

Документация к продукту SPB-BANK (Mikhail)

Ссылка на задачу	SCRUM-79
Общая задача по курсу	SCRUM-4
Статус страницы	В работе
Описание	Документация к сайту https://demo.bspb.ru/

▼ Описание фичей к которым идут требования

1. Экран ввода логин/пароль - demo/demo
2. Экран ввода номера для смс: номер - *1111; Код из смс - 0000
3. Экран персональных предложений - виджет "Потребительский кредит (короткая заявка)"
4. Экран реферальных программ - кнопка "Подробнее об акции"
5. Экран переписки с банком - кнопка "Отметить все как прочитанные"

▼ Описание требований

1. Длина логина < 15 символов, только латинские буквы.
 - 1.1 Пароль < 25 символов, только нижний регистр.
 - 1.2 При валидных данных и нажатие на кнопку Войти осуществляется переход к окну подтверждения по смс.
 - 1.3 При невалидных данных даётся 3 попытки авторизации, после блокируется аккаунт.
2. Номер для смс должен быть верифицирован в настройках аккаунта.
 - 2.1 Код подтверждения смс не должен быть < 4 знаков и содержать только цифры.
 - 2.2 Возможность повторной отправки кода осуществляется каждые 120 секунд.
 - 2.3 При верном коде и нажатие кнопки "Войти" должен осуществляться переход на главную страницу аккаунта.
 - 2.4 При 3 неудачных попытках входа должна происходить блокировка аккаунта.

3. Переход с главной страницы на экран персональных предложений нажатием на иконку “конфеты”.

3.1 Нажатием на виджет "Потребительский кредит (короткая заявка)" переход к оформлению заявки.

3.2 Возможность оформления ещё одной кредитной карты.

4. При нажатие на поле “ПРИГЛАСИТЬ ДРУЗЕЙ” на главной странице должен осуществляться переход на страницу реферальных программ: <https://idemo.bspb.ru/referral-program> .

4.1 Напротив каждого рекомендованного продукта должна находиться кнопка “Подробнее об акции”.

4.2 Внизу экрана при нажатие на кнопку “Подробнее об акции” осуществляется переход на страницу с описанием условий акции.

4.3 На странице условий акции должна быть кнопка перехода на страницу оформления предложенного продукта.

5. На главной странице при нажатии на иконку сообщений в верхнем правом углу осуществляется переход на страницу переписки с банком.

5.1 Индикация непрочитанных сообщений должна находиться как на странице переписки с банком, так и в числовом виде на иконке.

5.2 На странице переписки с банком непрочитанные сообщения должны выделяться жирным курсивом.\

5.3 При нажатие на кнопку “Отметить все как прочитанные” индикация непрочитанных сообщений должна меняться (исчезают цифры над иконкой, жирный курсив становится обычным).

Тест-кейсы и баг-репорты

Описание	Тест-кейс
----------	-----------

Переход на
страницу
авторизации при
нажатии на
кнопку
“Подробнее об
акции”

<https://app.qase.io/case/DEMO-18>

DEMO-23 / Projects / 17

Проверка перехода на страницу с описанием условий акций
"Подробнее об акции"


На <https://demo.bank.ru/offer/program> при рекомендации друзьям перейдите на страницу условий продукта нажатием кнопки "Подробнее об акции"

Severity	Priority	Behavior	Type
☒ Normal	🔼 Medium	Not set	Functional
Layer	Is Flaky	Milestone	Automation
Not set	No	-	Not automated
Status			
Actual			

Pre-conditions

- Авторизован на сайте <https://demo.bank.ru/>
- Logins: demo/demo
- Код из SMS: 0000

Steps to reproduce

Step	Action	Input data	Expected result
1	Нажать на кнопку "ПРОИ ПАСИТЬ ДРУЗЬЕ" на главной странице		Переход на страницу https://demo.bank.ru/offer/program
			
2	Внизу страницы нажать на кнопку " Подробнее об акции " напротив предложения "Карта для новых племей Ярды"		Переход на страницу с описанием условий акции

[Экран переписки
с банком] При
нажатии на
кнопку “Отметить
все как
прочитанные” не
обновляется
иконка
сообщений

<https://app.qase.io/case/DEMO-16>

DEMO-21 / Projects / 15

Проверка обновлений сообщений при нажатие на кнопку
"Отметить все как прочитанные"

На странице <https://demo.bank.ru/contacts> проверить на корректность кнопку сообщений "Отметить все как прочитанные"


Severity	Priority	Behavior	Type
☒ Trivial	🔼 High	Not set	Usability
Layer	Is Flaky	Milestone	Automation
Not set	No	-	Not automated
Status			
Actual			

Pre-conditions

- Зайти на сайт <https://demo.bank.ru/>
- Пройти авторизацию для Олега Королёва
- login: demo
- password: demo
- Код из SMS: 0000

Steps to reproduce

Step	Action	Input data	Expected result
1	Нажать на кнопку "Переписка с Банком" в виде письма в правом верхнем углу экрана		Переход на страницу переписки с банком https://demo.bank.ru/contacts
2	Нажать на кнопку " Отметить все как прочитанные "		<ul style="list-style-type: none">Все сообщения отмечены как прочитанныеИконка в виде письма в верхнем правом углу экрана без цифр



[Экран Потребительский кредит (короткая заявка)] Страница не найдена при нажатие на виджет "Потребительский кредит (короткая заявка)"

<https://app.qase.io/case/DEMO-17>

DEMO-22 / Projects / 16

Оформление кредита через кнопку "Персональные предложения"

Оформление потребительского кредита(короткая заявка) через кнопку "Персональные предложения" на странице <https://demo.kash.ru/>

Severity	Priority	Behavior	Type
Minor	High	Not set	Functional
Layer	Is Flaky	Milestone	Automation
Not set	No	-	Not automated
Status	Actual		

Pre-conditions

- Зайти на сайт <https://demo.kash.ru/>
- Авторизоваться как Ольга Коротева log/pass - demodemo
- Код из SMS: 0000

Steps to reproduce

Step	Action	Input data	Expected result
1	На главной странице в правом экране углу нажать на кнопку в виде конверты "Персональные предложения"		Переход на страницу персональных предложений
2	Нажать на кнопку "Потребительский кредит (короткая заявка)"		Переход на страницу оформления заявки

[Экран оплаты мобильной связи] Нет доступа к функции оплаты

<https://app.qase.io/case/DEMO-19>

DEMO-24 / Projects / 17

Оплата Мобильной связи в демо-версии

На странице <https://demo.kash.ru/mobility/phone-payment/> попытка оплаты мобильной связи

Severity	Priority	Behavior	Type
Major	High	Not set	Security
Layer	Is Flaky	Milestone	Automation
Not set	No	-	Not automated
Status	Actual		

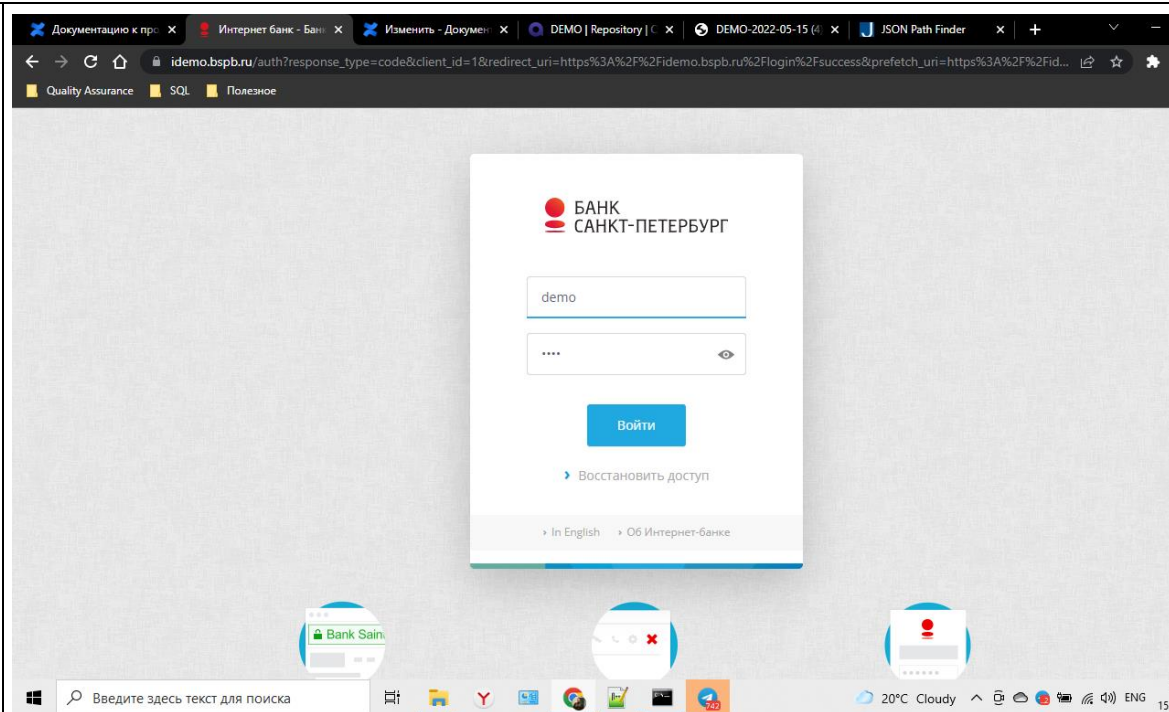
Pre-conditions

- Авторизация на <https://demo.kash.ru/>
- log/pass: demodemo
- Код из SMS: 0000

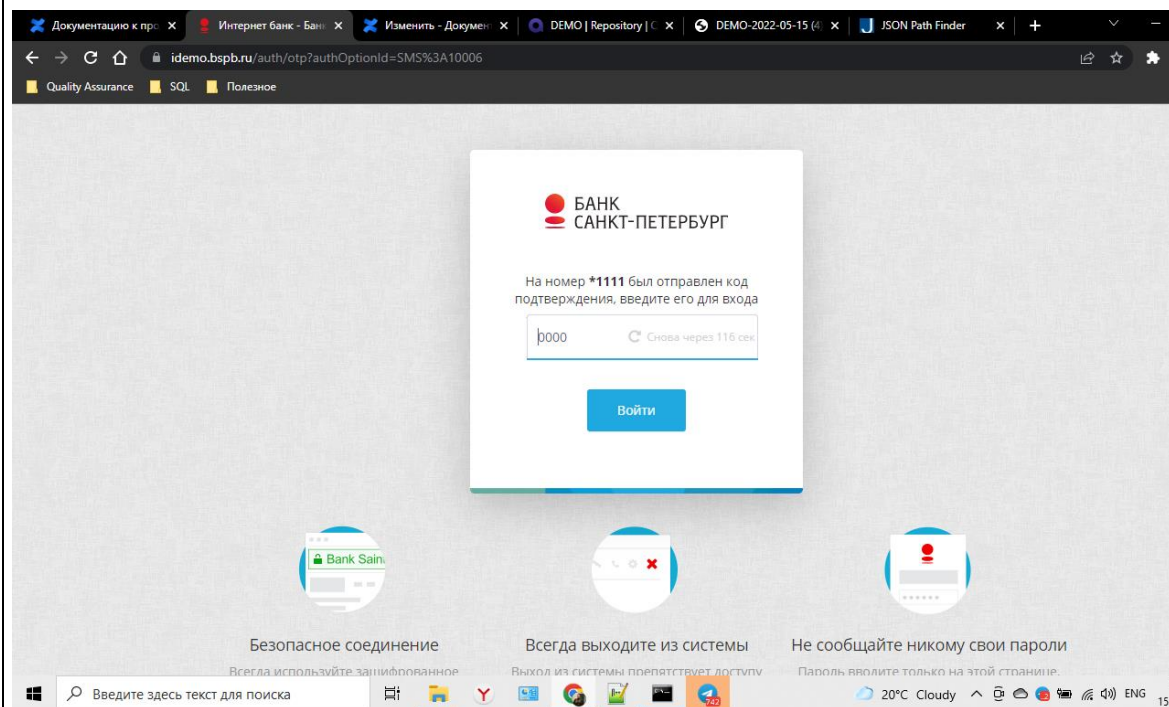
Steps to reproduce

Step	Action	Input data	Expected result
1	Нажать на кнопку "Платежи и переводы"		Переход на страницу платежей и переводов
2	Нажать на иконку "Мой телефон" с желто-черным логотипом		Переход на страницу оплаты номера
3	Указать текущую дату, сумму и нажать кнопку "Далее"	Сумма - 1000 р.	Переход на страницу подтверждения оплаты

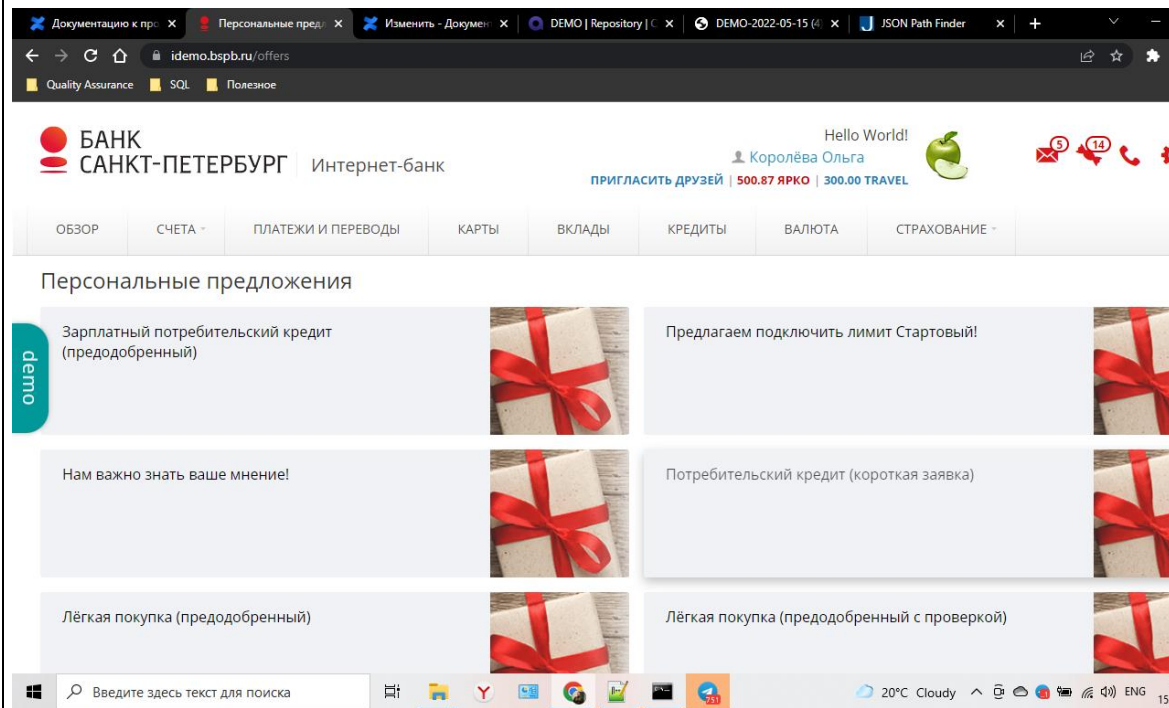
Экран ввода
логин/пароль -
demo/demo



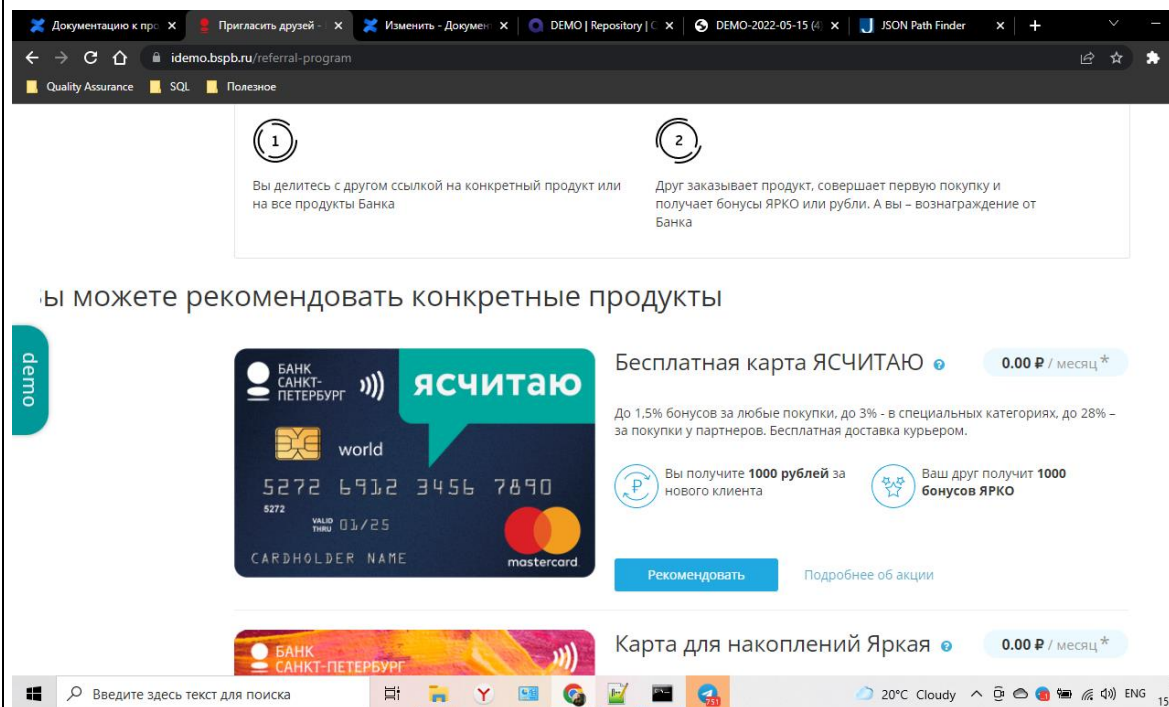
Экран ввода
номера для смс:
номер - *1111;
Код из смс - 0000



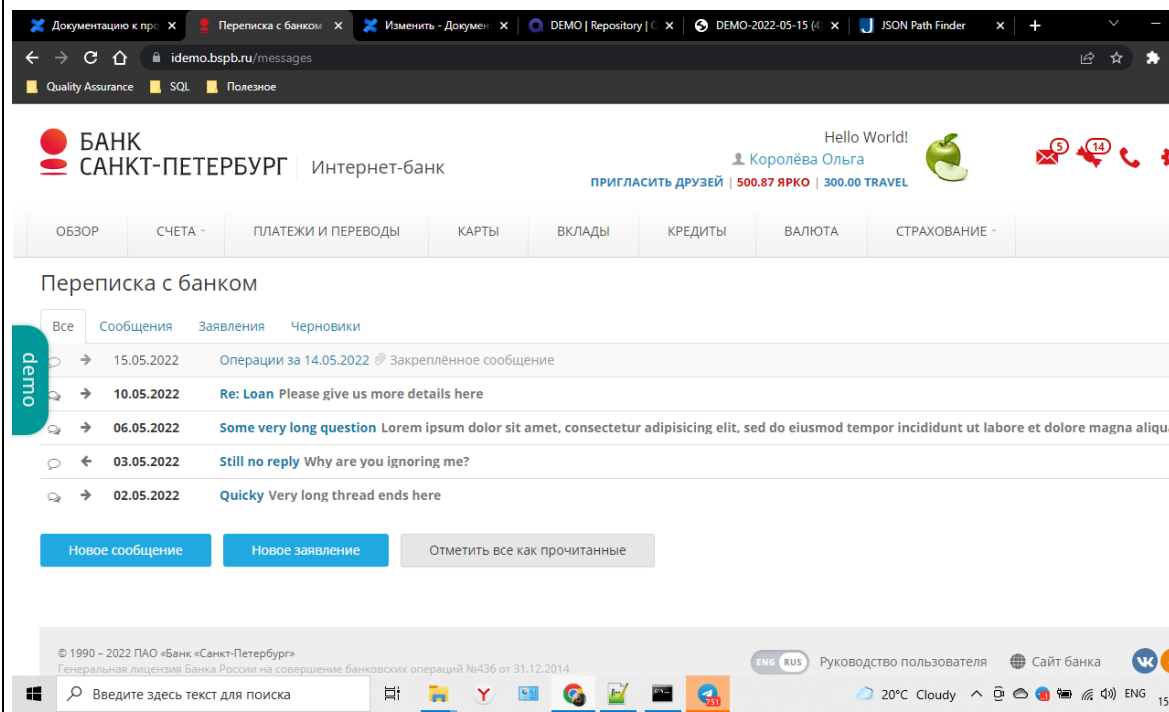
Экран
персональных
предложений -
виджет
"Потребительский
кредит (короткая
заявка)"



Экран
реферальных
программ -
кнопка
"Подробнее об
акции"

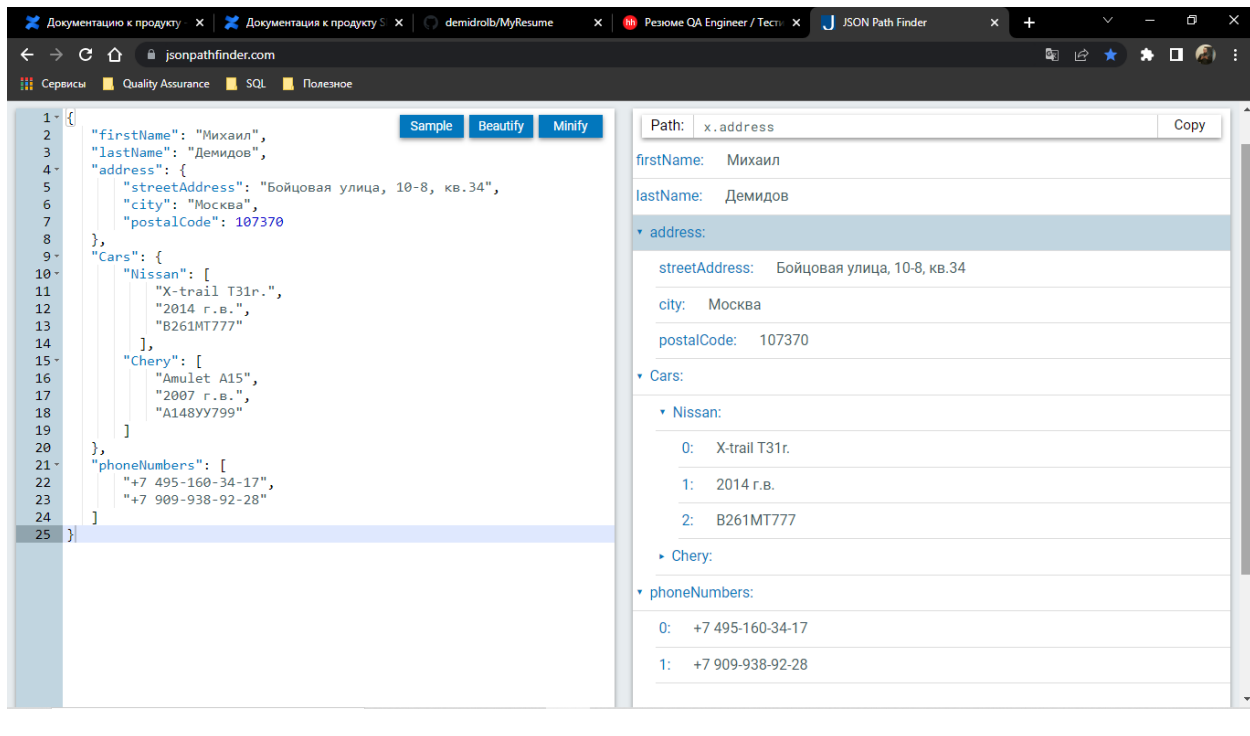


Экран переписки
с банком - кнопка
“Отметить все как
прочитанные”



Json пользователя

```
{
  "firstName": "Михаил",
  "lastName": "Демидов",
  "address": {
    "streetAddress": "Бойцовая улица, 10-8, кв.34",
    "city": "Москва",
    "postalCode": 107370
  },
  "Cars": {
    "Nissan": [
      "X-trail T31r.",
      "2014 г.в.",
      "B261MT777"
    ],
    "Chery": [
      "Amulet A15",
      "2007 г.в.",
      "A148UY799"
    ]
  },
  "phoneNumbers": [
    "+7 495-160-34-17",
    "+7 909-938-92-28"
  ]
}
```



Database spb-bank

CREATE DATABASE spbank;

USE spbank;

CREATE TABLE Authorization (
id INT(10) PRIMARY KEY AUTO_INCREMENT,
login VARCHAR(30) NOT NULL,
password VARCHAR(50) NOT NULL
);

INSERT INTO Authorization (id, login, password) VALUES (1, 'Dem', '12345'), (2, 'Svet', 'qaz1'), (3, 'Ira', '1q2w3e'), (4, 'Pet9', '9phet'), (5, 'Ivan', 'van9'), (6, 'Nick', 'n2i3c4'), (7, 'George', '928374');

CREATE TABLE Users (
id INT (10) PRIMARY KEY AUTO_INCREMENT,
firstname VARCHAR(30),
lastname VARCHAR(30),
Auth_id INT(10),
FOREIGN KEY Users (Auth_id) REFERENCES Authorization (id)
);

INSERT INTO Users (id, firstname, lastname, Auth_id) VALUES (1, 'Demidov', 'Mkhail', 1);
INSERT INTO Users (firstname, lastname, Auth_id) VALUES ('Petr', 'Yasnetsov', 2);

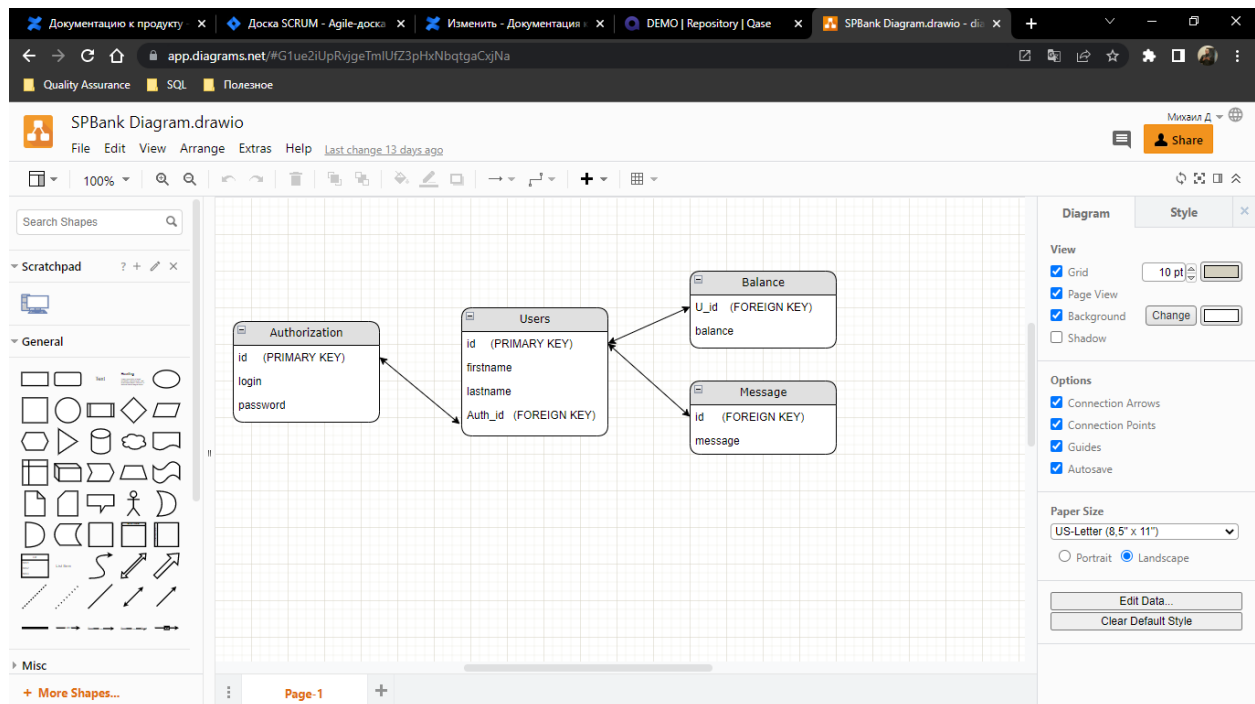

```
INSERT INTO Users (firstname, lastname, Auth_id) VALUES ('Irina', 'Demidova', 3), ('Petr', 'Yasnecov', 4), ('Ivan', 'Ivanov', 5), ('Nikolai', 'Volosyk', 6), ('George', 'Steblov', 7);
```

```
CREATE TABLE Balance (
U_id INT(10),
balance INT(30) NOT NULL,
FOREIGN KEY Balance (U_id) REFERENCES Users (id)
);
```

```
INSERT INTO Balance (U_id, balance) VALUES (1, 1500000), (2, 1750000), (3, 1940000), (4, 2750000), (5, 550000), (6, 280000), (7, 5200000);
```

```
CREATE TABLE Message (
id INT(50),
message TEXT(3000),
FOREIGN KEY Message (id) REFERENCES Users (id)
);
```

```
INSERT INTO Message (id, message) VALUES (1, 'I want more'), (1, 'i so greedy'), (2, 'i also so greedy'), (3, 'Хочу много денег'), (4, 'I am very cute mentor'), (5, 'I would like a good job'), (5, 'My love is in Paris'), (6, 'my child is so beatifull'), (7, 'I work and live in Moscow');
```



Выборка по БД:

```
SELECT A.login, A.password, U.firstname, U.lastname, B.balance, M.message FROM
Authorization A JOIN Users U ON A.id = U.Auth_id JOIN Balance B ON U.id = B.U_id JOIN
Message M ON U.id = M.id;
```

```
SELECT DISTINCT A.login, A.password, U.firstname, U.lastname, B.balance, M.message
FROM Authorization A JOIN Users U ON A.id = U
.Auth_id JOIN Balance B ON U.id = B.U_id JOIN Message M ON U.id = M.id;
```

Развернуть с помощью docker

```
docker pull mysql/mysql-server:latest
docker images (проверка, что вы установили нужный образ) (приблизительно, как
должно быть: mysql/mysql-server latest 5a9594052aec 3 days ago
438MB)
docker run -p 3307:3307 -d --name=mysql mysql/mysql-server:latest
docker ps (проверка, что контейнер запустился) (приблизительно, как должно
быть:
5e3af0cdbc8a mysql/mysql-server:latest "/entrypoint.sh mysql..." 19
minutes ago Up 19 minutes (healthy) 3306/tcp, 33060-33061/tcp,
0.0.0.0:3307->3307/tcp mysql
)
docker logs mysql
docker exec -it mysql bash
mysql -u root -p (пароль, скопированный из логов)
ALTER USER 'root'@'localhost' IDENTIFIED BY 'root';
```

Коллекция Postman



PetApi3004.postm..._collection.json

PetApi3004.json

```

{
  "info": {
    "_postman_id": "79173940-f974-4d07-beld-6d0aedc786a1",
    "name": "PetApi3004",
    "schema":
"https://schema.getpostman.com/json/collection/v2.1.0/collection.json"
  },
  "item": [
    {
      "name": "CreatePet",
      "event": [
        {
          "listen": "test",
          "script": {
            "exec": [
              pm.response.json();\r",
              "pm.environment.set(\"id\", jsonData.id);\r",
              "\r",
              "pm.test(\"Status code\r",
              "\r",
              "is 200\", function () {\r",
              "\r",
              "pm.response.to.have.status(200);\r",
              "\r",
              "});\r",
              "\r",
              "pm.test(\"Successful\r",
              "\r",
              "POST request\", function () {\r",
              "\r",
              "pm.expect(pm.response.code).to.be.oneOf([201, 202, 200]);\r",
              "});\r",
              "\r",
              "pm.test(\"Response time\r",
              "\r",
              "is less than 200ms\", function () {\r",
              "\r",
              "pm.expect(pm.response.responseTime).to.be.below(1000);\r",
              "});\r",
              "\r",
              "pm.test(\"Status code\r",
              "\r",
              "name has string\", function () {\r",
              "\r",
              "pm.response.to.have.status(\"OK\");\r",
              "\r",
              "});\r",
              "\r",
              "\r",
              ""
            ],
            "type": "text/javascript"
          }
        ]
      }
    }
  ],
  "request": {
    "method": "POST",
    "header": [],
    "body": {
      "mode": "raw",

```

```

        "raw": "{\r\n  \"id\": 826,\r\n
\"category\": {\r\n    \"id\": 826,\r\n    \"name\": \"Plusha\"\r\n  },\r\n
\"name\": \"Plusha\",\r\n  \"photoUrls\": [\r\n
\"https://planetofhotels.com/guide/sites/default/files/styles/node__blog_post
__bp_banner/public/2020-06/Exotic-tropical-beach-sunse.jpg\"\r\n  ],\r\n
\"tags\": [\r\n    {\r\n      \"id\": 826,\r\n      \"name\": \"Plusha\"\r\n    }
\r\n  ],\r\n  \"status\": \"available\"\r\n}",
        "options": {
            "raw": {
                "language": "json"
            }
        },
        "url": {
            "raw":
"https://petstore.swagger.io/v2/pet",
            "protocol": "https",
            "host": [
                "petstore",
                "swagger",
                "io"
            ],
            "path": [
                "v2",
                "pet"
            ]
        },
        "response": [
            {
                "name": "IdPet",
                "event": [
                    {
                        "listen": "test",
                        "script": {
                            "exec": [
                                "pm.test(\"Status code
is 200\", function () {\r\n",
                                "
                                "});\r\n",
                                "\r\n",
                                "pm.test(\"Response time
is less than 200ms\", function () {\r\n",
                                "
                                "pm.expect(pm.response.responseTime).to.be.below(1000);\r\n",
                                "});\r\n",
                                "\r\n",
                                "pm.test(\"Status code
name has string\", function () {\r\n",
                                "
                                "pm.response.to.have.status(\"OK\");\r\n",
                                "});\r\n",
                                "\r\n",
                                "pm.test(\"Content-Type
is present\", function () {\r\n",

```

```

pm.response.to.have.header("Content-Type");\r",
    "});"
    ],
    "type": "text/javascript"
  }
},
"request": {
  "method": "GET",
  "header": [],
  "url": {
    "raw":
"https://petstore.swagger.io/v2/pet/826",
    "protocol": "https",
    "host": [
      "petstore",
      "swagger",
      "io"
    ],
    "path": [
      "v2",
      "pet",
      "826"
    ]
  }
},
"response": []
},
{
  "name": "PutPet",
  "event": [
    {
      "listen": "test",
      "script": {
        "exec": [
is 200\"", function () {\r",
pm.response.to.have.status(200);\r",
        "});\r",
        "\r",
        "pm.test(\"Response time
is less than 200ms\"", function () {\r",
pm.expect(pm.response.responseTime).to.be.below(1000);\r",
        "});\r",
        "\r",
        "pm.test(\"Status code
name has string\"", function () {\r",
pm.response.to.have.status("OK");\r",
        "});\r",
        "\r",
        "pm.test(\"Content-Type
is present\"", function () {\r",

```

```

        "
pm.response.to.have.header("Content-Type");\r",
        "));"
    ],
    "type": "text/javascript"
  }
},
"request": {
  "method": "PUT",
  "header": [],
  "body": {
    "mode": "raw",
    "raw": "{\r\n  \"id\": 826,\r\n
\"category\": {\r\n    \"id\": 826,\r\n    \"name\": \"Plusha31\"\r\n  },\r\n
\"name\": \"Plusha32\",\r\n  \"photoUrls\": [\r\n
\"https://planetofhotels.com/guide/sites/default/files/styles/node__blog_post
__bp_banner/public/2020-06/Exotic-tropical-beach-sunse.jpg\"\r\n  ],\r\n
\"tags\": [\r\n    {\r\n      \"id\": 826,\r\n      \"name\":
\"Plusha33\"\r\n    }\r\n  ],\r\n  \"status\": \"available\"\r\n}",
    "options": {
      "raw": {
        "language": "json"
      }
    }
  },
  "url": {
    "raw":
"https://petstore.swagger.io/v2/pet",
    "protocol": "https",
    "host": [
      "petstore",
      "swagger",
      "io"
    ],
    "path": [
      "v2",
      "pet"
    ]
  }
},
"response": []
},
{
  "name": "DelPet",
  "event": [
    {
      "listen": "test",
      "script": {
        "exec": [
          "pm.test(\"Status code
is 200\", function () {\r",
          "
pm.response.to.have.status(200, 404);\r",
          "});\r",
          "\r",

```

```

        "pm.test(\"Response time
is less than 200ms\", function () {\r",
        "
pm.expect(pm.response.responseTime).to.be.below(1000);\r",
        "});\r",
        "\r",
        "pm.test(\"Status code
"
name has string\", function () {\r",
        "
pm.response.to.have.status(\"OK\");\r",
        "
        "});\r",
        "\r",
        "\r",
        "\r",
        "pm.test(\"Body is
correct\", function () {\r",
        "
pm.response.to.have.body({\r",
        "
        "\code\": 200,\r",
        "\type\":
        "\message\":
        "});\r",
        "});\r",
        "\r",
        "\r",
        ""
    ],
    "type": "text/javascript"
}
}
],
"request": {
    "method": "DELETE",
    "header": [],
    "url": {
        "raw":
"https://petstore.swagger.io/v2/pet/826",
        "protocol": "https",
        "host": [
            "petstore",
            "swagger",
            "io"
        ],
        "path": [
            "v2",
            "pet",
            "826"
        ]
    }
},
"response": []
}
]
}

```

Запуск коллекции PetApi через cmd (командную строку)

```
^C
C:\Users\lacer>newman run https://www.getpostman.com/collections/c724f283358752d59a5
newman

PetApi3004

→ CreatePet
POST https://petstore.swagger.io/v2/pet [200 OK, 597B, 611ms]
✓ Status code is 200
✓ Successful POST request
✓ Response time is less than 200ms
✓ Status code name has string

→ IdPet
GET https://petstore.swagger.io/v2/pet/826 [200 OK, 597B, 144ms]
✓ Status code is 200
✓ Response time is less than 200ms
✓ Status code name has string
✓ Content-Type is present

→ PutPet
PUT https://petstore.swagger.io/v2/pet [200 OK, 603B, 148ms]
✓ Status code is 200
✓ Response time is less than 200ms
✓ Status code name has string
✓ Content-Type is present

→ DelPet
DELETE https://petstore.swagger.io/v2/pet/826 [200 OK, 371B, 134ms]
✓ Status code is 200
✓ Response time is less than 200ms
✓ Status code name has string
✓ Body is correct
```

	executed	failed
iterations	1	0
requests	4	0
test-scripts	4	0
prerequest-scripts	0	0
assertions	16	0

```
newman run https://www.getpostman.com/collections/c724f283358752d59a5
```

→ CreatePet

```
POST https://petstore.swagger.io/v2/pet [200 OK, 597B, 642ms]
✓ Status code is 200
✓ Successful POST request
✓ Response time is less than 200ms
✓ Status code name has string
```

→ IdPet

```
GET https://petstore.swagger.io/v2/pet/826 [200 OK, 597B, 142ms]
✓ Status code is 200
✓ Response time is less than 200ms
✓ Status code name has string
✓ Content-Type is present
```

→ PutPet

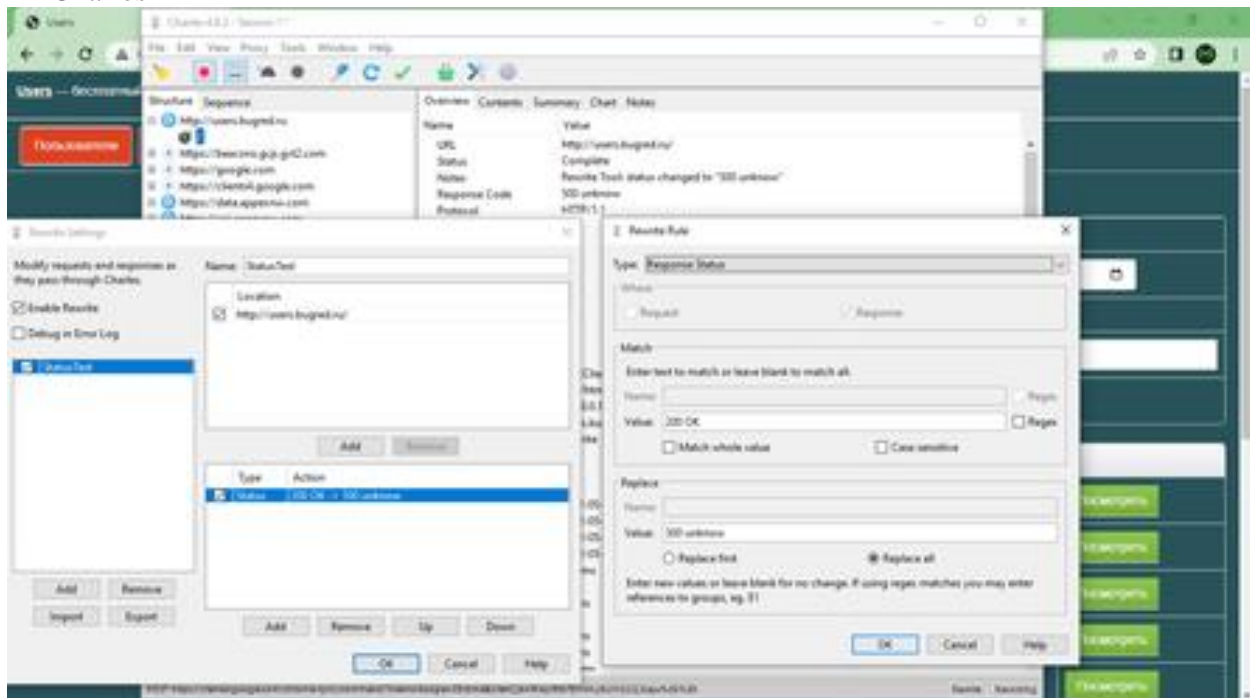
```
PUT https://petstore.swagger.io/v2/pet [200 OK, 603B, 141ms]
✓ Status code is 200
✓ Response time is less than 200ms
✓ Status code name has string
✓ Content-Type is present
```

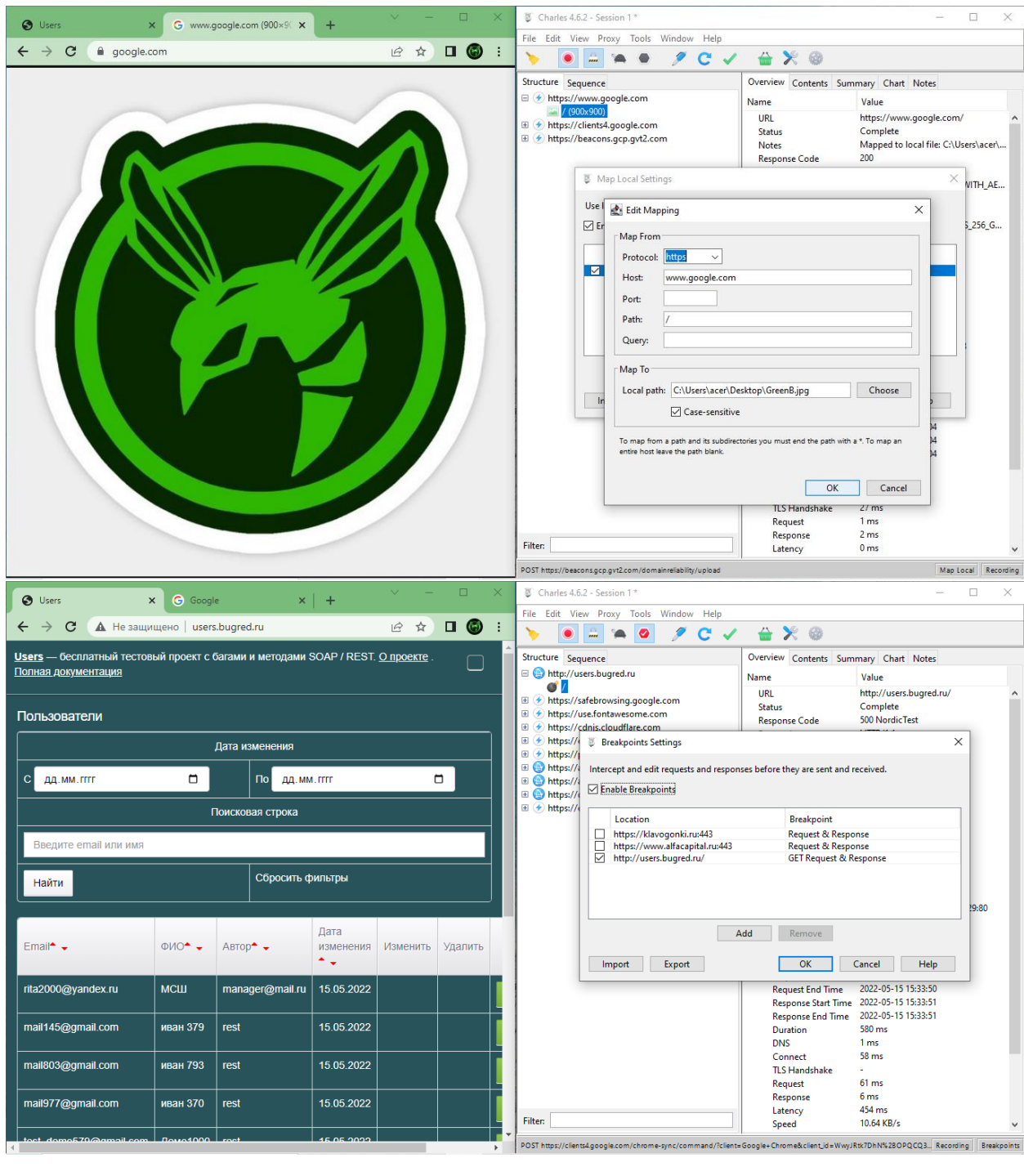
→ DelPet

```
DELETE https://petstore.swagger.io/v2/pet/826 [200 OK, 371B, 138ms]
✓ Status code is 200
✓ Response time is less than 200ms
✓ Status code name has string
✓ Body is correct
```


	executed	failed
iterations	1	0
requests	4	0
test-scripts	4	0
prerequisite-scripts	0	0
assertions	16	0
total run duration: 1528ms		
total data received: 864B (approx)		
average response time: 265ms [min: 138ms, max: 642ms, s.d.: 217ms]		

Charles





✓ Команды bash

a) `cd ; cd /mnt/c/Users/acer/test1`
Домашняя директория... путь... папка test1

b) `mkdir HomeWork`
Создание папки HomeWork

c) `cd HomeWork`

Переход в папку HomeWork

d) touch impression.txt
создание файла impression.txt

e) ls ; vim impression.txt ; i
просмотр содержимого; открытие вим; внесение изменений

f) 'Esc' ; :wq
режим команд в виме; выход с сохранением данных

g) cat impression.txt | grep NordicIT
просмотр содержимого файла и поиск по слову NordicIT

h) cat impression.txt | grep NordicIT >> Timp
просмотр, поиск строки по слову, перенаправление вывода+создание нового файла
Timp

i) ls
просмотр содержимого

j) ls -a
просмотр скрытого содержимого

k) pwd
путь

l) cp ./Timp /mnt/c/Users/acer/test1/MyResume
копирование папки Timp из HomeWork в MyResume