| Вызов API | Результат выполнения запроса | | Примечание | |
| --- | --- | --- | --- | --- |
| 1. GET-запрос /api/key  1.1 Получение API-ключа, существующая пара e-mail, password. | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail: зарегистрированный (test123@yandex.ru)  password: соответствующий пользователю (12345)  Query params: - | Код ответа 200 (OK)  Тело ответа представляет собой JSON c правильной структурой, содержит объект key: string  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Позитивный тест | |
| 1.2 Получение API-ключа, незарегистрированный e-mail. | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail: незарегистрированный( 1@yandex.ru)  password: 12345 | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением об отсутствии пользователя в БД  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Негативный тест, некорректный ввод | |
| 1.3 Получение API-ключа, зарегистрированный e-mail, не соответствующий password. | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail: зарегистрированный (test123@yandex.ru)  password: 1234567 | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением об отсутствии пользователя в БД  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Негативный тест с некорректным вводом. | |
| 1.4 Получение API-ключа, неправильный тип e-mail | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail:  пустое поле  без @ (test123.yandex.ru)  без домена (test123@.ru)  длиной в 321 символ  спецсимволы (?»;: /|\)  пробел  русские, китайские символы  password: 123456 | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением об отсутствии пользователя в БД  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Негативный тест с некорректным вводом | |
| 1.5 Получение API-ключа, неправильный тип password | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail: test123@yandex.ru  password:  пустое поле  1 символ  пробел  русские символы  число  строка >255 символов | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением об отсутствии пользователя в БД  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Негативный тест с некорректным вводом.  Текст сообщения о неправильном пароле рекомендуется не генерировать по соображениям безопасности.  Уточнить в требованиях о минимальной длине пароля | |
| 1.6 Получение API-ключа, отсутствие заголовка e-mail в запросе. | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail: отсутствует заголовок запроса  password: 123456 | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Деструктивный тест | |
| 1.7 Получение API-ключа, отсутствие заголовка password в запросе. | | | | |
| GET  <https://petfriends1.herokuapp.com/api/key>  Headers:  e-mail: test123@yandex.ru  password: отсутствует заголовок запроса | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: Sun, 30 May 2021 12:13:24 GMT  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1с | | Деструктивный тест | |
| 2. POST-запрос /api/pets  2.1 Добавление питомца c фото, корректные данные | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key: действующий ключ API  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число,  'pet\_photo': файл до 10 mb в формате JPG, JPEG, PNG' | Код ответа 200 (Created)  В теле приходит json 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  pet\_photo: base64 image  user\_id: guid  Заголовки ответа:  - обязательные:  content-type: text/html charset=utf-8  date: <дата>  - опциональные:  connection: keep-alive  server: gunicorn/20.0.4  via: 1.1 vegur  Время выполнения: не дольше 3 с | | Базовый позитивный тест | |
| 2.2 Добавление питомца с фото, некорректный api\_key | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  ключ с истекшим сроком действия  пустая строка  пробел  набор случайных символов  спецсимволы  кодировка (кириллица, китайские символы)  число  отсутствие заголовка  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число,  'pet\_photo': файл до 10 mb в формате JPG, JPEG, PNG' | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о неверном auth\_key  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом | |
| 2.3 Добавление питомца с фото, некорректное поле 'name' | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': пустая строка  пробел  спецсимволы  число  строка > 255 символов  строка > 1000 символов  отсутствие поля  'animal\_type': ‘Dog’,  'age': 4,  'pet\_photo': 'image/jpeg' | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 2.4 Добавление питомца с фото, некорректное поле 'animal\_type' | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type':  пустая строка  пробел  спецсимволы  число  строка > 255 символов  строка > 1000 символов  отсутствие поля  'age': 4,  'pet\_photo': 'image/jpeg' | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 2.5 Добавление питомца с фото, некорректное поле 'age' | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type': 'Dog'  'age': число от 1 до 255  число > 255  отрицательное число  ноль  строка  отсутствие поля  'pet\_photo': 'image/jpeg' | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 2.6 Добавление питомца с фото, некорректное поле 'pet\_photo' | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type': 'Dog'  'age': 4  'pet\_photo': 'image/tiff'  'text/html'  пустая строка  битый файл  графический нулевого размера  графический 50 mb  пустая строка  отсутствие поля | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Битый файл, большой файл, отсутствие поля - деструктивный | |
| 2.7 Добавление питомца с фото, отсутствие тела запроса | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  действующий ключ API  Payload:  отсутствует | Код ответа 400 (Bad Request)  В теле ответа HTML код c сообщением о том, что отсутствует тело запроса  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Деструктивный тест | |
| 2.8 Добавление питомца с фото, неверный формат тела запроса | | | | |
| POST  https://petfriends1.herokuapp.com/api/pets  Headers:  auth\_key:  действующий ключ API  Payload:  Бесструктурная строка (“name: Dog, animal\_type: Dog, age: 4”) | од ответа 400 (Bad Request)  В теле ответа HTML код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Деструктивный тест | |
| 3. POST-запрос /api/create\_pet\_simple  3.1 Добавление питомца без фото, корректные данные | | | | |
| POST  https://petfriends1.herokuapp.com/api/ create\_pet\_simple  Headers:  auth\_key: действующий ключ API  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число. | Код ответа 200 (Created)  В теле приходит json 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  pet\_photo: пустая строка  user\_id: guid  Заголовки ответа:  - обязательные:  content-type: application/json charset=utf-8  date: <дата>  - опциональные:  connection: keep-alive  server: gunicorn/20.0.4  via: 1.1 vegur  Время выполнения: не дольше 1 с | | Базовый позитивный тест | |
| 3.2 Добавление питомца без фото, некорректный api\_key | | | | |
| POST  https://petfriends1.herokuapp.com/api/ create\_pet\_simple  Headers:  auth\_key:  ключ с истекшим сроком действия  пустая строка  пробел  набор случайных символов  спецсимволы  кодировка (кириллица, китайские символы)  число  отсутствие заголовка  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число. | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о неверном auth\_key  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | Негативный тест c некорректным вводом | | |
| 3.3 Добавление питомца без фото, некорректное поле 'name' | | | | |
| POST  https://petfriends1.herokuapp.com/api/ create\_pet\_simple  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': пустая строка  пробел  спецсимволы  число  строка > 255 символов  строка > 1000 символов  отсутствие поля  'animal\_type': ‘Dog’,  'age': 4. | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 3.4 Добавление питомца без фото, некорректное поле 'animal\_type' | | | | |
| POST  https://petfriends1.herokuapp.com/api/ create\_pet\_simple  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type':  пустая строка  пробел  спецсимволы  число  строка > 255 символов  строка > 1000 символов  отсутствие поля  'age': 4. | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 3.5 Добавление питомца без фото, некорректное поле 'age' | | | | |
| POST  https://petfriends1.herokuapp.com/api/ create\_pet\_simple  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type': 'Dog'  'age': число от 1 до 255  число > 255  отрицательное число  ноль  строка  отсутствие поля | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный |
| 3. DELETE-запрос /api/pets/{pet\_id} 3.1 Удаление питомца, корректные данные | | | | |
| DELETE https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Path params:  pet\_id = корректный id питомца | Код ответа 200 (OK)  В теле приходит json 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  pet\_photo: base64 image  user\_id: guid  Заголовки ответа:  - обязательные:  content-type: application/json charset=utf-8  date: <дата>  - опциональные:  connection: keep-alive  server: gunicorn/20.0.4  via: 1.1 vegur  Время выполнения: не дольше 1 с | | Базовый позитивный сценарий | |
| 3.2 Удаление питомца, некорректный api\_key | | | | |
| DELETE https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  ключ с истекшим сроком действия  пустая строка  пробел  набор случайных символов  спецсимволы  кодировка (кириллица, китайские символы)  число  отсутствие заголовка  Path params:  pet\_id = корректный id питомца | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о неверном auth\_key  Заголовки ответа:  - обязательные  content-type: text/html charset=utf-8;  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 3.3 Удаление питомца, некорректный pet\_id | | | | |
| DELETE https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Path params:  id питомца:  пустая строка  строка > 255 символов  пробел  набор случайных символов  спецсимволы  кодировка (кириллица, китайские символы)  число | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: text/html charset=utf-8;  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1 с | | Негативный тест c некорректным вводом | |
| 3.4 Удаление питомца, pet\_id, привязанный к другому auth\_key | | | | |
| DELETE https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Path params:  id питомца: действующий, созданный аккаунтом с другим auth\_key | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о несоответствии {pet\_id} и auth\_key  Заголовки ответа:  - обязательные  content-type: text/html charset=utf-8;  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 1 с | | Негативный тест c некорректным вводом | |
| 4. PUT-запрос /api/pets/{pet\_id} 4.1 Изменение питомца, корректные данные | | | | |
| PUT https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число,  'pet\_photo': файл до 10 mb в формате JPG, JPEG, PNG'. | Код ответа 200 (OK)  В теле приходит json с корректной структурой  В теле приходят следующие поля в объектах pets:  age: целое число (соответствует отправленному значению),  animal\_type: строка (соответствует отправленному значению),  created\_at: строка со снимком времени  (timestamp)  id: uuid  name: строка (соответствует отправленному значению)  pet\_photo: base64 string  user\_id: guid  Заголовки ответа:  - обязательные:  content-type: application/json charset=utf-8  date: <дата>  - опциональные:  connection: keep-alive  server: gunicorn/20.0.4  via: 1.1 vegur  Время выполнения: не дольше 3 с | | Базовый позитивный сценарий | |
| 4.2 Изменение питомца, некорректный api\_key | | | | |
| PUT https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  ключ с истекшим сроком действия  пустая строка  пробел  набор случайных символов  спецсимволы  кодировка (кириллица, китайские символы)  число  отсутствие заголовка  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число,  'pet\_photo': файл до 10 mb в формате JPG, JPEG, PNG'. | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о неверном auth\_key  Заголовки ответа:  - обязательные  content-type: text/html; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом | |
| 4.3 Изменение питомца, некорректное поле 'name' | | | | |
| PUT  https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': пустая строка  пробел  спецсимволы  число  строка > 255 символов  строка > 1000 символов  отсутствие поля  'animal\_type': ‘Dog’,  'age': 4,  'pet\_photo': 'image/jpeg' | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: application/json; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 4.4 Изменение питомца, некорректное поле 'animal\_type' | | | | |
| PUT  https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type':  пустая строка  пробел  спецсимволы  число  строка > 255 символов  строка > 1000 символов  отсутствие поля  'age': 4,  'pet\_photo': 'image/jpeg' | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: application/json; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 4.5 Изменение питомца, некорректное поле 'age' | | | | |
| PUT  https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type': 'Dog'  'age': число от 1 до 255  число > 255  отрицательное число  ноль  строка  отсутствие поля  'pet\_photo': 'image/jpeg' | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: application/json; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |
| 4.6 Изменение питомца, pet\_id, привязанный к другому auth\_key | | | | |
| PUT  https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Path params:  id питомца: действующий, созданный аккаунтом с другим auth\_key  Payload:  formData:  'name': строка (кириллица, латиница),  'animal\_type': строка (кириллица, латиница),  'age': число,  'pet\_photo': файл до 10 mb в формате JPG, JPEG, PNG'. | Код ответа 403 (Forbidden)  Тело ответа: HTML-код c сообщением о несоответствии {pet\_id} и auth\_key  Заголовки ответа:  - обязательные  content-type: text/html charset=utf-8;  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом | |
| 4.7 Изменение питомца, некорректное поле 'pet\_photo' | | | | |
| PUT  https://petfriends1.herokuapp.com/api/pets/{pet\_id}  Headers:  auth\_key:  действующий ключ API  Payload:  formData:  'name': 'Dog'  'animal\_type': 'Dog'  'age': 4,  'pet\_photo': 'image/tiff'  'text/html'  пустая строка  битый файл  графический нулевого размера  графический 50 mb  пустая строка  отсутствие поля | Код ответа 400 (Bad Request)  Тело ответа: HTML-код c сообщением о неверно составленном запросе  Заголовки ответа:  - обязательные  content-type: application/json; charset=utf-8  date: <дата>  - необязательные:  access-control-allow-origin: \*  connection: keep-alive  content-length: 146  server: gunicorn/20.0.4 via: 1.1 vegur  Время выполнения: не дольше 3 с | | Негативный тест c некорректным вводом  Отсутствие поля - деструктивный | |

**Пользовательские сценарии**

|  |  |
| --- | --- |
| 1. Создание питомца с фото | |
| Предусловие | В системе создан пользователь с почтовым адресом test123@yandex.ru и паролем 123456 |
| Шаги для повторения | 1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос создания питомца — POST /api/pets с корректными данными (имя, породу, возраст, фотографию питомца). Получаем id созданного пользователем питомца. |
| Ожидание | В ответе на запрос GET /api/pets приходит созданный питомец с корректно загруженным файлом из шага 2 и данными питомца. |
| 2. Создание питомца без фото | |
| Предусловие | В системе создан пользователь с почтовым адресом test123@yandex.ru и паролем 123456 |
| Шаги для повторения | 1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос создания питомца — POST /api/pets с корректными данными (имя, породу, возраст питомца). Получаем id созданного пользователем питомца. |
| Ожидание | В ответе на запрос GET /api/pets приходит созданный питомец с корректно загруженным данными питомца. |
| 3. Редактирование информации о питомце | |
| Предусловие | В системе создан пользователь с почтовым адресом test123@yandex.ru и паролем 123456 |
| Шаги для повторения | 1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос создания питомца — POST /api/pets с корректными данными (имя, породу, возраст, фотографию питомца). Получаем id созданного пользователем питомца.  3. Выполнить запрос редактирования питомца – PUT /api/pets/{pet\_id}, применив полученный на шаге 2 pet\_id, изменить имя, породу, возраст и фотографию питомца. |
| Ожидание | В ответе на запрос GET /api/pets приходит питомец с корректно загруженным данными питомца, которые соответствуют выполненным изменениям на шаге 3. |
| 4. Удаление питомца | |
| Предусловие | В системе создан пользователь с почтовым адресом test123@yandex.ru и паролем 123456 |
| Шаги для повторения | 1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос создания питомца — POST /api/pets с корректными данными (имя, породу, возраст, фотографию питомца). Получаем id созданного пользователем питомца.  3. Выполнить запрос удаления питомца – DELETE /api/pets/{pet\_id}, применив полученный на шаге 2 pet\_id. |
| Ожидание | В ответе на запрос GET /api/pets приходит список питомцев, в котором отсутствует питомец с ранее созданным и удалённым питомцем, pet\_id которого соответствовал полученному на шаге 2. |
| 5. Получение списка питомцев | |
| Предусловие | В системе создан пользователь с почтовым адресом test123@yandex.ru и паролем 123456 |
| Шаги для повторения | 1. Выполнить запрос GET /api/key с корректными данными пользователя. Получаем ключ для пользователя.  2. Выполнить запрос получения списка питомцев GET /api/pets с фильтром my\_pets |
| Ожидание | В ответе на запрос GET /api/pets приходит список питомцев, созданных данным пользователем. |