

## Тест кейсы для API Дом питомца

Вызов API	Результат выполнения запроса	Примечания
<p><b>GET</b> <code>https://petfriends1.herokuapp.com/api/key</code></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>email: зарегистрированный</li><li>password: верный</li></ul>	<ul style="list-style-type: none"><li>Код ответа 200 OK</li><li>Вход в личный кабинет пользователя</li><li>Заголовки ответа содержат:<ul style="list-style-type: none"><li>content-type: text/json</li><li>date: &lt;дата&gt;</li></ul></li><li>В теле приходит секретный ключ для дальнейшей работы с API</li><li>Время выполнения: не дольше 1с</li></ul>	<p>Позитивный тест с опциональными параметрами. Введены корректные данные (email и password), личный кабинет пользователя доступен для дальнейшей работы</p>
<p><b>GET</b> <code>https://petfriends1.herokuapp.com/api/key</code></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>email: незарегистрированный</li><li>password: любой</li></ul>	<ul style="list-style-type: none"><li>Код ответа 403 Forbidden</li><li>Вход в личный кабинет пользователя не выполнен</li><li>Заголовки ответа содержат:<ul style="list-style-type: none"><li>content-type: text/html;</li><li>charset=utf-8</li><li>date: &lt;дата&gt;</li></ul></li><li>В теле приходит информация о том, что пользователь с такими данными не найден в базе данных</li><li>Время выполнения: не дольше 1с</li></ul>	<p>Негативный тест с некорректным вводом. Ожидаемый код ответа 401 Unauthorized (запрос не был применен, поскольку ему не хватает действительных учетных данных для целевого ресурса)</p>

<p>GET https://petfriends1.herokuapp.com/api/key</p> <p>Headers:</p> <ul style="list-style-type: none"><li>email: набор     рандомных символов</li><li>password: любой</li></ul>	<ul style="list-style-type: none"><li>Код ответа 403 Forbidden</li><li>Вход в личный кабинет     пользователя не выполнен</li><li>Заголовки ответа содержат:<ul style="list-style-type: none"><li>content-type: text/html;</li><li>charset=utf-8</li></ul></li><li>В теле приходит информация о том, что пользователь с такими данными не найден в базе данных<ul style="list-style-type: none"><li>Время выполнения: не дольше     1с</li></ul></li></ul>	Негативный тест с некорректным вводом
<p>GET https://petfriends1.herokuapp.com/api/key</p> <p>Headers:</p> <ul style="list-style-type: none"><li>email: число</li><li>password: любой</li></ul>	<ul style="list-style-type: none"><li>Код ответа 403 Forbidden</li><li>Вход в личный кабинет     пользователя не выполнен</li><li>Заголовки ответа содержат:<ul style="list-style-type: none"><li>content-type: text/html;</li><li>charset=utf-8</li></ul></li><li>В теле приходит информация о том, что пользователь с такими данными не найден в базе данных<ul style="list-style-type: none"><li>Время выполнения: не дольше     1с</li></ul></li></ul>	Негативный тест с некорректным вводом
<p>GET https://petfriends1.herokuapp.com/api/key</p> <p>Headers:</p> <ul style="list-style-type: none"><li>email: кириллица</li><li>password: любой</li></ul>	<ul style="list-style-type: none"><li>Код ответа 403 Forbidden</li><li>Вход в личный кабинет     пользователя не выполнен</li><li>Заголовки ответа содержат:<ul style="list-style-type: none"><li>content-type: text/html;</li><li>charset=utf-8</li></ul></li><li>В теле приходит информация о том, что возникла ошибка TypeError<ul style="list-style-type: none"><li>Время выполнения: не дольше     1с</li></ul></li></ul>	Негативный тест с некорректным вводом. Не удалось прочитать свойство «заголовки» из «RequestInit»: строка содержит кодовую точку, отличную от ISO-8859-1

<p><b>GET</b> <code>https://petfriends1.herokuapp.com/api/key</code></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>• email: пустой ввод</li><li>• password: любой</li></ul>	<ul style="list-style-type: none"><li>• система требует ввод email</li><li>• Время выполнения: не дольше 1с</li></ul>	<p>Негативный тест с пустым вводом. Email является обязательным полем для ввода</p>
<p><b>GET</b> <code>https://petfriends1.herokuapp.com/api/key</code></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>• email: зарегистрированный</li><li>• password: неверный</li></ul>	<ul style="list-style-type: none"><li>• Код ответа 403 Forbidden</li><li>• Вход в личный кабинет пользователя не выполнен</li><li>• Заголовки ответа содержат:<ul style="list-style-type: none"><li>- content-type: text/html;</li><li>charset=utf-8</li><li>- date: &lt;дата&gt;</li></ul></li><li>• В теле приходит информация о том, что пользователь с такими данными не найден в базе данных</li><li>• Время выполнения: не дольше 1с</li></ul>	<p>Негативный тест с некорректным вводом. Ожидаемый код ответа 401 Unauthorized (запрос не был применен, поскольку ему не хватает действительных учетных данных для целевого ресурса)</p>
<p><b>GET</b> <code>https://petfriends1.herokuapp.com/api/key</code></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>• email: зарегистрированный</li><li>• password: пустой</li></ul>	<ul style="list-style-type: none"><li>• система требует ввода password</li><li>• Время выполнения: не дольше 1с</li></ul>	<p>Негативный тест с пустым вводом. Password является обязательным полем для ввода</p>

POST  
https://petfriends1.herokuapp.com/api/create\_pet\_simple

Headers:

- auth\_key: корректный ключ API
- formData:
- name: введены корректные значения
- animal\_type: введены корректные значения
- age: введены корректные значения в цифровом формате

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах pets: -
  - age: целое число -
  - animal\_type: строка -
  - created\_at: строка со снимком времени (timestamp) - id: uuid
  - name: строка -
  - user\_id: guid
- Заголовки ответа: -
  - content-type: text/html;
  - charset=utf-8 -
  - date: <дата> - connection: keep-alive - server: gunicorn/20.0.4 - vary: Origin, Cookie - via: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест. При вводе имени на кириллице может возникнуть ошибка (на сайте имя будет отображаться некорректно)

POST  
https://petfriends1.herokuapp.com/api/create\_pet\_simple

- Headers:
- auth\_key: корректный ключ API
  - formData:
  - name: введены цифровые значения
  - animal\_type: введены корректные значения
  - age: введены корректные значения в цифровом формате

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах pets: -  
age: целое число -  
animal\_type: строка -  
created\_at: строка со снимком времени (timestamp) - id: uuid  
- name: строка -  
user\_id: guid
- Заголовки ответа: -  
content-type: text/html;  
charset=utf-8 -  
date: <дата> - connection:  
keep-alive - server:  
unicorn/20.0.4 - vary: Origin,  
Cookie - via: 1.1  
vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест. Тест пройден, но все же вряд ли кто-то назвал бы животное цифрой. Я бы поставила ограничения на ввод только буквенных символов

POST  
https://petfriends1.herokuapp.com/api/create\_pet\_simple

- Headers:
- auth\_key: корректный ключ API
  - formData:
  - name: введены спецсимволы
  - animal\_type: введены корректные значения
  - age: введены корректные значения в цифровом формате

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах pets: -  
age: целое число -  
animal\_type: строка -  
created\_at: строка со снимком времени (timestamp) - id: uuid  
- name: строка -  
user\_id: guid
- Заголовки ответа: -  
content-type: text/html;  
charset=utf-8 -  
date: <дата> - connection:  
keep-alive - server:  
unicorn/20.0.4 - vary: Origin,  
Cookie - via: 1.1  
vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест. Тест пройден, но все же вряд ли кто-то назвал бы животное спецсимволом. Я бы поставила ограничения на ввод только буквенных символов

**POST**  
**https://petfriends1.herokuapp.com/api/create\_pet\_simple**

**Headers:**

- **auth\_key:** корректный ключ API
- **formData:**
  - **name:** пустая строка
  - **animal\_type:** введены корректные значения
  - **age:** введены корректные значения в цифровом формате

- **система требует ввод данных в поле “name”**
- **Время выполнения: не дольше 1с**

**Негативный тест с пустым вводом. name является обязательным полем для ввода**

POST  
https://petfriends1.herokuapp.com/api/create\_pet\_simple

Headers:

- auth\_key: корректный ключ API
- formData:
- name: введены корректные значения
- animal\_type: введены цифровые значения
- age: введены корректные значения в цифровом формате

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах pets: -
  - age: целое число -
  - animal\_type: строка -
  - created\_at: строка со снимком времени (timestamp) - id: uuid
  - name: строка -
  - user\_id: guid -
- Заголовки ответа: -
  - content-type: text/html;
  - charset=utf-8 -
  - date: <дата> - connection: keep-alive - server: gunicorn/20.0.4 - vary: Origin, Cookie - via: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест. Тест пройден, но все же вряд ли существует тип животного с цифровым значением. Я бы поставила ограничения на ввод только буквенных символов. На сайте информация в карточке питомца может исказиться (вместо имени появятся символы, возраст и тип животного будут не те, что указывал клиент)



POST  
https://petfriends1.herokuapp.com/api/create\_pet\_simple

Headers:

- auth\_key: корректный ключ API
- formData:
- name: введены корректные значения
- animal\_type: введены спецсимволы
- age: введены корректные значения в цифровом формате

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах pets: -
  - age: целое число -
  - animal\_type: строка -
  - created\_at: строка со снимком времени (timestamp) - id: uuid
  - name: строка -
  - user\_id: guid
- Заголовки ответа: -
  - content-type: text/html;
  - charset=utf-8 -
  - date: <дата> - connection: keep-alive - server: gunicorn/20.0.4 - vary: Origin, Cookie - via: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест. Тест пройден, но все же вряд ли существует тип животного со спецсимволом. Я бы поставила ограничения на ввод только буквенных символов. На сайте информация в карточке питомца может исказиться (вместо имени появятся символы, возраст и тип животного будут не те, что указывал клиент)

**POST**  
[https://petfriends1.herokuapp.com/api/create\\_pet\\_simple](https://petfriends1.herokuapp.com/api/create_pet_simple)

**Headers:**

- **auth\_key:** корректный ключ API
- **formData:**
  - **name:** введены корректные значения
  - **animal\_type:** пустая строка
  - **age:** введены корректные значения в цифровом формате

- система требует ввод данных в поле “animal\_type”
- Время выполнения: не дольше 1с

Негативный тест с пустым вводом. animal\_type является обязательным полем для ввода

**POST**  
[https://petfriends1.herokuapp.com/api/create\\_pet\\_simple](https://petfriends1.herokuapp.com/api/create_pet_simple)

**Headers:**

- **auth\_key:** корректный ключ API
- **formData:**
  - **name:** введены корректные значения
  - **animal\_type:** введены корректные значения
  - **age:** введены буквенные значения

- система требует ввод цифровых значений
- Время выполнения: не дольше 1с

Негативный тест с некорректным вводом. age является полем для ввода цифровых значений

**POST**  
**https://petfriends1.herokuapp.com/api/create\_pet\_simple**

**Headers:**

- **auth\_key:** корректный ключ API
- **formData:**
  - **name:** введены корректные значения
  - **animal\_type:** введены корректные значения
  - **age:** введены спецсимволы

- система требует ввод цифровых значений
- **Время выполнения:** не дольше 1с

Негативный тест с некорректным вводом. age является полем для ввода цифровых значений

**POST**  
**https://petfriends1.herokuapp.com/api/create\_pet\_simple**

**Headers:**

- **auth\_key:** корректный ключ API
- **formData:**
  - **name:** введены корректные значения
  - **animal\_type:** введены корректные значения
  - **age:** пустая строка

- система требует ввод цифровых значений
- **Время выполнения:** не дольше 1с

Негативный тест с пустым вводом. age является обязательным полем для ввода

**POST**  
`https://petfriends1.herokuapp.com/api/pets`

**Headers:**

- `auth_key`: корректный ключ API
- `formData`:
- `name`: введены корректные значения
- `animal_type`: введены корректные значения
- `age`: введены корректные значения в цифровом формате

**File:**

- корректный файл

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах `pets`:
  - `age`: целое число
  - `animal_type`: строка
  - `created_at`: строка со снимком времени (timestamp)
  - `id`: uuid
  - `name`: строка
  - `user_id`: guid
  - `pet_photo (type)`: image/jpeg
- Заголовки ответа:
  - `content-type`: text/html;
  - `charset`=utf-8
  - `date`: <дата>
  - `connection`: keep-alive
  - `server`: gunicorn/20.0.4
  - `vary`: Origin, Cookie
  - `via`: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест. При вводе имени на кириллице может возникнуть ошибка (на сайте имя будет отображаться некорректно)

**POST**  
`https://petfriends1.herokuapp.com/api/pets`

**Headers:**

- `auth_key`: корректный ключ API
- `formData`:
- `name`: введены корректные значения
- `animal_type`: введены корректные значения
- `age`: введены корректные значения в цифровом формате

**File:**

- текстовый файл

- Код ответа 500 INTERNAL SERVER ERROR
- В теле приходит HTML с корректной структурой
- В теле приходят следующие поля в объектах `pets`:
  - `age`: целое число
  - `animal_type`: строка
  - `created_at`: строка со снимком времени (timestamp)
  - `id`: uuid
  - `name`: строка
  - `user_id`: guid
  - `pet_photo (type)`:

Негативный тест с некорректным вводом. Сервер обнаружил внутреннюю ошибку и не смог выполнить ваш запрос. Либо сервер перегружен, либо ошибка в приложении

	<p>application/msword</p> <ul style="list-style-type: none"><li>• <b>Заголовки ответа:</b><ul style="list-style-type: none"><li>- content-type: text/html;</li><li>- charset=utf-8</li><li>- date: &lt;дата&gt;</li><li>- connection: keep-alive</li><li>- server: gunicorn/20.0.4</li><li>- vary: Origin, Cookie</li><li>- via: 1.1 vegur</li></ul></li><li>• <b>Время выполнения: не дольше 1с</b></li></ul>	
<p><b>POST</b> <a href="https://petfriends1.herokuapp.com/api/pets">https://petfriends1.herokuapp.com/api/pets</a></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>• auth_key: корректный ключ API</li><li>• formData:<ul style="list-style-type: none"><li>- name: введены корректные значения</li><li>- animal_type: введены корректные значения</li><li>- age: введены корректные значения в цифровом формате</li></ul></li></ul> <p><b>File:</b></p> <ul style="list-style-type: none"><li>• попытка добавить два фото</li></ul>	<ul style="list-style-type: none"><li>• Система не дает возможности добавить в одну карточку два файла</li><li>• <b>Время выполнения: не дольше 1с</b></li></ul>	<p><b>Негативный тест с некорректным вводом</b></p>

**POST**  
**https://petfriends1.herokuapp.com/api/pets**

**Headers:**

- **auth\_key:** корректный ключ API
- **formData:**
  - **name:** введены корректные значения
  - **animal\_type:** введены корректные значения
  - **age:** введены корректные значения в цифровом формате

**File:**

- **пустой ввод**

- Система требует добавление файла в графу **pet\_photo**
- **Время выполнения:** не дольше 1с

Негативный тест с пустым вводом. **pet\_photo** является обязательным полем для ввода

**DELETE**  
**https://petfriends1.herokuapp.com/api/pets/{pet\_id}**

**Headers:**

- **auth\_key:** корректный ключ API

**Path:**

- **существующий ID**

- **Код ответа 200**
- В теле приходит HTML с корректной структурой
- В теле приходят URL удаленного объекта
- **Заголовки ответа:**
  - **content-type:** text/html;
  - **charset=**utf-8
  - **date:** <дата>
  - **connection:** keep-alive
  - **server:** gunicorn/20.0.4
  - **vary:** Origin, Cookie
  - **via:** 1.1 vegur
- **Время выполнения:** не дольше 1с

Базовый позитивный тест

**DELETE**  
https://petfriends1.herokuapp.com/api/pets/{pet\_id}

**Headers:**

- auth\_key: корректный ключ API

**Path:**

- несуществующий ID

- Код ответа 200
- В теле приходит HTML с корректной структурой
- В теле приходят URL удаленного объекта
- Заголовки ответа:
  - content-type: text/html;
  - charset=utf-8
  - date: <дата>
  - connection: keep-alive
  - server: gunicorn/20.0.4
  - vary: Origin, Cookie
  - via: 1.1 vegur
- Время выполнения: не дольше 1с

Негативный тест с некорректным вводом. Удаляется сущность с несуществующим ID

**DELETE**  
https://petfriends1.herokuapp.com/api/pets/{pet\_id}

**Headers:**

- auth\_key: корректный ключ API

**Path:**

- пустой ввод

- Система требует добавление информации в графу pet\_id
- Время выполнения: не дольше 1с

Негативный тест с пустым вводом. pet\_id является обязательным полем для ввода

PUT  
https://petfriends1.herokuapp.com/api/pets/{pet\_id}

Headers:

- auth\_key: корректный ключ API
- formData:
  - name: введены корректные значения
  - animal\_type: введены корректные значения
  - age: введены корректные значения в цифровом формате

Path:

- существующий ID

- Код ответа 200
- В теле приходит json с корректной структурой
- В теле приходят URL измененного объекта
- Заголовки ответа:
  - content-type: application/json
  - date: <дата>
  - connection: keep-alive
  - server: gunicorn/20.0.4
  - vary: Origin, Cookie
  - via: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест

PUT  
https://petfriends1.herokuapp.com/api/pets/{pet\_id}

Headers:

- auth\_key: корректный ключ API
- formData:
  - name: отсутствует
  - animal\_type: введены корректные значения
  - age: отсутствует

Path:

- существующий ID

- Код ответа 200
- В теле приходит json с корректной структурой
- В теле приходят URL измененного объекта
- Заголовки ответа:
  - content-type: application/json
  - date: <дата>
  - connection: keep-alive
  - server: gunicorn/20.0.4
  - vary: Origin, Cookie
  - via: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест



PUT  
https://petfriends1.herokuapp.com/api/pets/{pet\_id}

Headers:

- auth\_key: корректный ключ API
- formData:
- name: введены корректные значения
- animal\_type: введены корректные значения
- age: введены корректные значения в цифровом формате

Path:

- несуществующий ID

- Код ответа 400 BAD REQUEST
- В теле приходит HTML с корректной структурой
- В теле приходят URL несуществующего объекта, который пытались изменить
- Заголовки ответа:
  - content-type: text/html
  - charset=utf-8
  - date: <дата>
  - connection: keep-alive
  - server: gunicorn/20.0.4
  - vary: Origin, Cookie
  - via: 1.1 vegur
- Время выполнения: не дольше 1с

Негативный тест. Питомец с таким идентификатором не найден

PUT  
https://petfriends1.herokuapp.com/api/pets/{pet\_id}

Headers:

- auth\_key: корректный ключ API
- formData:
- name: отсутствует
- animal\_type: отсутствует
- age: отсутствует

Path:

- существующий ID

- Код ответа 200
- В теле приходит json с корректной структурой
- В теле приходят URL объекта, который хотели изменить
- Заголовки ответа:
  - content-type: application/json
  - date: <дата>
  - connection: keep-alive
  - server: gunicorn/20.0.4
  - vary: Origin, Cookie
  - via: 1.1 vegur
- Время выполнения: не дольше 1с

Базовый позитивный тест

<p><b>PUT</b> <code>https://petfriends1.herokuapp.com/api/pets/{pet_id}</code></p> <p><b>Headers:</b></p> <ul style="list-style-type: none"><li>• <code>auth_key</code>: корректный ключ API</li><li>• <code>formData</code>:</li><li>- <code>name</code>: отсутствует</li><li>- <code>animal_type</code>: отсутствует</li><li>- <code>age</code>: отсутствует</li></ul> <p><b>Path:</b></p> <ul style="list-style-type: none"><li>• отсутствует</li></ul>	<ul style="list-style-type: none"><li>• Система требует ввод данных в графу <code>pet_id</code></li><li>• Время выполнения: не дольше 1с</li></ul>	<p>Негативный тест с пустым вводом. <code>pet_id</code> является обязательным полем для ввода</p>
--	--	---

## Приемочное тестирование API (пользовательские сценарии)

Предусловие	В системе создан пользователь с ником <code>con.t2012@yandex.ru</code> и паролем 17011989
1.1 Шаги для повторения (взято из модуля как пример)	<ol style="list-style-type: none"><li>1. Выполнить запрос <code>GET /api/key</code> с корректными данными пользователя. Получаем ключ для пользователя</li><li>2. Выполнить запрос создания питомца — <code>POST /api/pets</code> с корректными данными. Получаем <code>id</code> созданного пользователем питомца</li><li>3. Выполнить загрузку фотографии — <code>POST /api/pets/set_photo/{pet_id}</code></li></ol>
1.2 Ожидание	В ответе на запрос <code>GET /api/pets</code> приходит созданный питомец с корректно загруженным файлом из шага 3 и данными питомца

2.1 Шаги для повторения	<ol style="list-style-type: none"> <li>1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя</li> <li>2. Выполнить запрос GET/api/pets с корректными данными пользователя. Получаем информацию о созданных объектах, в частности ID каждого питомца</li> <li>3. Выполнить запрос на удаление питомца — DELETE /api/pets/{pet_id} с корректными данными и нужным нам ID</li> <li>4. Получаем ссылку на удаленный объект</li> </ol>
2.2 Ожидание	В ответе на запрос GET /api/pets приходят данные о загруженных питомцах, исключая удаленного
3.1 Шаги для повторения	<ol style="list-style-type: none"> <li>1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя</li> <li>2. Выполнить запрос GET/api/pets с корректными данными пользователя. Получаем информацию о созданных объектах, в частности ID каждого питомца</li> <li>3. Выполнить запрос на изменение данных питомца — PUT /api/pets/{pet_id} с корректными данными и нужным нам ID. Изменяем информации. (имя, возраст или тип животного)</li> <li>4. Получаем ссылку на измененный объект</li> </ol>
3.2 Ожидание	В ответе на запрос GET /api/pets приходят измененные данные в карточке питомца, с которой мы работали
4.1 Шаги для повторения	<ol style="list-style-type: none"> <li>1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя</li> <li>2. Выполнить запрос создания питомца — POST /api/create_pet_simple с корректными данными. Получаем id созданного пользователем питомца</li> <li>3. Выполнить загрузку фотографии — POST /api/pets/set_photo/{pet_id}</li> </ol>
4.2 Ожидание	В ответе на запрос GET /api/pets приходит созданный питомец с корректно загруженным файлом из шага 3 и данными питомца

5.1 Шаги для повторения	<ol style="list-style-type: none"><li>1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя</li><li>2. Выполнить запрос GET/api/pets с корректными данными пользователя. Получаем информацию о созданных объектах, в частности ID каждого питомца</li></ol>
5.2 Ожидание	В ответе на запрос GET /api/pets приходит список всех созданных питомцев