**Сценарии тестирования для запросов API “Pet Friends”**

| **Таблица плана тестирования GET запроса api/key** |
| --- |

**GET/api/key**

| **Параметры пути** | **Параметры строки** | **Заголовки** | **Тело запроса** |
| --- | --- | --- | --- |
| --------- | --------- | 1. Email 2. Password | -------- |

**Заголовки**

| **Email: string** | **Password: string** |
| --- | --- |
| Валидный email | Валидный password |
| Невалидный email | Валидный password |
| Валидный email | Невалидный password |
| Невалидный email | Невалидный password |
| Пустая строка | Пустая строка |
| Валидный email | Пустая строка |
| Пустая стрка | Валидный password |
| Кодировка  (кириллица) | \*Валидный password |
| Валидный email | Кодировка  (кириллица) |
| Строка = 255 символов | Валидный password |
| Валидный email | Строка = 255 символов |
| Вызов неподдерживаемых запросов:  put/patch/delete  с валидным emeil | Вызов неподдерживаемых запросов:  put/patch/delete  с валидным password |

**Сценарии для проверки метода GET /api/key**

| **Вызов API** | **Результат выполнения запроса** | **Примечание** |
| --- | --- | --- |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [forteslesson@gmail.com](mailto:forteslesson@gmail.com)  password: qwerty | * Код ответа 200 * В теле приходит json c корректной структурой * В теле приходит следующее поле: (key: строка) * Заголовки ответа содержат:   Обязательно - Content-type: aplication/json  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Позитивный тест с опциональными параметрами |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [lesson@gmail.com](mailto:forteslesson@gmail.com)  password: qwerty | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением об неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным email |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [forteslesson@gmail](mailto:forteslesson@gmail.com)  password: ytrewq | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением об неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным password |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [lesson@gmail.com](mailto:forteslesson@gmail.com)  password: ytrewq | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением о неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректными email и password |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: пустая строка  password: пустая строка | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением о неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректными email и password |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [lesson@gmail.com](mailto:forteslesson@gmail.com)  password: Пустая строка | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением о неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным password |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: Пустая строка  password: qwerty | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением о неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным email |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: фортеслессон@гмаил.ком  password: qwerty | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля email * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным email |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [forteslesson@gmail](mailto:forteslesson@gmail.com)  password: йцукен | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля email * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным password |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: (Строка = 255 символов)  password: qwerty | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением о неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным email |
| **Get**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [forteslesson@gmail](mailto:forteslesson@gmail.com)  password: (Строка = 255 символов) | * Код ответа 403 (forbidden) * В теле приходит HTML - код с сообщением о неавторизованном пользователе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным password |
| **PUT/PATCH/DELETE**  https://petfriends1.herokuapp.com/api/key  **Headers**  email: [forteslesson@gmail.com](mailto:forteslesson@gmail.com)  password: qwerty | * Код ответа 405 (Method not allowed) * Тело ответа: HTML-код c сообщением о недопустимом HTTP-методе * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Деструктивное тестирование.  Вызов запроса с недопустимыми методами |

| **Таблица плана тестированя POST запроса api/pets** |
| --- |

**POST/api/pets**

| **Параметры пути** | **Параметры строки** | **Заголовки** | **Тело запроса** |
| --- | --- | --- | --- |
| --------- | --------- | 1. auth\_key | 1. name 2. animal\_type 3. age 4. pet\_photo |

**Проверка заголовка (auth\_key)**

| **auth\_key: string** | **name: string** | **animal\_type: string** | **age: int** | **pet\_photo: file: base64 string** |
| --- | --- | --- | --- | --- |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |
| Токен с истекшим сроком действия | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |
| Набор случайных символов | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |
| Спецсимволы | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |
| Кодировка  (кириллица) | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |
| Число | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |

**Проверка тела запроса (name, animal\_type, age)**

| **auth\_key: string** | **name: string** | **animal\_type: string** | **age: int** | **pet\_photo: file: base64 string** |
| --- | --- | --- | --- | --- |
| Корректный токен | Пустая строка | Корректное значение animal\_type | Корректное значение age | Корректное значение pet\_photo |
| Корректный токен | Корректное значение name | Пустая строка | Корректное значение age | Корректное значение pet\_photo |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | Пустая строка | Корректное значение pet\_photo |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Пустая строка |
| Корректный токен | Значение name = int | Значение animal\_type  = int | Корректное значение age | Корректное значение pet\_photo |
| Корректный токен | Значение name: спецсимволы | Значение animal\_type: спецсимволы | Корректное значение age | Корректное значение pet\_photo |
| Корректный токен | Значение name:  256 символов | Значение animal\_type:  256 символов | Корректное значение age | Корректное значение pet\_photo |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | значение age = str | Корректное значение pet\_photo |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | значение age = float | Корректное значение pet\_photo |

**Проверка тела запроса(pet\_photo)**

| Корректный токен | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Текстовый файл |
| --- | --- | --- | --- | --- |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Файл отсутствует |
| Корректный токен | Корректное значение name | Корректное значение animal\_type | Корректное значение age | Битый файл |

**Сценарии для проверки метода POST /api/pets**

| **Вызов API** | **Результат выполнения запроса** | **Примечание** |
| --- | --- | --- |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 200(created) * В теле приходит HTML с данными добавленного питомца * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Базовый позитивный тест |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Токен с истекшим сроком действия  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Набор случайных символов  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: спецсимволы  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: кодировка (кириллица)  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: = int  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Пустая строка  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным name |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: пустое значение  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным animal\_type |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Пустая строка  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля age * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным age |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Пустое значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля pet\_photo * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_photo |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: = int  animal\_type: = int  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name и animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным name и animal\_type |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: спецсимволы  animal\_type: спецсимволы  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name и animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным name и animal\_type |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: 256 символов  animal\_type: 256 символов  age: Корректное значение  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name и animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным name и animal\_type |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: = str  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля age * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным age |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: = float  pet\_photo: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля age * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным age |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Текстовый файл | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля pet\_photo * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_photo |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Файл отсутствует | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля pet\_photo * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_photo |
| **POST**  https://petfriends1.herokuapp.com/api/pets  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное  значение  age: Корректное значение  pet\_photo: Файл отсутствует | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля pet\_photo * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_photo |

| **Таблица плана тестирования DELETE запроса api/pets/{pet\_id}** |
| --- |

**DELETE / api/pets/{pet\_id}**

| **Параметры пути** | **Параметры строки** | **Заголовки** | **Тело запроса** |
| --- | --- | --- | --- |
| 1. pet\_id | ------------ | 1. auth\_key | -------------- |

**Проверка параметров пути (pet\_id)**

| **pet\_id: string** | **auth\_key: string** |
| --- | --- |
| Существующий id | Действующий токен |
| Пустое значение | Действующий токен |
| Несуществующий id | Действующий токен |
| Значение = int | Действующий токен |
| значение = 256 символов | Действующий токен |

**Проверка заголовков (auth\_key)**

| **pet\_id: string** | **auth\_key: string** |
| --- | --- |
| Существующий id | Токен с истекшем сроком действия |
| Существующий id | Набор случайных символов |
| Существующий id | Спецсимволы |
| Существующий id | Кириллица |
| Существующий id | Значение = int |

**Сценрии проверки метода DELETE / api/pets/{pet\_id}**

| **Вызов API** | **Результат выполнения запроса** | **Примечание** |
| --- | --- | --- |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен | * Код ответа 200(created) * В теле приходит HTML с данными удаленного питомца * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Базовый позитивный тест |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: пустое значение  **Headers:**  auth\_key: Корректный токен | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Несуществующий id  **Headers:**  auth\_key: Корректный токен | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: = int  **Headers:**  auth\_key: Корректный токен | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: = 256 символов  **Headers:**  auth\_key: Корректный токен | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Токен с истекшим сроком давности | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Набор случайных символов | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Спецсимволы | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Кириллица | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Delete**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: = int | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |

| **Таблица плана тестирования PUT запроса api/pets/{pet\_id}** |
| --- |

**PUT / api/pets/{pet\_id}**

| **Параметры пути** | **Параметры строки** | **Заголовки** | **Тело запроса** |
| --- | --- | --- | --- |
| 1. pet\_id | ------------ | 1. auth\_key | 1. name 2. animal\_type 3. age |

**Проверка параметров пути (pet\_id)**

| **pet\_id: string** | **auth\_key: string** | **name: string** | **animal\_\_type: string** | **age: int** |
| --- | --- | --- | --- | --- |
| \*Существующий id | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Пустое значение | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Несуществующий id | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Значение = int | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*значение = 256 символов | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |

**Проверка заголовков (auth\_key)**

| **pet\_id: string** | **auth\_key: string** | **name: string** | **animal\_\_type: string** | **age: int** |
| --- | --- | --- | --- | --- |
| \*Существующий id | Токен с истекшем сроком действия | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Существующий id | Набор случайных символов | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Существующий id | Спецсимволы | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Существующий id | Кириллица | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |
| \*Существующий id | Значение = int | Корректное значение  name | Корректное значение  animal\_type | Корректное значение  age |

**Проверка тела запроса (name, animal\_type, age)**

| **pet\_id: string** | **auth\_key: string** | **name: string** | **animal\_\_type: string** | **age: int** |
| --- | --- | --- | --- | --- |
| \*Существующий id | Действующий токен | Пустая строка | Корректное значение  animal\_type | Корректное значение  age |
| \*Существующий id | Действующий токен | Корректное значение  name | Пустая строка | Корректное значение  age |
| \*Существующий id | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | Пустая строка |
| \*Существующий id | Действующий токен | Значение name = int | Значение animal\_type  = int | Корректное значение  age |
| \*Существующий id | Действующий токен | Значение name: спецсимволы | Значение animal\_type: спецсимволы | Корректное значение  age |
| \*Существующий id | Действующий токен | Значение name:  256 символов | Значение animal\_type: 256 символов | Корректное значение  age |
| Существующий id | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | значение age = str |
| Существующий id | Действующий токен | Корректное значение  name | Корректное значение  animal\_type | значение age = float |

**Сценарии для проверки метода PUT / api/pets/{pet\_id}**

| **Вызов API** | **Результат выполнения запроса** | **Примечание** |
| --- | --- | --- |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 200 * В теле приходит json c корректной структурой * В теле приходит следующее поле: (   age: целое число  animal\_type: строка  created\_add: строка со снимком времени  id: uuid  name: строка  pet\_photo: base64 image  user\_id: guid)   * Заголовки ответа содержат:   Обязательно - Content-type: aplication/json  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Базовый позитивный тест |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Пустое значение  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Несуществующий id  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: =int  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: = 256 символов  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный pet\_id неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным pet\_id |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Токен с истекшем сроком действия  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Набор случайных символов  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Спецсимволы  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Кириллица  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Значение = int  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 403 (forbidden) * В теле приходит HTML с сообщением что указанный auth\_key неверен * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным auth\_key |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Пустая строка  animal\_type: Корректное значение  age: Корректное значение | * Код ответа 200 * В теле приходит json c корректной структурой * В теле приходит следующее поле: (   age: целое число  animal\_type: строка  created\_add: строка со снимком времени  id: uuid  name: строка  pet\_photo: base64 image  user\_id: guid)   * Заголовки ответа содержат:   Обязательно - Content-type: aplication/json  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur | Позитивный тест  значение name: пустая строка |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Пустая строка  age: Корректное значение | * Код ответа 200 * В теле приходит json c корректной структурой * В теле приходит следующее поле: (   age: целое число  animal\_type: строка  created\_add: строка со снимком времени  id: uuid  name: строка  pet\_photo: base64 image  user\_id: guid)   * Заголовки ответа содержат:   Обязательно - Content-type: aplication/json  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur | Позитивный тест  значение animal\_type: пустая строка |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: Пустое значение | * Код ответа 200 * В теле приходит json c корректной структурой * В теле приходит следующее поле: (   age: целое число  animal\_type: строка  created\_add: строка со снимком времени  id: uuid  name: строка  pet\_photo: base64 image  user\_id: guid)   * Заголовки ответа содержат:   Обязательно - Content-type: aplication/json  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur | Позитивный тест  значение age: пустое значение |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: = int  animal\_type: = int  age: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name и поля animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным значением поля: name и animal\_type |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: = спецсимволы  animal\_type: = спецсимволы  age: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name и поля animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным значением поля: name и animal\_type |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: = 256 символов  animal\_type: = 256 символов  age: Корректное значение | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля name и поля animal\_type * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным значением поля: name и animal\_type |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: = str | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля age * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным значением поля: age |
| **Put**  https://petfriends1.herokuapp.com/api/pets/id  **Puth:**  pet\_id: Корректный токен  **Headers:**  auth\_key: Корректный токен  **FormData:**  name: Корректное значение  animal\_type: Корректное значение  age: = float | * Код ответа 400 (Bad request) * В теле приходит HTML - код с сообщением о неверном значении поля age * Заголовки ответа содержат:   Обязательно - Content-type: text/html; charset=utf - 8  Data:<дата>  Опционально -  connection: keep-alive; server: gunicorn/20.0.4; via: 1.1 vegur   * Время выполнения: не дольше 1с. | Негативный тест с некорректным значением поля: age |

**Пользовательские сценарии для запросов API “Pet Friends”**

| **План сценариев:**   1. Как авторизованный пользователь сервиса, я хочу добавить нового питомца без фотографии. 2. Как авторизованный пользователь сервиса, я хочу добавить фотографию к уже созданному питомцу (без фотографии). 3. Как авторизованный пользователь сервиса, я хочу изменить имя у уже созданного питомца. 4. Как авторизованный пользователь сервиса, я хочу добавить ещё одного питомца с фотографией 5. Как авторизованный пользователь сервиса, я хочу удалить созданного питомца. |
| --- |

**Сценарии**

| **Сценарий** | Как авторизованный пользователь сервиса, я хочу добавить нового питомца без фотографии. |
| --- | --- |
| **Предусловие** | В системе создан пользователь с ником test\_user и паролем P@ssw0rd |
| **Шаги для**  **повторения** | 1. Выполнить запрос для получения API ключа **get/api/key** с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос для создания питомца без фотографии — **post/api/create\_pet\_simple** с корректными данными.  3. Выполнить запрос **get/api/pets** для получения информации о добавленных питомцах |
| **Ожидаемый**  **результат** | В ответ на запрос **GET /api/pets** приходит json с корректными данными созданного питомца. |

| **Сценарий** | Как авторизованный пользователь сервиса, я хочу добавить фотографию к уже созданному питомцу (без фотографии). |
| --- | --- |
| **Предусловие** | В системе создан пользователь с ником test\_user и паролем P@ssw0rd |
| **Шаги для**  **повторения** | 1. Выполнить запрос для получения API ключа **get/api/key** с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о добавленных питомцах. Получаем id добавленного питомца  3. Выполнить запрос для добавления фотографии **POST**  [**​/api​/pets​/set\_photo​/{pet\_id**}](https://petfriends1.herokuapp.com/apidocs/#/default/post_api_pets_set_photo__pet_id_)  4. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о добавленных питомцах |
| **Ожидаемый**  **результат** | В ответ на запрос **GET /api/pets** приходит json с корректными данными созданного питомца с добавленным файлом фотографии. |

| **Сценарий** | Как авторизованный пользователь сервиса, я хочу изменить имя у уже созданного питомца. |
| --- | --- |
| **Предусловие** | В системе создан пользователь с ником test\_user и паролем P@ssw0rd |
| **Шаги для**  **повторения** | 1. Выполнить запрос для получения API ключа **get/api/key** с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о добавленных питомцах. Получаем id добавленного питомца  3. Выполнить запрос для изменения информации о питомце **PUT**  [**​/api​/pets​/{pet\_id}**](https://petfriends1.herokuapp.com/apidocs/#/default/put_api_pets__pet_id_)**.** Меняем имя питомца  4. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о добавленных питомцах |
| **Ожидаемый**  **результат** | В ответ на запрос **GET /api/pets** приходит json с корректными данными питомца с изменившемся именем. |

| **Сценарий** | Как авторизованный пользователь сервиса, я хочу добавить ещё одного питомца с фотографией |
| --- | --- |
| **Предусловие** | В системе создан пользователь с ником test\_user и паролем P@ssw0rd |
| **Шаги для**  **повторения** | 1. Выполнить запрос для получения API ключа **get/api/key** с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос для создания питомца с фотографией **POST**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/post_api_pets) с корректными данными.  3. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о добавленных питомцах |
| **Ожидаемый**  **результат** | В ответ на запрос **GET /api/pets** приходит json с корректными данными созданного питомца. |

| **Сценарий** | Как авторизованный пользователь сервиса, я хочу удалить созданного питомца. |
| --- | --- |
| **Предусловие** | В системе создан пользователь с ником test\_user и паролем P@ssw0rd |
| **Шаги для**  **повторения** | 1. Выполнить запрос для получения API ключа **get/api/key** с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о добавленных питомцах. Получаем id добавленного питомца  3. Выполнить запрос для удаления питомца **DELETE**[**/api​/pets​/{pet\_id}**](https://petfriends1.herokuapp.com/apidocs/#/default/delete_api_pets__pet_id_)  4. Выполнить запрос **GET**[**/api​/pets**](https://petfriends1.herokuapp.com/apidocs/#/default/get_api_pets) для получения информации о питомцах. |
| **Ожидаемый**  **результат** | В ответ на запрос **GET /api/pets** приходит json с корректными данными, с отсутствующим, удаленным питомцем питомца. |