

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.

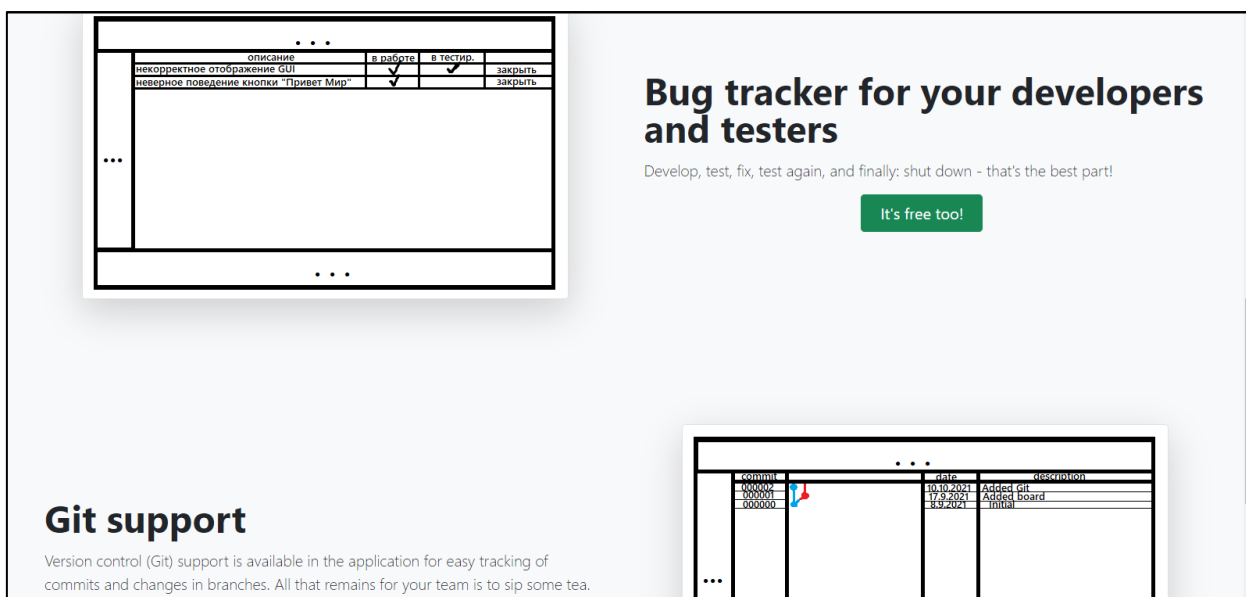
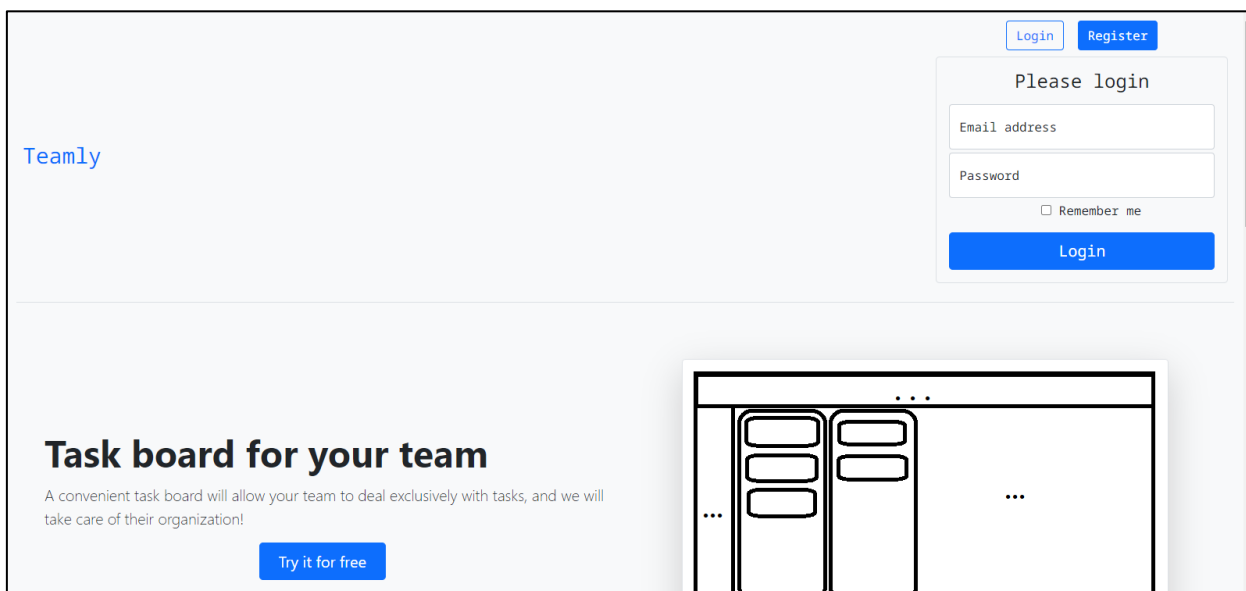
## Содержание

HTML. Разработка макетов и верстка шаблонов .....	3
Клиентское программирование. Vue js .....	5
Серверное программирование. API .....	7
Проектирование Базы Данных .....	11
Серверное программирование. REST API .....	13
Взаимодействие frontend с backend.....	18

## 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.

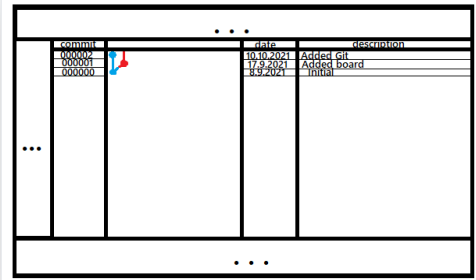
В качестве фреймворка был выбран Bootstrap версии 5.



## Git support

Version control (Git) support is available in the application for easy tracking of commits and changes in branches. All that remains for your team is to sip some tea.

Available in the app



commit	date	description
000000	10.10.2021	Added Git
000001	10.9.2021	Added board
000000	8.9.2021	Initial

© 2021 Artyom Chernykh

## Teamly

### Registration form

Email address

Phone number

Password

Password confirm

☐ I accept the terms of the [license agreement](#)

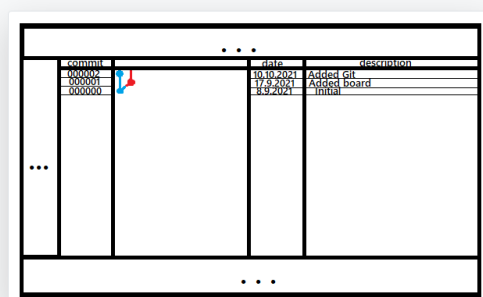
Register

© 2021 Artyom Chernykh

## Fully functional app with Git support!

Register in the system and get the Teamly app: like the web version, but much better! Git support, offline work, productivity and integration: you can get it all right now!

Try it now!

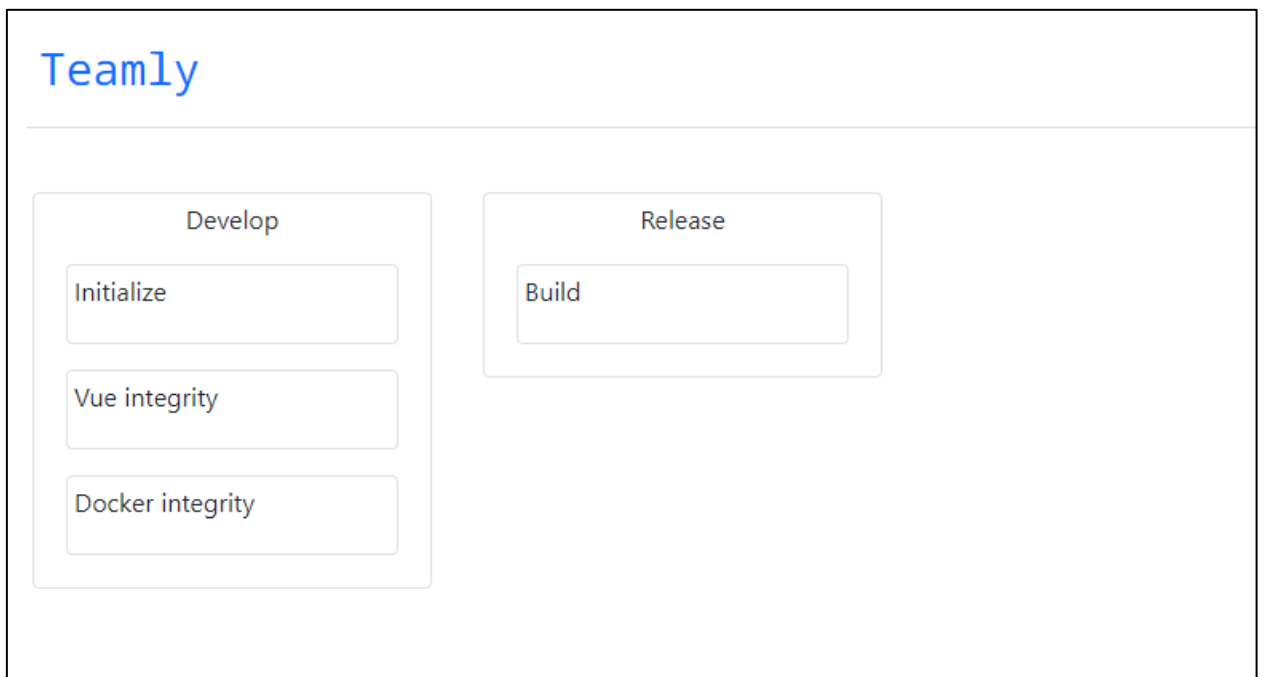
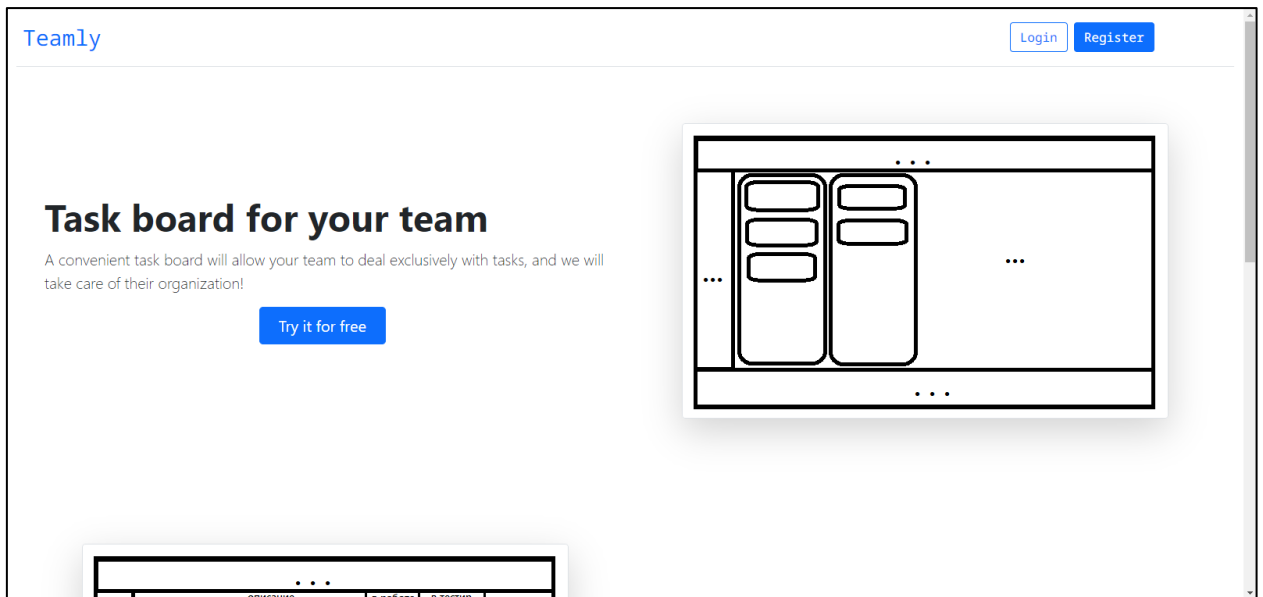


commit	date	description
000000	10.10.2021	Added Git
000001	10.9.2021	Added board
000000	8.9.2021	Initial

© 2021 Artyom Chernykh

## 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.



# Teamly

## Develop

### Initialize

Made initialize version.

### Vue integrity

### Docker integrity

This task creates for test  
very long description.

## Release

### Build

Build release version.

# 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**.

The screenshot shows the Postman interface with a POST request to `{{user_url}}/register?name=dev.cher&displayName=Artyom Chernykh&email=my@dev.cher&password=dev.cher`. The 'Params' tab is active, displaying a table of query parameters.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> name	dev.cher	User name.
<input checked="" type="checkbox"/> displayName	Artyom Chernykh	Displayed user name.
<input checked="" type="checkbox"/> email	my@dev.cher	User email.
<input checked="" type="checkbox"/> password	dev.cher	User password.
Key	Value	Description

The 'Body' tab is selected, showing a status of 200 OK and a response of `1 true`.

The screenshot shows the Postman interface with a GET request to `{{user_url}}?name=dev.cher&password=dev.cher`. The 'Params' tab is active, displaying a table of query parameters.

KEY	VALUE
<input checked="" type="checkbox"/> name	dev.cher
<input checked="" type="checkbox"/> password	dev.cher
Key	Value

The 'Body' tab is selected, showing a status of 200 OK and a response of `1 {"error": "null"}`.

Teamly

Board

GET Board

GET Task

GET Stage

User

GET Login

POST Register

GET

{{user\_url}}?name=dev.cher&password=dev.cher2

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	name	dev.cher
<input checked="" type="checkbox"/>	password	dev.cher2
	Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

```

1  "error": "Password not correct"

```

Teamly

Board

GET Board

GET Task

GET Stage

User

GET Login

POST Register

GET

{{user\_url}}?name=dev.cher2&password=dev.cher2

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	name	dev.cher2
<input checked="" type="checkbox"/>	password	dev.cher2
	Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

```

1  "error": "User not finded"

```

Teamly

Board

GET Board

GET Task

GET Stage

User

GET Login

POST Register

GET

{{board\_url}}?id=0

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Body

Cookies

Headers (5)

Test Results

Status: 200 OK Time: 117 ms Size: 722 B Save Response

Pretty

Raw

Preview

Visualize

JSON

```

1  {
2    "id": 0,
3    "name": "Teamly task board",
4    "description": "A convenient task board will allow your team to dealexclusively with tasks, and we will take care of their organization!",
5    "stages": [
6      {
7        "id": 0,
8        "name": "Develop",
9        "tasks": [
10         {
11           "id": 0,
12           "name": "Initialize",
13           "description": "Made initialize version."
14         },
15         {
16           "id": 1,
17           "name": "Vue integrity",
18           "description": "Integrates vue framework."
19         },
20         {
21           "id": 2,
22           "name": "Docker integrity",
23           "description": "This task creates for very long description."
24         }
25       ]
26     }
27   }

```



Teamly / Board / Stage

GET{{board\_url}}/stage?id=1

ParamsAuthorizationHeaders (7)BodyPre-request ScriptTestsSettings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	id	1
	Key	Value

BodyCookiesHeaders (5)Test Results

PrettyRawPreviewVisualizeJSON

```
1  {
2    "id": 1,
3    "name": "Release",
4    "tasks": [
5      {
6        "id": 3,
7        "name": "Build",
8        "description": "Build release version."
9      }
10   ]
11 }
```

Teamly / Board / Task

GET{{board\_url}}/task?id=2

ParamsAuthorizationHeaders (7)BodyPre-request ScriptTestsSettings

Query Params

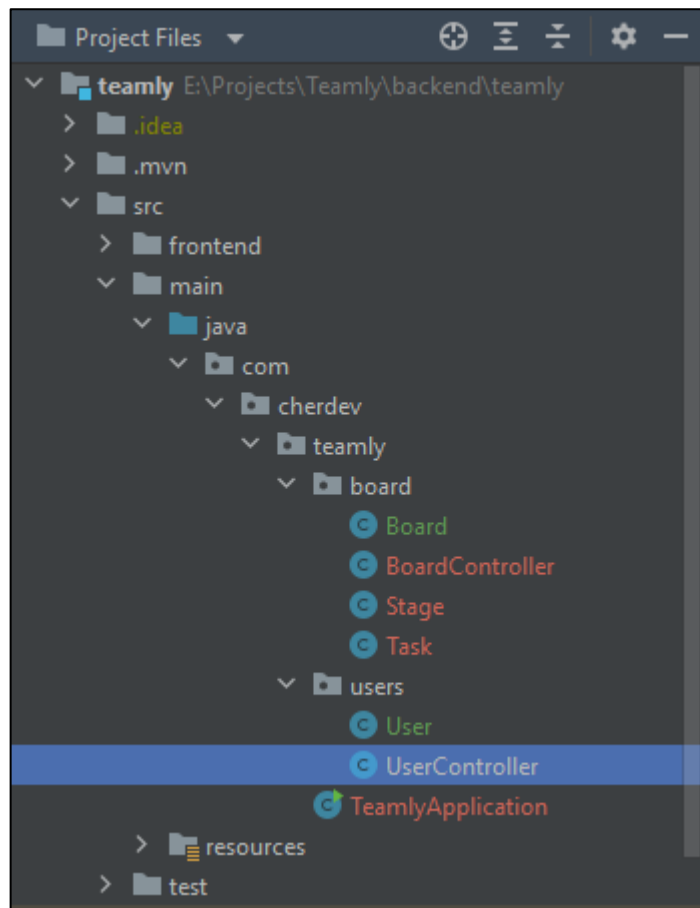
	KEY	VALUE
<input checked="" type="checkbox"/>	id	2
	Key	Value

BodyCookiesHeaders (5)Test Results

PrettyRawPreviewVisualizeJSON

```
1  {
2    "id": 2,
3    "name": "Docker integrity",
4    "description": "This task creates for very long description."
5  }
```

	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ	⋮	Persist All	Reset All
<input checked="" type="checkbox"/>	base_url	http://localhost:8080/api	http://localhost:8080/api			
<input checked="" type="checkbox"/>	board_url	{{base_url}}/board	{{base_url}}/board			
<input checked="" type="checkbox"/>	user_url	{{base_url}}/user	{{base_url}}/user			
	Add a new variable					



# Проектирование Базы Данных

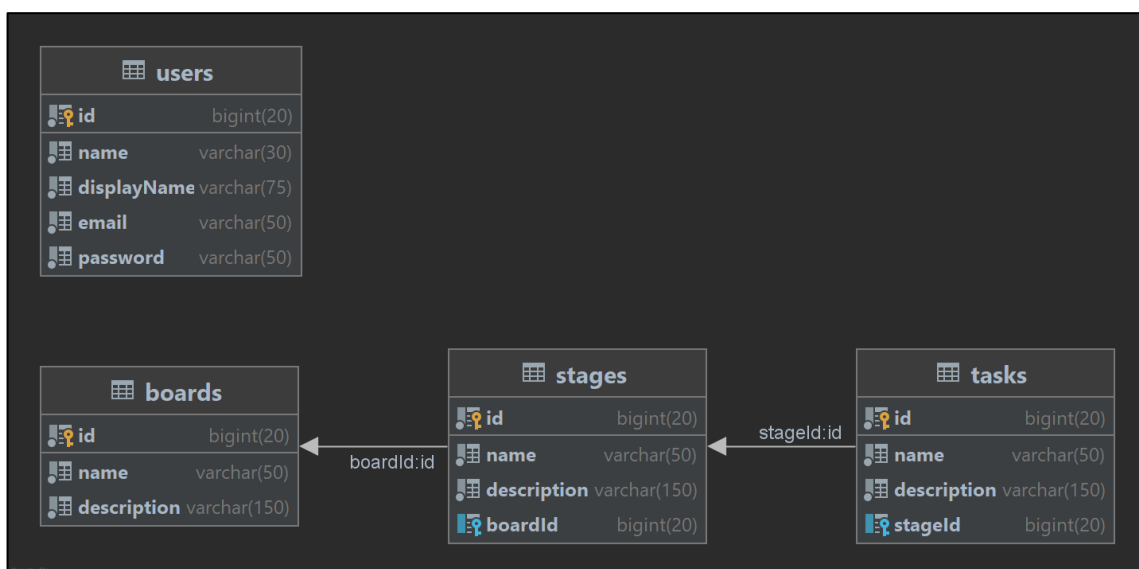
Целью ставилось проектирование Базы Данных (БД) для функционирования системы.

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.



	id	name	displayName	email	password
1	1	admin	Administrator	admin@teamly...	superpass321!

	id	name	description
1	1	Teamly	Board for service test.

	id	name	description	boardId
1	1	Develop	Tasks in develop.	1
2	2	Test	Tasks in test.	1

	id	name	description	stageId
1	1	Initialize	Made initialize version.	1
2	2	Vue integrity	Integrates vue framework.	1
3	3	Docker integrity	This task creates for test very long description.	1
4	4	Build	Build release version.	2

	b.name	s.name	t.name	description
1	Teamly	Develop	Initialize	Made initialize version.
2	Teamly	Develop	Vue integrity	Integrates vue framework.
3	Teamly	Develop	Docker integrity	This task creates for test very long description.
4	Teamly	Test	Build	Build release version.

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
);

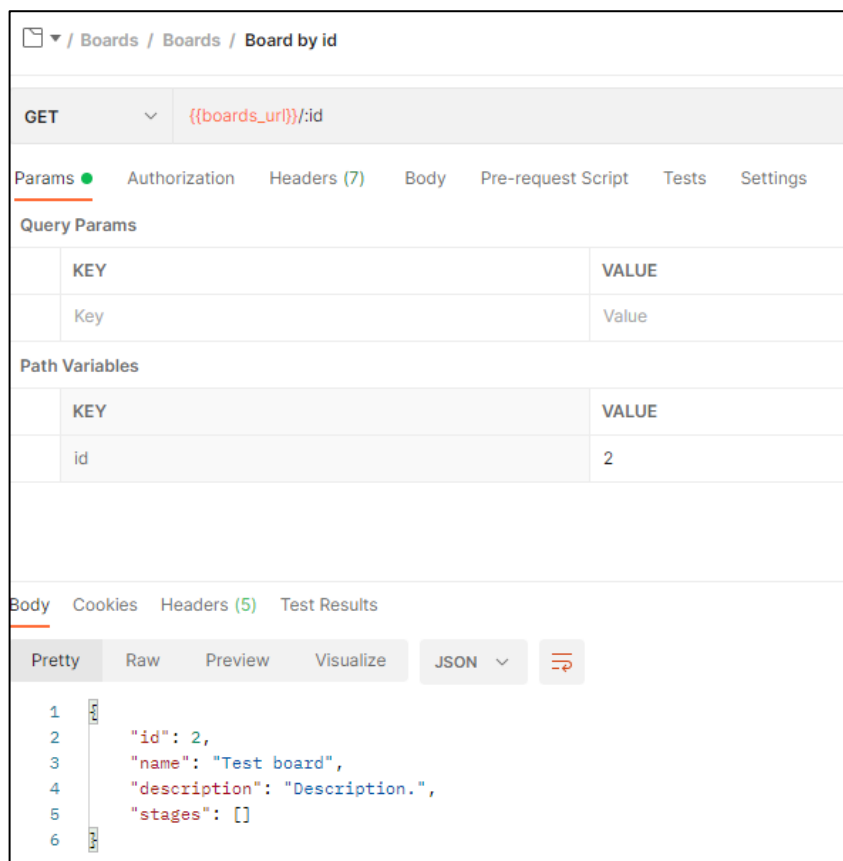
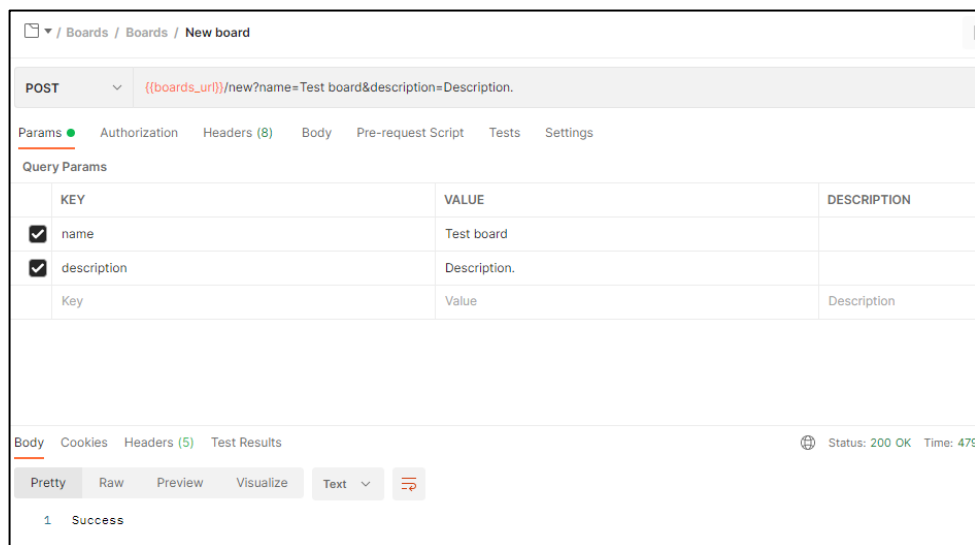
SELECT b.name, s.name, t.name, t.description
FROM boards b
JOIN stages s on b.id = s.boardId
JOIN tasks t on s.id = t.stageId;
```

# 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»:
  - 1.1. Boards. Для взаимодействия с сущностью «доска»;
  - 1.2. Stages. Для взаимодействия с сущностью «стадия»;
  - 1.3. Tasks. Для взаимодействия с сущностью «задача».
2. Users. Для взаимодействия с пользователями.



Boards / Stages / New stage

POST

▼

{{boards\_url}}/stages/new?name=Stage to create&description=Description.&board\_id=2

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	name	Stage to create
<input checked="" type="checkbox"/>	description	Description.
<input checked="" type="checkbox"/>	board_id	2
	Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

▼

1

Success

Boards / Tasks / New task

POST

▼

{{boards\_url}}/tasks/new?name=Task to create&description=Description.&stage\_id=3

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	name	Task to create
<input checked="" type="checkbox"/>	description	Description.
<input checked="" type="checkbox"/>	stage_id	3
	Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

▼

1

Success

📄 / Boards / Boards / Board by id

GET ▼ `{{boards_url}}/id`

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE
Key	Value

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON ▼ ≡

```

1  {
2    "id": 2,
3    "name": "Test board",
4    "description": "Description.",
5    "stages": [
6      {
7        "id": 3,
8        "name": "Stage to create",
9        "tasks": [
10       {
11         "id": 8,
12         "name": "Task to create",
13         "description": "Description.",
14         "parentTask": null
15       }
16     ]
17   }
18 ]
19 }
```

Teamly / Users / Register

POST ▼ `{{users_url}}/register?name=user&displayName=User&email=user@teamly.com&password=pass321! ...`

Params ● Authorization Headers (8) Body Pre-request Script Tests Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	name	user
<input checked="" type="checkbox"/>	displayName	User
<input checked="" type="checkbox"/>	email	user@teamly.com
<input checked="" type="checkbox"/>	password	pass321!
	Key	Value

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize Text ▼ ≡

```

1  Success
```

Teamly / Users / Login

GET

▼

{{users\_url}}/login?name=user&password=pass321! ...

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	name	user
<input checked="" type="checkbox"/>	password	pass321!
	Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

▼

1

Success



Teamly / Users / All users

GET ▼ `{{users_url}}`

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE
Key	Value

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON ▼ ≡

```

7      "password": "superpass321!"
8    },
9    {
10     "id": 2,
11     "name": "dev.cher",
12     "displayName": "Artyom Chernykh",
13     "email": "devcher@teamly.com",
14     "password": "pass321!"
15   },
16   {
17     "id": 3,
18     "name": "user",
19     "displayName": "User",
20     "email": "user@teamly.com",
21     "password": "pass321!"
22   }
23 ]

```

dc2021

Home
web2021

Edit
Share

APIs
Teamly
Tasks
All tasks GET
Task by id GET
New task. POST
Stages
Users
Boards
All boards GET
Board by id GET
New board. POST
Schemas
Invite to Workspace
artcher2000@yandex.ru

## All tasks

Get all tasks.

Request

GET api/boards/tasks

Responses 200

A set of tasks.

Body application/json

array of:

id	number	required
Task id.		
name	string	required
Task name.		
	>= 1 characters	
description	string	required
Task description.		

GET api/boards/tasks
Send Request
Mocking

Request Sample: Shell / cURL

```

curl --request GET \
--url https://dc2021.stoplight.io/api/boards/tasks \
--header 'Content-Type: application/json'

```

Response Example

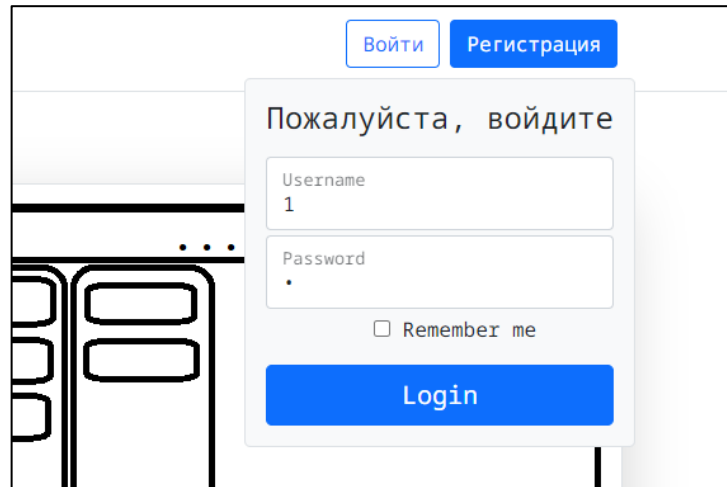
```

1  [
2  {
3    "id": 0,
4    "name": "string",
5    "description": "string",
6    "parentTask": {}
7  }
8  ]

```

## Взаимодействие frontend с backend

Целью ставилось научиться взаимодействию frontend с backend посредством API.



Войти Регистрация

Пожалуйста, войдите

Username  
1

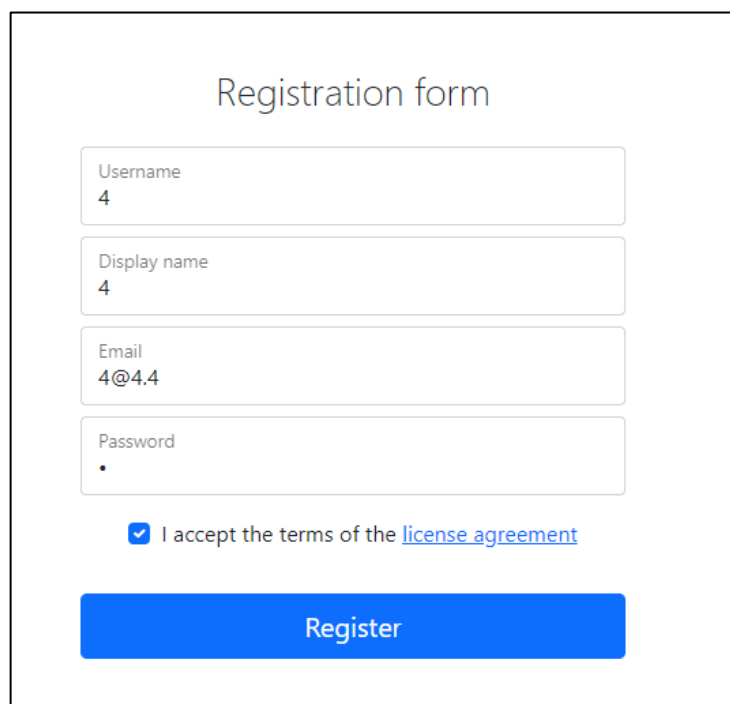
Password  
•

☐ Remember me

Login



Выйти



Registration form

Username  
4

Display name  
4

Email  
4@4.4

Password  
•

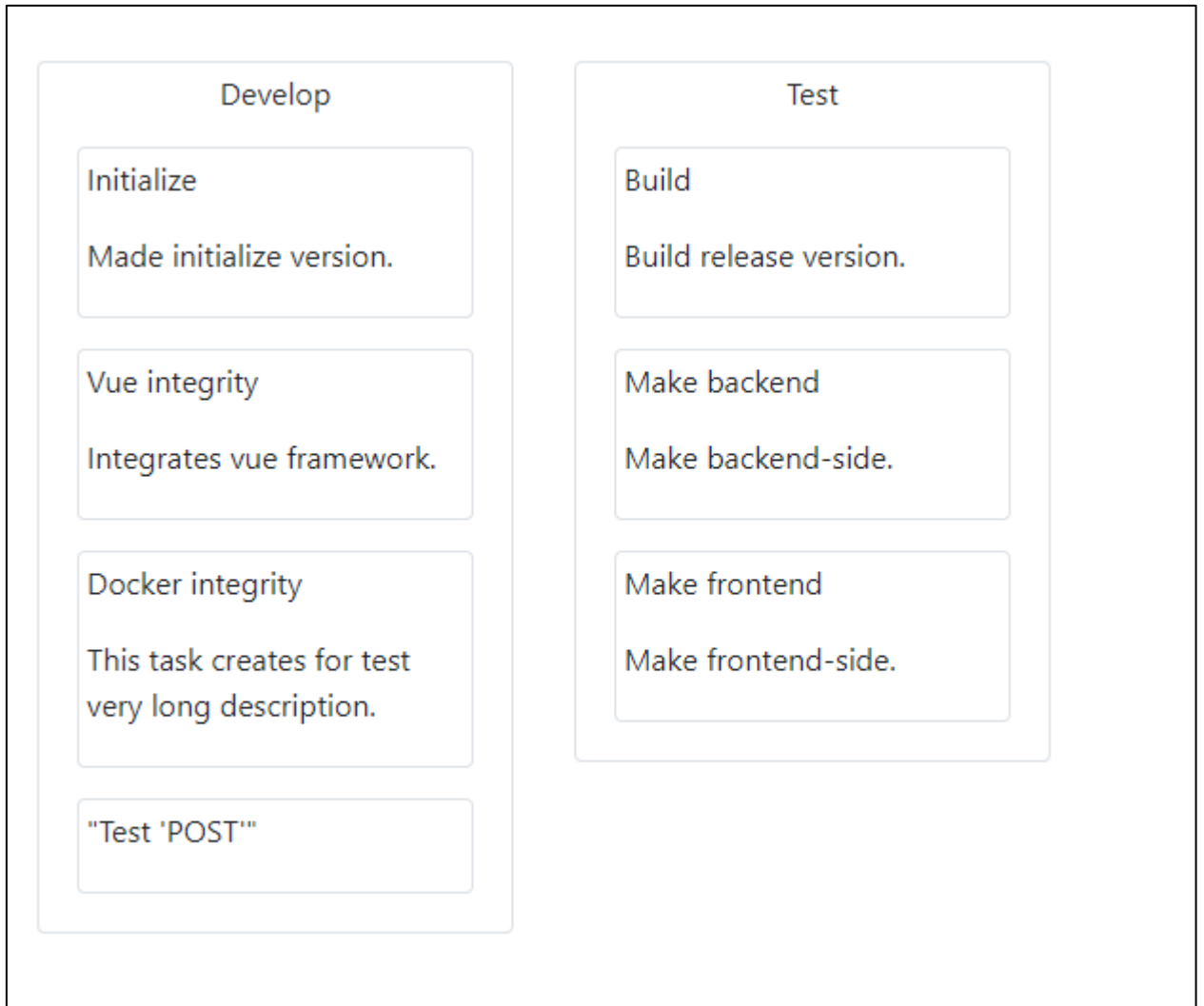
☒ I accept the terms of the [license agreement](#)

Register

```

    },
    {
      "id": 7,
      "name": "4",
      "displayName": "4",
      "email": "4@4.4",
      "password": "4"
    }
  ]
}

```



```

    },
    {
      "description": "This task creates for test very long description.",
      "parentTask": null
    },
    {
      "id": 7,
      "name": "\"Test 'POST'\"",
      "description": "",
      "parentTask": null
    }
  ]
]

```