Федеральное государственное бюджетное образовательное учреждение высшего

образования

«Чувашский государственный университет им. И.Н. Ульянова»

Кафедра вычислительной техники

Лабораторная работа № 8

**Документирование вебAPI**

Вариант 9

Выполнил: Иванов В.С.

студент группы ИВТ-41-22

Проверил: кандидат технических наук

Ржавин Вячеслав Валентинович

Чебоксары, 2025

JSON-структуры данных

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Endpoint | Параметры | Ответы |
| Get | /plots |  | {  "id": 1,  "owner\_id": 5,  "renter\_id": 3,  "price\_item\_id": 12,  "address": "ул. Ленина, д. 10",  "area": 150.5,  "description": "Земельный участок под сельское хозяйство"  } |
| /plots/{id} |  | {  "id": 1,  "owner\_id": 5,  "renter\_id": null,  "price\_item\_id": 12,  "address": "ул. Ленина, д. 10",  "area": 150.5,  "description": "Земельный участок под сельское хозяйство"  } |
| /plots/{id}/shedule |  | {  "plot\_id": 1,  "schedule": [  {  "date": "2024-01-15",  "status": "booked",  "renter\_id": 3  }  ]  } |
| /users/me/bookings |  | {  "bookings": [  {  "id": 7,  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-01-01",  "end\_date": "2024-12-31",  "status": "active"  }  ]  } |
| /applications |  | {  "applications": [  {  "id": 10,  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-02-01",  "end\_date": "2024-02-28",  "status": "pending"  }  ]  } |
| /reports |  | {  "report\_type": "payments",  "period\_start": "2024-01-01",  "period\_end": "2024-01-31",  "data": [  {  "id": 15,  "contract\_id": 7,  "amount": 15000.25,  "payment\_date": "2024-01-15",  "status": "completed"  }  ],  "total\_amount": 15000.25  } |
| Post | /applications | {  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-02-01",  "end\_date": "2024-02-28"  } | {  "id": 10,  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-02-01",  "end\_date": "2024-02-28",  "status": "pending"  } |
| /applicastions/{id}/conracts |  | {  "id": 7,  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-01-01",  "end\_date": "2024-12-31",  "status": "active"  } |
| /applicastions/{id}/payment | {  "amount": 15000.25,  "payment\_date": "2024-01-15"  } | {  "id": 15,  "contract\_id": 7,  "amount": 15000.25,  "payment\_date": "2024-01-15",  "status": "completed"  } |
| /applicastions/{id}/acceptance\_act | {  "act\_details": "Акт приема-передачи участка",  "act\_date": "2024-02-01"  } | {  "id": 8,  "contract\_id": 7,  "act\_details": "Акт приема-передачи участка",  "act\_date": "2024-02-01",  "status": "signed"  } |
| /plots | {  "owner\_id": 5,  "price\_item\_id": 12,  "address": "ул. Новая, д. 25",  "area": 200.0,  "description": "Участок под строительство"  } | {  "id": 2,  "owner\_id": 5,  "renter\_id": null,  "price\_item\_id": 12,  "address": "ул. Новая, д. 25",  "area": 200.0,  "description": "Участок под строительство"  } |
| Put | /users/me | {  "name": "Новое имя",  "phone": "+79123456789",  "email": "new@example.com"  } | {  "id": 3,  "name": "Новое имя",  "phone": "+79123456789",  "email": "new@example.com"  } |
| /plots/{id} | {  "address": "ул. Обновленная, д. 1",  "area": 210.0,  "description": "Обновленное описание"  } | {  "id": 1,  "owner\_id": 5,  "renter\_id": null,  "price\_item\_id": 12,  "address": "ул. Обновленная, д. 1",  "area": 210.0,  "description": "Обновленное описание"  } |
| /applications/{id}/approve |  | {  "id": 10,  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-02-01",  "end\_date": "2024-02-28",  "status": "approved"  } |
| /applications/{id}/reject |  | {  "id": 10,  "plot\_id": 1,  "renter\_id": 3,  "start\_date": "2024-02-01",  "end\_date": "2024-02-28",  "status": "rejected"  } |
| Delete | /plots/{id} |  | {  "message": "Участок удален",  "id": 1  } |

Ручная спецификация OpenAPI



1. Объект openapi

openapi: "3.0.2"

1. Объект info

Объект info содержит основную информацию о вашем API, включая заголовок, описание, версию, ссылку на лицензию, ссылку на условия обслуживания и контактную информацию. Многие из свойств являются необязательными.

info:

title: API системы аренды земельных участков

description: API для управления арендой земельных участков, заявками и договорами

version: 1.0.0

contact:

name: Техническая поддержка

email: support@landrental.com

url: https:// rent.com/support

license:

name: Проприетарная лицензия

url: <https://rent.com/license>

1. Объект servers

В объекте servers указывается базовый путь, используемый в ваших запросах API. Базовый путь - это часть URL, которая находится перед конечной точкой.

servers:

- url: https://api.landrental.com/data/1.0/

description: Продакшен сервер

- url: https://staging-api.landrental.com/data/1.0/

description: Тестовый сервер

- url: https://dev-api.landrental.com/data/1.0/

description: Сервер разработки

1. Объект paths

Объект paths содержит соль информации нашей API. Объект paths имеет несколько подобъектов: [объект Элемент пути](https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md#pathItemObject), [объект Operation](https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md#operationObject) и многое другое.

paths:

/plots:

get:

tags:

- Участки

summary: Получить список всех участков

description: Возвращает полный список земельных участков без фильтрации с поддержкой пагинации

operationId: getPlots

externalDocs:

description: Подробнее о фильтрации участков

url: https://docs.landrental.com/api/v1/plots

parameters:

- name: page

in: query

description: Номер страницы для пагинации

required: false

schema:

type: integer

default: 1

minimum: 1

- name: per\_page

in: query

description: Количество элементов на странице

required: false

schema:

type: integer

default: 20

minimum: 1

maximum: 100

responses:

'200':

description: Успешный ответ со списком участков

content:

application/json:

schema:

type: object

properties:

plots:

type: array

items:

$ref: '#/components/schemas/Plot'

pagination:

$ref: '#/components/schemas/Pagination'

examples:

success:

value:

plots:

- id: 1

owner\_id: 5

renter\_id: null

price\_item\_id: 12

address: "ул. Ленина, д. 10"

area: 150.5

description: "Земельный участок под сельское хозяйство"

- id: 2

owner\_id: 3

renter\_id: 7

price\_item\_id: 15

address: "ул. Центральная, д. 25"

area: 200.0

description: "Участок под строительство"

pagination:

total: 2

page: 1

per\_page: 20

total\_pages: 1

'400':

description: Неверные параметры запроса

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

examples:

invalidPage:

value:

code: "VALIDATION\_ERROR"

message: "Параметр 'page' должен быть положительным числом"

'500':

description: Внутренняя ошибка сервера

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

1. Объект components

Объект components уникален среди других объектов в спецификации OpenAPI. В components хранятся переиспользуемые определения, которые могут появляться в нескольких местах в документе спецификации. В нашем сценарии документации API мы будем хранить детали для объектов parameters и responses в объекте components.

components:

schemas:

Plot:

type: object

required:

- id

- owner\_id

- price\_item\_id

- address

- area

- description

properties:

id:

type: integer

format: int64

description: Уникальный идентификатор участка

example: 1

owner\_id:

type: integer

format: int64

description: Идентификатор владельца участка

example: 5

renter\_id:

type: integer

format: int64

nullable: true

description: Идентификатор текущего арендатора

example: 3

price\_item\_id:

type: integer

format: int64

description: Идентификатор строки прейскуранта

example: 12

address:

type: string

description: Физический адрес участка

example: "ул. Ленина, д. 10"

maxLength: 255

area:

type: number

format: float

description: Площадь участка в квадратных метрах

example: 150.5

minimum: 0

description:

type: string

description: Детальное описание участка

example: "Земельный участок под сельское хозяйство"

maxLength: 1000

Error:

type: object

required:

- code

- message

properties:

code:

type: string

description: Код ошибки

example: "INTERNAL\_SERVER\_ERROR"

message:

type: string

description: Сообщение об ошибке

example: "Произошла внутренняя ошибка сервера"

details:

type: string

description: Детальная информация об ошибке

example: "Database connection failed"

1. Объект security

[Объект security](https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md#securityRequirementObject) указывает протокол безопасности или авторизации, используемый при отправке запросов.

security:

- BearerAuth: []

И вставим объект securitySchemes в объект components (на том же уровне, что и parameters и responses):

securitySchemes:

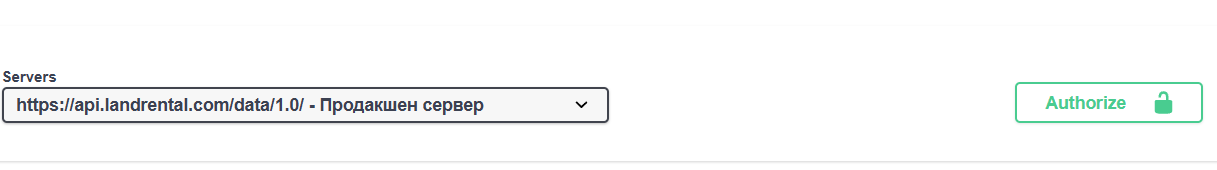
BearerAuth:

type: http

scheme: bearer

bearerFormat: JWT

description: JWT токен авторизации



Появилась кнопка авторизации

1. Объект tags

На корневом уровне [объект tags](https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md#tagObject) перечисляет все теги, которые используются в [объектах operations](https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md#operationObject) (которые появляются в объекте paths, как описано в [Шаге 4: объект paths](https://starkovden.github.io/step4-paths-object.html) ).

tags:

- name: Участки

description: Операции с земельными участками

- name: Заявки

description: Управление заявками на аренду

- name: Договоры

description: Операции с договорами аренды

- name: Пользователи

description: Управление пользователями и профилями

- name: Прейскурант

description: Управление прейскурантом

- name: Строка прейскуранта

description: Управление строкой прейскуранта

1. Объект externalDocs

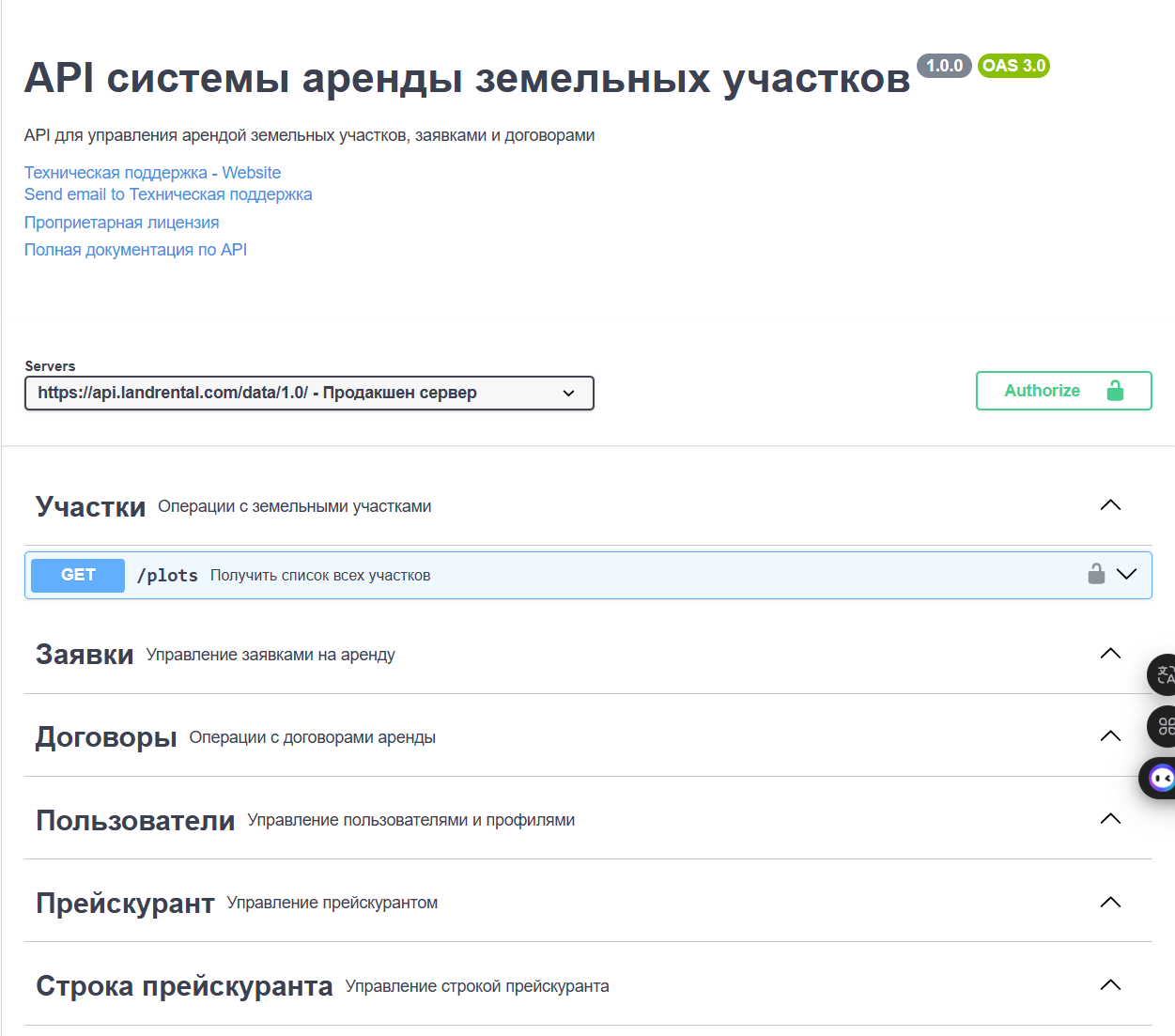
[Объект externalDocs](https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md#external-documentation-object) позволяет добавлять ссылки на внешнюю документацию. Можно добавлять ссылки на внешние документы в объекте paths.

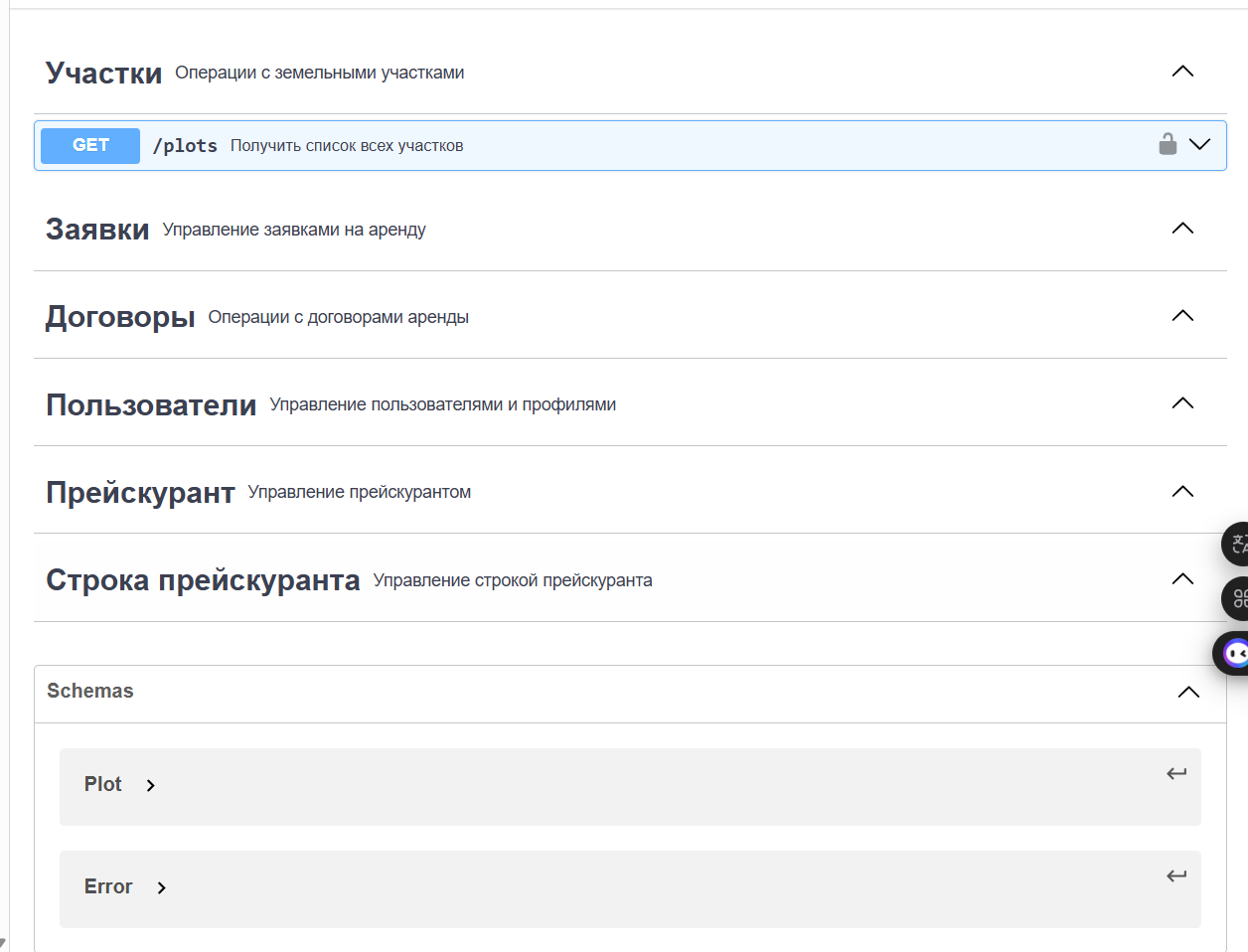
externalDocs:

description: Полная документация по API

url: <https://docs.rent.com/api/v1>

полученная спецификация





**openapi**: "3.0.2"

**info**:

  title: API системы аренды земельных участков

  description: API для управления арендой земельных участков, заявками и договорами

  version: 1.0.0

  contact:

    name: Техническая поддержка

    email: support@landrental.com

    url: https://landrental.com/support

  license:

    name: Проприетарная лицензия

    url: https://landrental.com/license

**servers**:

  - url: https://api.landrental.com/data/1.0/

    description: Продакшен сервер

  - url: https://staging-api.landrental.com/data/1.0/

    description: Тестовый сервер

  - url: https://dev-api.landrental.com/data/1.0/

    description: Сервер разработки

**paths**:

**/plots**:

**get**:

      tags:

        - Участки

      summary: Получить список всех участков

      description: Возвращает полный список земельных участков

      operationId: getPlots

**security**:

        - BearerAuth: []

**externalDocs**:

        description: Подробнее о работе с участками

        url: https://docs.landrental.com/api/v1/plots

*responses*:

        '200':

          description: Успешный ответ со списком участков

          content:

            application/json:

**schema**:

                type: array

**items**:

                  $ref: *'#/components/schemas/Plot'*

              examples:

                success:

                  value:

                    - id: 1

                      owner\_id: 5

                      renter\_id: null

                      price\_item\_id: 12

                      address: "ул. Ленина, д. 10"

                      area: 150.5

                      description: "Земельный участок под сельское хозяйство"

                    - id: 2

                      owner\_id: 3

                      renter\_id: 7

                      price\_item\_id: 15

                      address: "ул. Центральная, д. 25"

                      area: 200.0

                      description: "Участок под строительство"

        '400':

          description: Неверные параметры запроса

          content:

            application/json:

**schema**:

                $ref: *'#/components/schemas/Error'*

        '500':

          description: Внутренняя ошибка сервера

          content:

            application/json:

**schema**:

                $ref: *'#/components/schemas/Error'*

**components**:

**schemas**:

**Plot**:

      type: object

      required:

        - id

        - owner\_id

        - price\_item\_id

        - address

        - area

        - description

      properties:

**id**:

          type: integer

          format: int64

          description: Уникальный идентификатор участка

          example: 1

**owner\_id**:

          type: integer

          format: int64

          description: Идентификатор владельца участка

          example: 5

**renter\_id**:

          type: integer

          format: int64

          nullable: true

          description: Идентификатор текущего арендатора

          example: 3

**price\_item\_id**:

          type: integer

          format: int64

          description: Идентификатор строки прейскуранта

          example: 12

**address**:

          type: string

          description: Физический адрес участка

          example: "ул. Ленина, д. 10"

          maxLength: 255

**area**:

          type: number

          format: float

          description: Площадь участка в квадратных метрах

          example: 150.5

          minimum: 0

**description**:

          type: string

          description: Детальное описание участка

          example: "Земельный участок под сельское хозяйство"

          maxLength: 1000

**Error**:

      type: object

      required:

        - code

        - message

      properties:

**code**:

          type: string

          description: Код ошибки

          example: "INTERNAL\_SERVER\_ERROR"

**message**:

          type: string

          description: Сообщение об ошибке

          example: "Произошла внутренняя ошибка сервера"

**details**:

          type: string

          description: Детальная информация об ошибке

          example: "Database connection failed"

**securitySchemes**:

    BearerAuth:

      type: http

      scheme: bearer

      bearerFormat: JWT

      description: JWT токен авторизации

**security**:

  - BearerAuth: []

**tags**:

  - name: Участки

    description: Операции с земельными участками

  - name: Заявки

    description: Управление заявками на аренду

  - name: Договоры

    description: Операции с договорами аренды

  - name: Пользователи

    description: Управление пользователями и профилями

  - name: Прейскурант

    description: Управление прейскурантом

  - name: Строка прейскуранта

    description: Управление строкой прейскуранта

**externalDocs**:

  description: Полная документация по API

  url: https://docs.rent.com/api/v1

Разработка веб страницы с помощью фреймворка Swagger UI