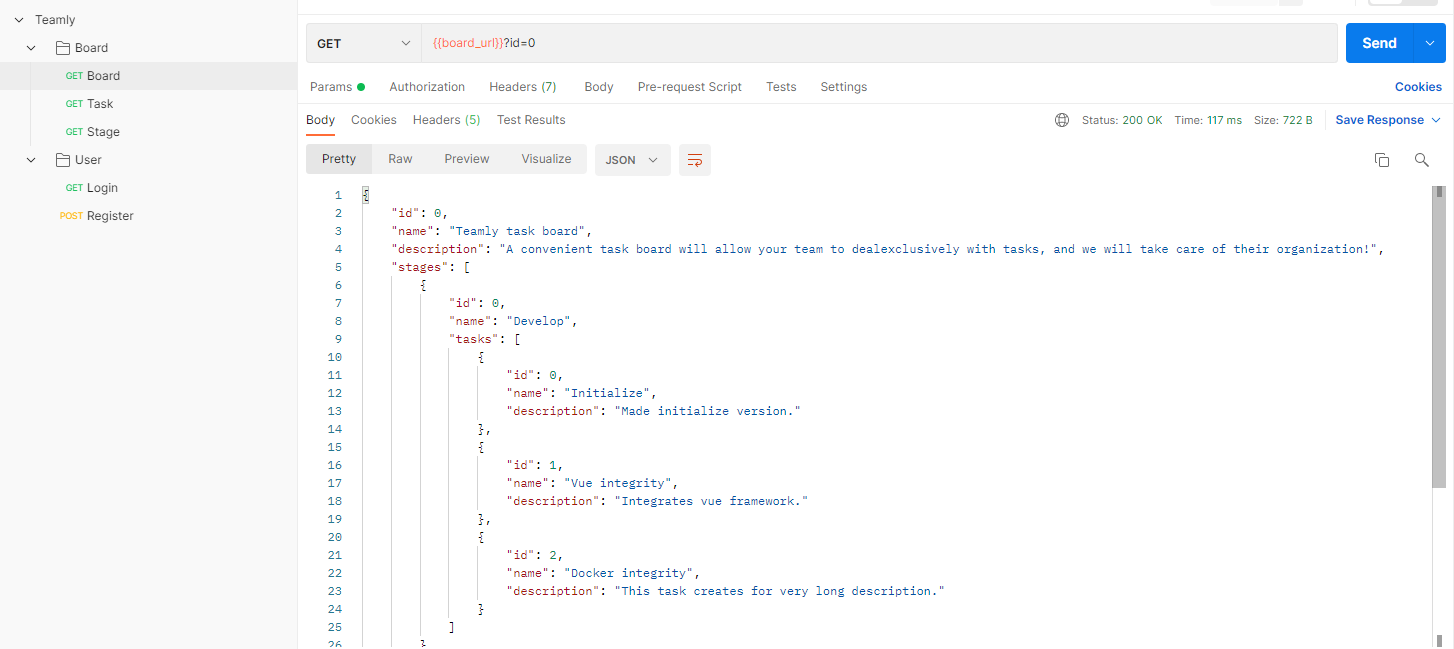
Запросы были протестированы в приложении **Postman**.

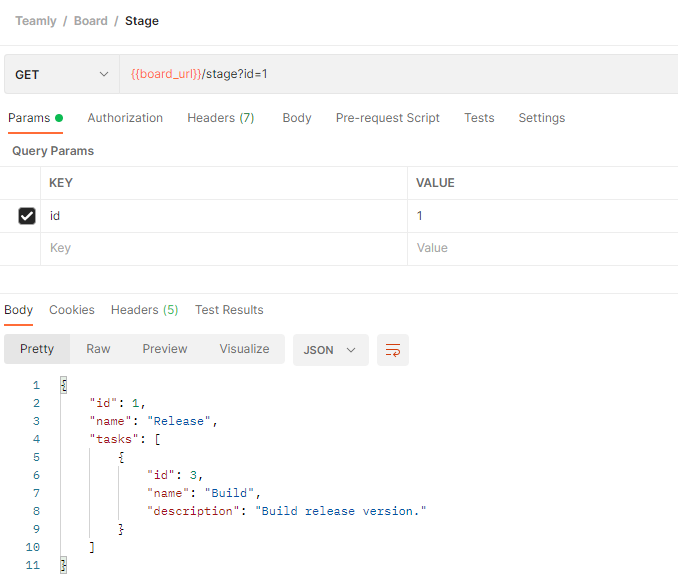


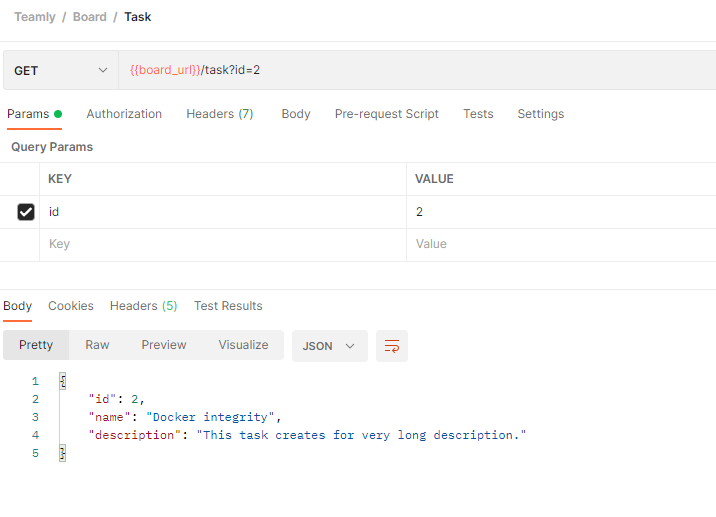


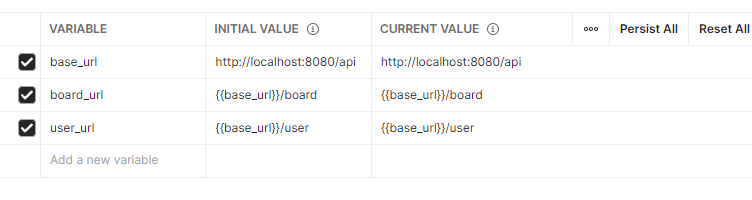


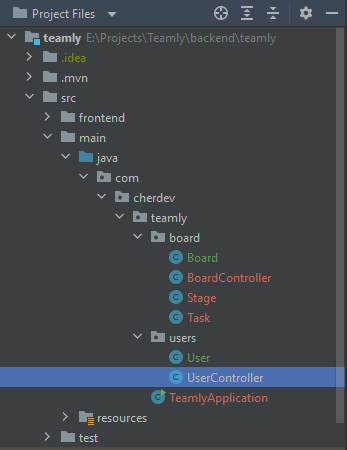












# Проектирование Базы Данных

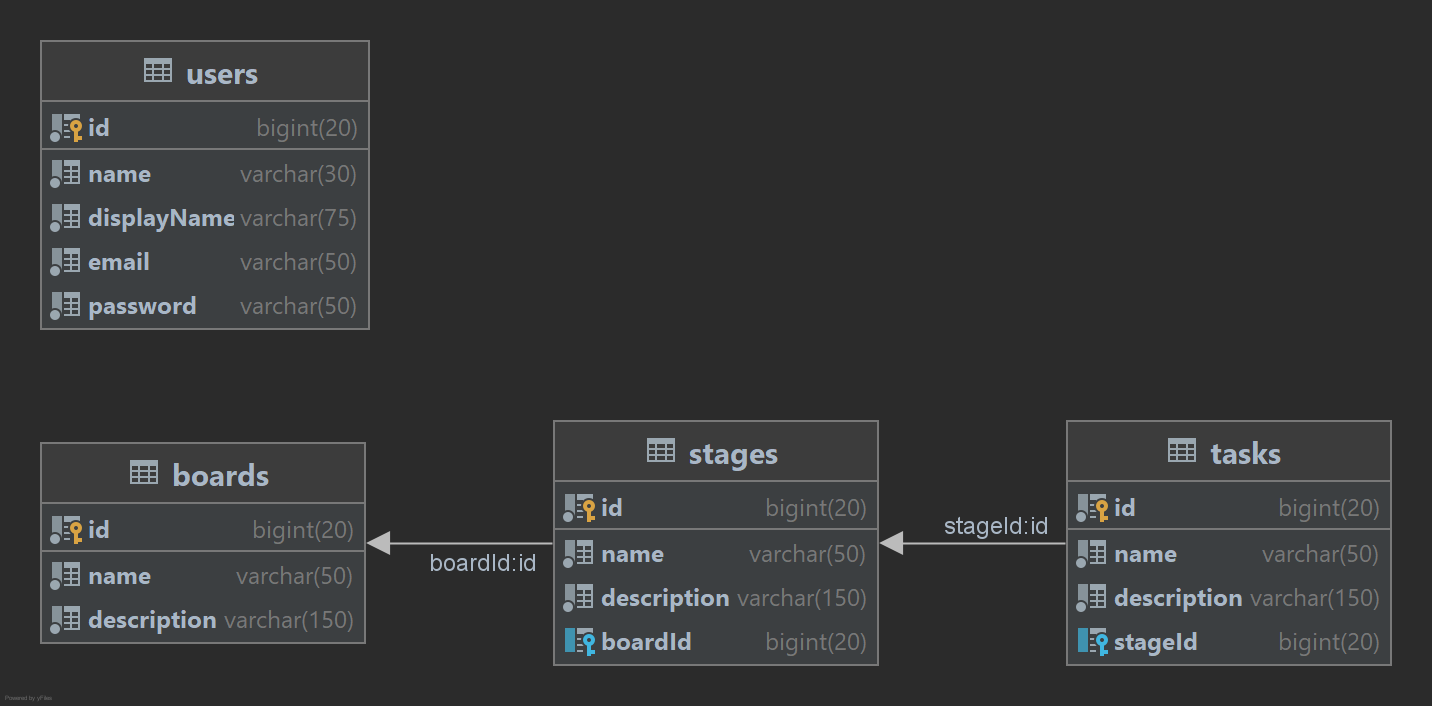
Целью ставилось проектирование Базы Данных (БД) для функционирования системы.

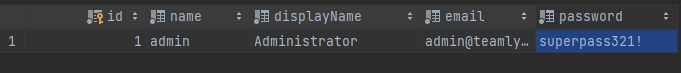
В БД (teamlydb) имеются 4 таблицы:

1. Пользователи (users): имя, отображаемое имя, адрес электронной почты, пароль;
2. Доски (boards): наименование, описание;
3. Стадии (stages): наименование, описание, доска-родитель;
4. Задачи (tasks): наименование, описание, стадия-родитель.

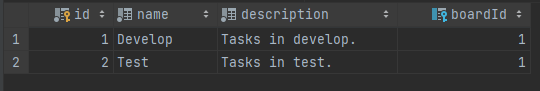
Также в БД были занесены некоторые тестовые данные.

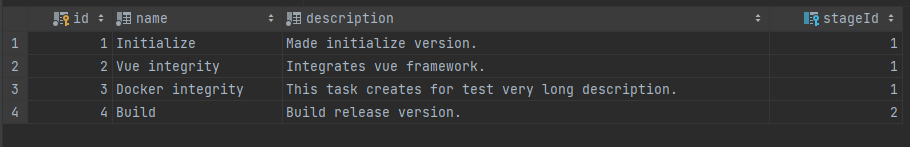
Для создания данной структуры БД был написан SQL-скрипт, исходный код которого доступен в приложении.

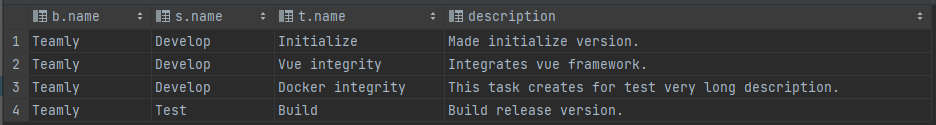
****











SQL:

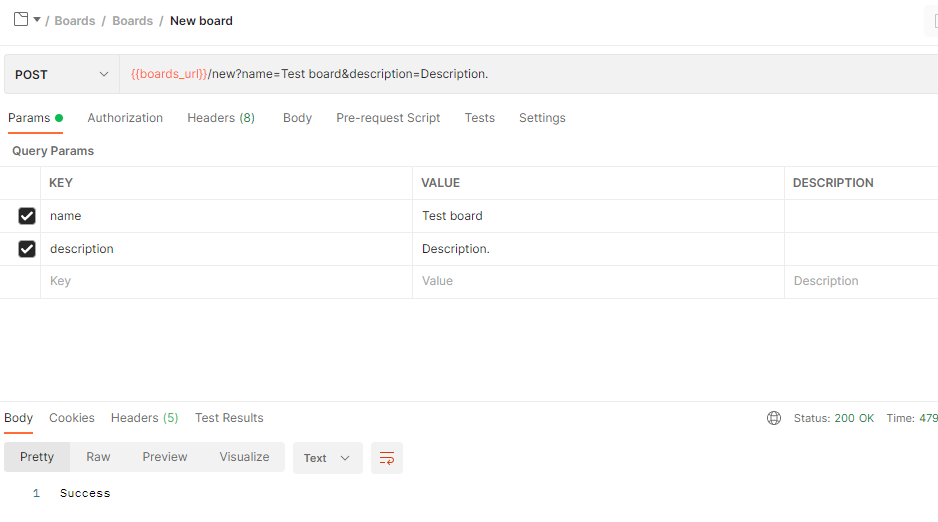
CREATE DATABASE teamlydb;  
  
use teamlydb;  
  
CREATE TABLE users  
(  
 id bigint primary key auto\_increment,  
 name varchar(30) not null,  
 displayName varchar(75) not null,  
 email varchar(50) not null,  
 password varchar(50) not null  
);  
  
DROP TABLE users;  
  
CREATE TABLE boards  
(  
 id bigint primary key auto\_increment,  
 name varchar(50) not null,  
 description varchar(50) not null  
);  
  
ALTER TABLE boards  
MODIFY description varchar(150) not null default '';  
  
CREATE TABLE stages  
(  
 id bigint primary key auto\_increment,  
 name varchar(50) not null,  
 description varchar(150) not null default '',  
 boardId bigint REFERENCES boards(id) ON DELETE CASCADE ON UPDATE CASCADE  
);  
  
CREATE TABLE tasks  
(  
 id bigint primary key auto\_increment,  
 name varchar(50) not null,  
 description varchar(150) not null default '',  
 stageId bigint REFERENCES stages(id) ON DELETE CASCADE ON UPDATE CASCADE  
);  
  
SELECT b.name, s.name, t.name, t.description  
FROM boards b  
JOIN stages s on b.id = s.boardId  
JOIN tasks t on s.id = t.stageId;

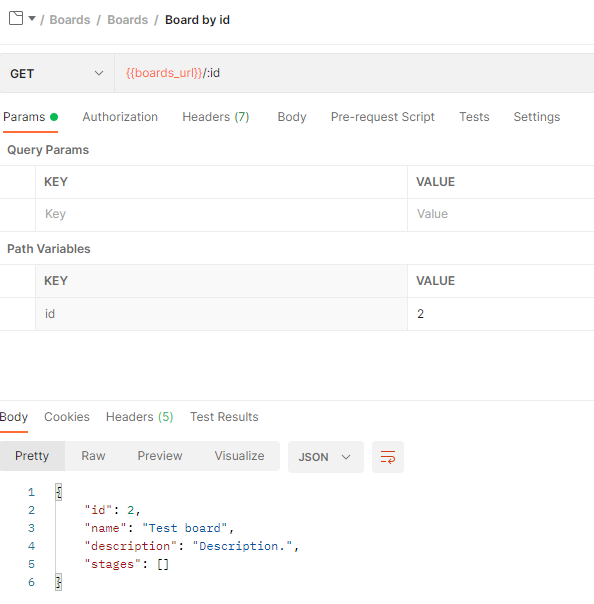
# Серверное программирование. REST API

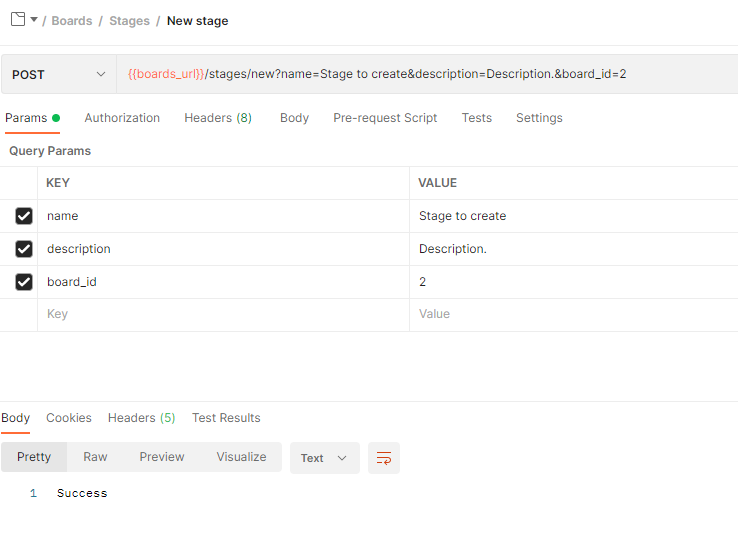
Целью ставилось научиться использовать базу данных на серверной части системы и отправлять ответы, используя REST API. Научиться составлять документацию к разработанному API.

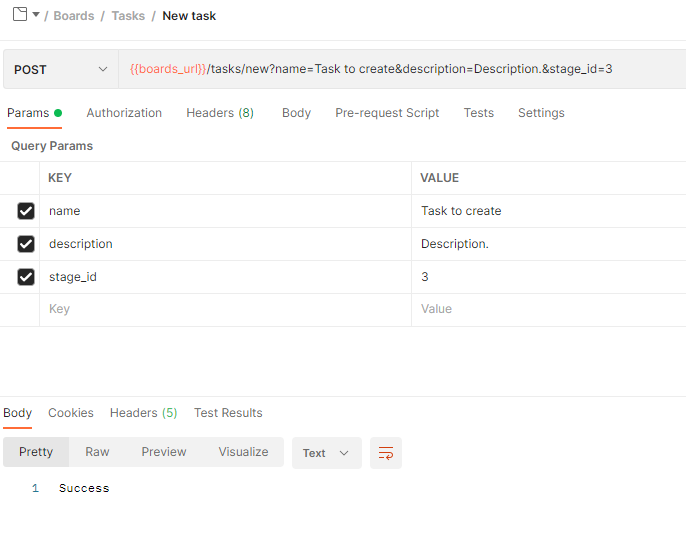
В системе используются следующие виды запросов:

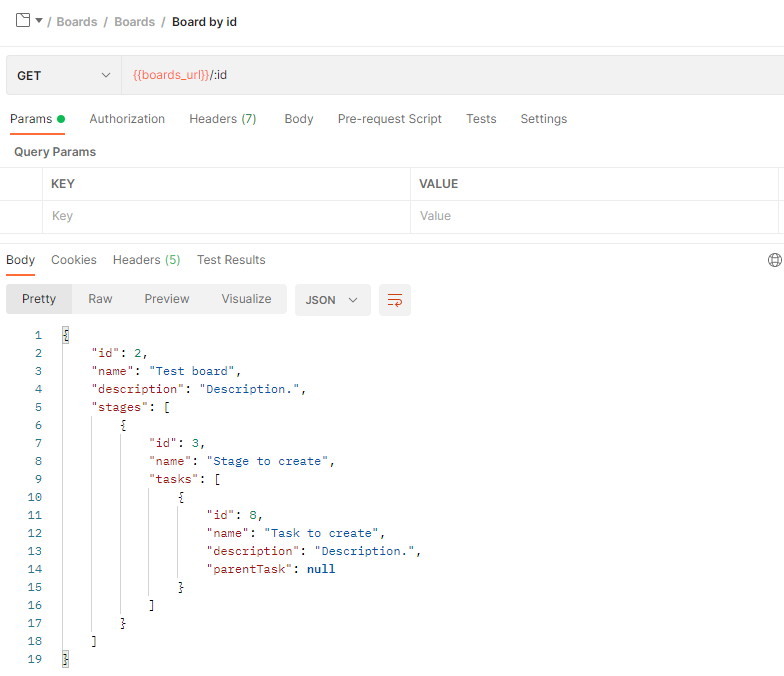
1. Boards. Для взаимодействия с подсистемой «Доски»:
   1. Boards. Для взаимодействия с сущностью «доска»;
   2. Stages. Для взаимодействия с сущностью «стадия»;
   3. Tasks. Для взаимодействия с сущностью «задача».
2. Users. Для взаимодействия с пользователями.

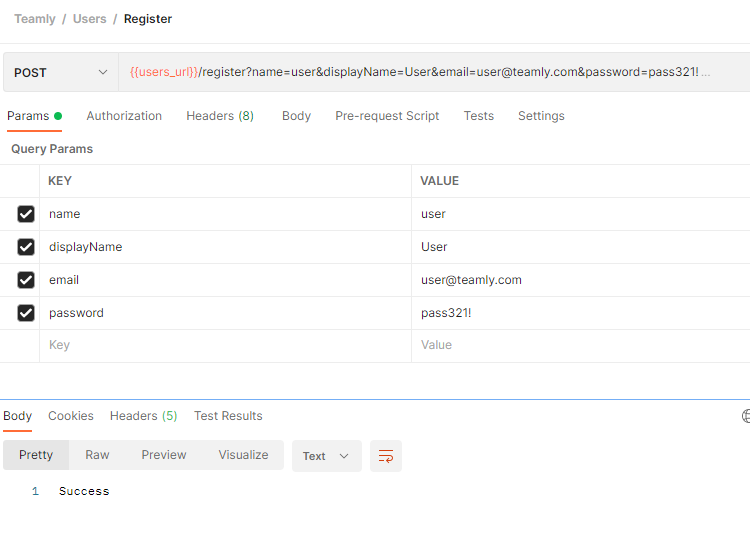


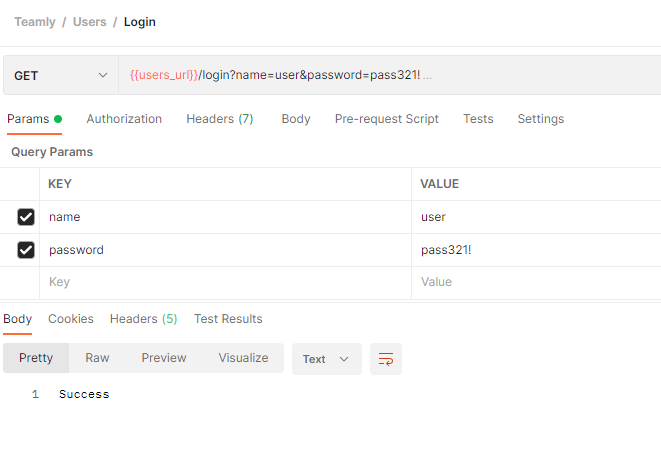


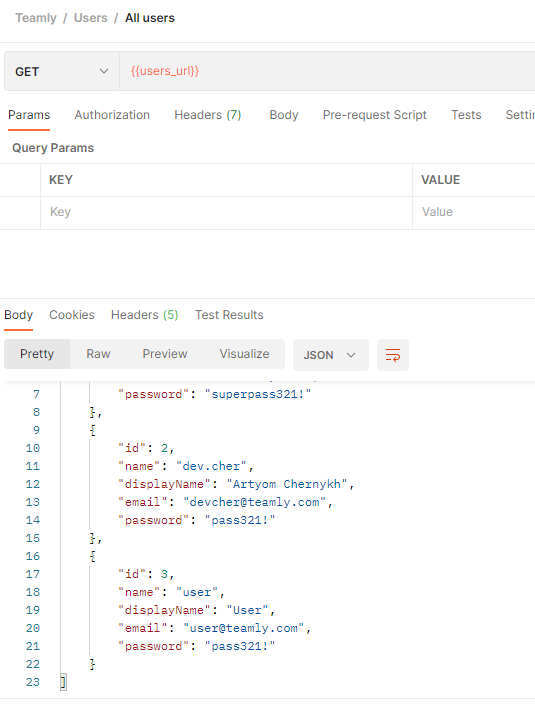


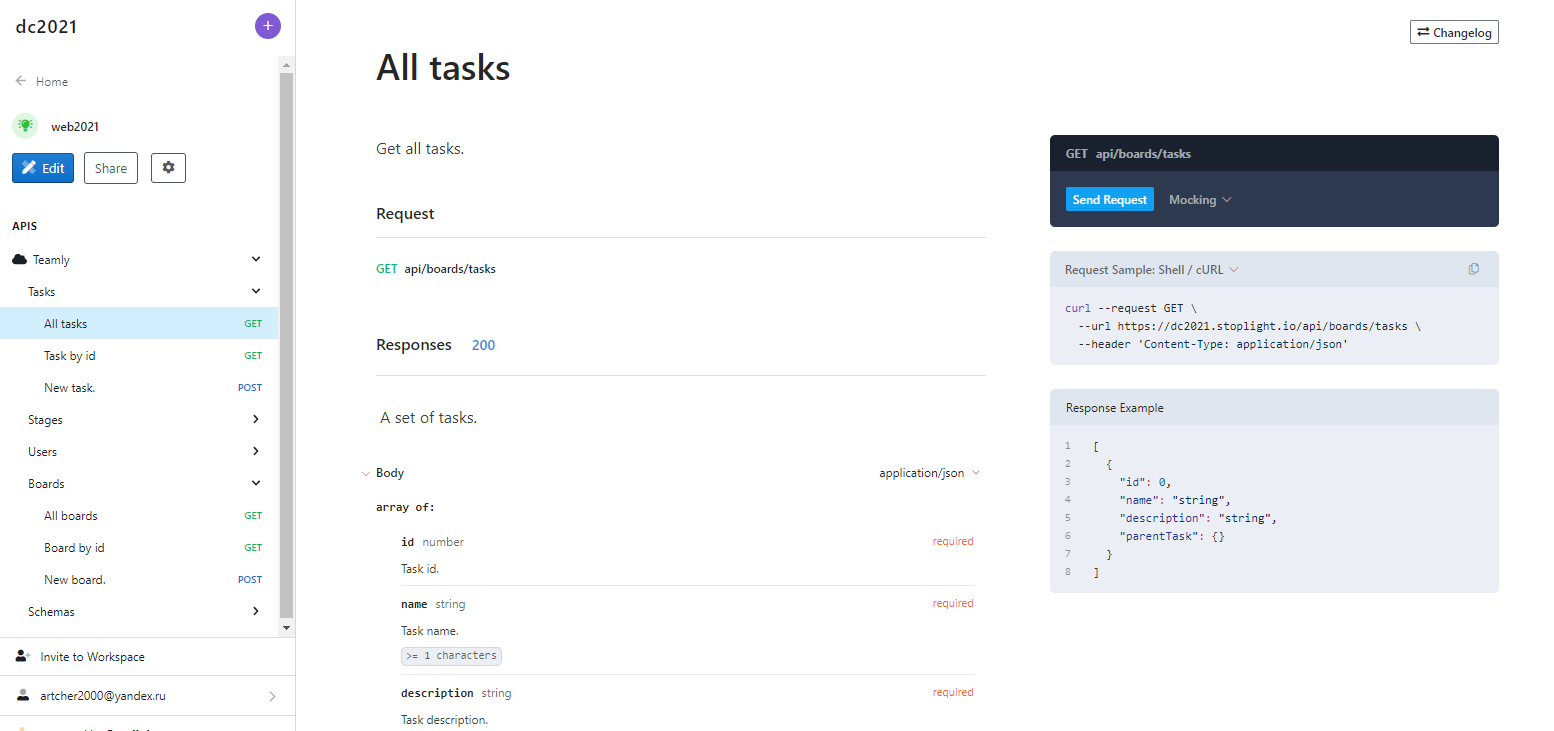












# Взаимодействие frontend с backend

Целью ставилось научиться взаимодействию frontend с backend посредством API.

