**Продуктовые требования**

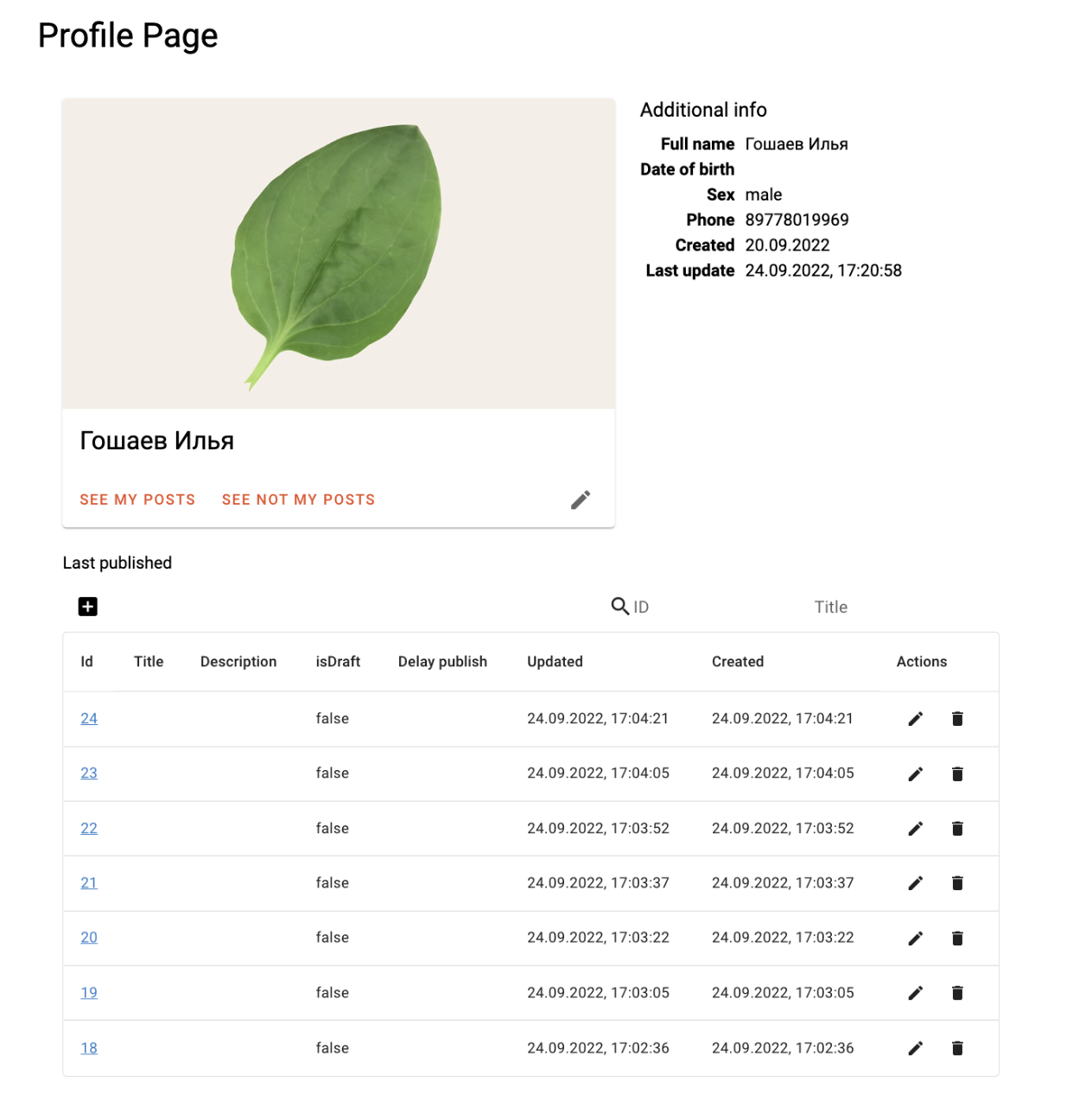
**“GeekTest – закрытая социальная сеть для тестировщиков”**

**\*запрашивать доступ к файлу не надо\***

Информационная система "GeekTest – закрытая cоциальная сеть для тестировщиков" предназначена для общения начинающих и опытных QA-специалистов.

**Модуль 3 – профиль**

Содержит информацию о личных данных пользователя, а также ленту его опубликованных постов. Позволяет редактировать данные, редактировать и удалять свои посты, осуществлять поиск по своим постам.



**Точка входа:** доступно по прямой ссылке, либо по “Hello, {userneme}” в хедере в правом углу.

**URL:** <https://test-stand.gb.ru/user/profile>

| **Требование** | **Отработка ошибки** |
| --- | --- |
| Страница доступна только авторизованным пользователям | API возвращает ошибку авторизации; фронт редиректит на страницу авторизации |
| Редактирование аватара доступно по клику на него | - |
| По клику на карандаш под аватаром открывается попап редактирования данных пользователя | - |
| По кнопке “See my posts” открывает стартовую страницу блога с личными постами | - |
| По кнопке “See not my posts” открывает первую страницу с чужими постами | - |
| По кнопке + открывается попап создания нового поста |  |
| По клику на карандашик рядом со списком постов открывается попап редактирования конкретного поста |  |
| По клику на урну пост удаляется |  |
| Можно искать посты по ID и заголовку |  |

**Техническая реализация**

GET [https://test-stand.gb.ru/api/users/profile/{user\_id](https://test-stand.gb.ru/api/users/profile/%7Buser_id)}

В ответе на запрос возвращается объект:

| {  "id": Integer,  "firstName": "TestName",  "lastName": "TestLastName",  "username": "testusername11",  "phone": "78887776660",  "status": "active",  "sex": "male",  "birthDate": "2022-10-01T00:00:00+00:00",  "avatar": {  "id": null,  "cdnUrl": ""  }  } |
| --- |

где:

id – идентификатор пользователя,

firstName – имя пользователя,

lastName – фамилия пользователя,

username – логин,

phone – номер телефона,

sex – пол (принимает значения male | female) ,

birthDate – дата рождения в формате ГГГГ-ММ-ДД (например: 2022-10-01T00:00:00+00:00),

avatar.id – id изображения аватара,

avatar.cdnUrl – ссылка на изображение аватарки.

Если мы запрашиваем юзера с чужим токеном, то в ответе

| {  "error": "critical",  "code": 400,  "message": "Вы запрашиваете чужого пользователя."  } |
| --- |

где:

code – 400,

message – текст ошибки

Если мы хотим обновить информацию о пользователе, то  
  
PUT <https://test-stand.gb.ru/gateway/users/profile/user_id>

где user\_id – id пользователя

В теле запроса передается:

| {  "image": file,  "firstName": "String",  "lastName": "String",  "username": "String",  "phone": "String",  "sex": "String",  "birthDate": "String"  } |
| --- |

где:

image – прикрепляется как файл, это аватарка пользователя,

firstName – имя пользователя,

lastName – фамилия пользователя,

username – логин,

phone – номер телефона,

sex – пол (принимает значения male | female) ,

birthDate – дата рождения в формате ГГГГ-ММ-ДД (например: 2022-10-01)

В случае неуспешного ответа, в response мы получаем:

| {  "error": "Неуспешная попытка создания аккаунта",  "code": 400,  "message": [  {  "message": "String",  "path": "username",  "code": "String"  }  ]  } |
| --- |

где:

**error** – параметр, содержащий текст "Неуспешная попытка создания аккаунта",

**code** – код ошибки, 400,

**message.message** – параметр, содержащий текст ошибки,

**message.code** – параметр, содержащий код ошибки (1 – не заполнено одно из обязательных полей, 2 – не уникальный логин, 3 – не заполнено ни одно поле).