

PROJECT IMPLEMENTATION NOTES

Configuration [according to application.yml]

'bks' is specified as a working database (specified as spring.datasource.url). Name and create it according your needs.

Values of spring.datasource.username and spring.datasource.password properties must be replaced with actual ones.

Specify as well properties needed to communicate with the message broker - spring.rabbitmq.host, spring.rabbitmq.username, spring.rabbitmq.password

The application log is located at /log/bks.log in the project root according to logging.file.name property

Except for the database, everything needed is created automatically when the application starts:

- tables (and sequence) according to Liquibase scripts
- exchange and queue via RabbitClient class according to app.client_exchange and app.client_queue properties in application.properties

I have renamed the User table to Client to avoid possible conflicts with the reserved SQL.

In the following description, I'm going to mention specifically clients.

Sure, it is necessary to store client passwords in the database in encrypted form, but since this is not explicitly indicated in the task, then ...

Functioning

To demonstrate a new client creation on *RabbitMQ* message receipt DTO of a new test client is formed on the application start, it is serialized, published to Rabbit MQ exchange, deserialized while been read from the queue and fires service method to register a new client into the database (check run methof of the *BksCrmApplication* class).

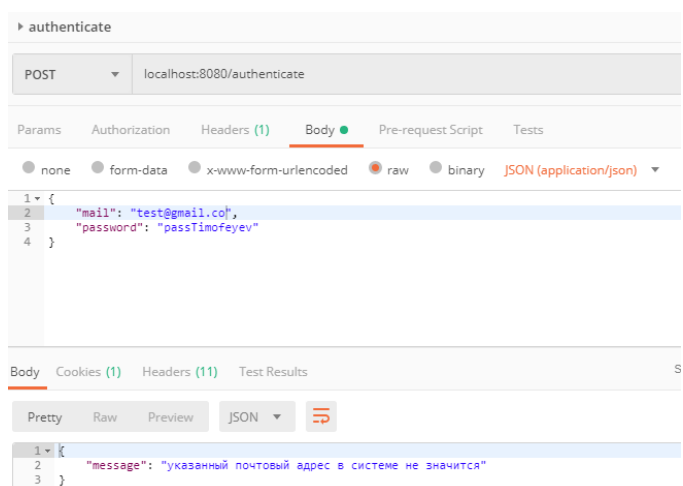
```
2023-03-27 01:38:30.227 INFO 11828 --- [pool-2-thread-4] com.bks.BksCrmApplication : зарегистрирован новый клиент: ClientDto(id=null, name=Andy, birthdate=2005-11-07, password=new_client_pass)
```

The result of the periodic increment of the balance of each client (until reaching a level of 207% of the initial value) is displayed in the log rounded to two decimal places

```
2023-03-27 00:15:44.701 INFO 8952 --- [BKS-1] com.bks.service.AccountServiceImpl : client ID - 1, current balance - 51.23, changed at - 2023/03/27 00:15:44
2023-03-27 00:15:44.702 INFO 8952 --- [BKS-1] com.bks.service.AccountServiceImpl : client ID - 2, current balance - 59.98, changed at - 2023/03/27 00:15:44
2023-03-27 00:15:44.702 INFO 8952 --- [BKS-1] com.bks.service.AccountServiceImpl : client ID - 3, current balance - 140.37, changed at - 2023/03/27 00:15:44
2023-03-27 00:16:13.998 INFO 8952 --- [BKS-2] com.bks.service.AccountServiceImpl : client ID - 1, current balance - 56.35, changed at - 2023/03/27 00:16:13
2023-03-27 00:16:13.999 INFO 8952 --- [BKS-2] com.bks.service.AccountServiceImpl : client ID - 2, current balance - 65.98, changed at - 2023/03/27 00:16:13
2023-03-27 00:16:13.999 INFO 8952 --- [BKS-2] com.bks.service.AccountServiceImpl : client ID - 3, current balance - 154.41, changed at - 2023/03/27 00:16:13
```

The application is managed by Spring Security, so one needs to log in to access REST points and Swagger. From the proposed options, the authentication using the mail + password combination was chosen.

Application generates custom exceptions, so in case of an erroneous mail + password combination you are greeted, for example, with the following message ('email is not registered in the system'):



authenticate

POST localhost:8080/authenticate Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```

1 {
2   "mail": "test@gmail.com",
3   "password": "passTimofeyev"
4 }

```

client				email_data			
id	date_of_birth	name	password	id	mail	client_id	
1	2000-01-01	Ivan Ivanov	passIvanov	1	test@gmail.com	1	
2	2000-02-02	Pavel Pavlov	passPavlov	2	test@yahoo.com	2	
3	2000-03-03	Timofey Timofeyev	passTimofeyev	3	test@gmail.com	3	
4	2005-11-07	Andy	new_client_pass				

Body Cookies (1) Headers (11) Test Results Status: 200 OK Time: 105 ms Size: 499 B Save Download

Pretty Raw Preview JSON

```

1 {
2   "jwt": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJ0ZXN0QGdtYWlsZmVibSIsImV4cCI6MTY3OTkwNjg0MiwiaWF0IjoxNjc5ODcwODQxLWlkcGVyY4siqCm"
3 }

```

get client by phone

GET

localhost:8080/api/client/getbyphone/:phone

Send

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

Cookies

Code

KEY	VALUE	DESCRIPTION
phone	79207865434	Description
KEY	VALUE	DESCRIPTION
Key	Value	Description

Body

Cookies (1)

Headers (10)

Test Results

Status: 403 Forbidden

Time: 37 ms

Size: 312 B

Pretty

Raw

Preview

Auto

id

1

api/client/getbyphone/{phone}

api/client/getbybirthdate ('younger than' selection)

get client by birthdate

GET localhost:8080/api/client/getbybirthdate?birthdate=02.02.2000&page=0&size=2

Params Authorization Headers (1) Body Pre-request Script Tests

KEY	VALUE	DESCRIPTION
Authorization	Bearer eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJ0ZXN0QGdtYWlsL...	
Key	Value	Description

Body Cookies (1) Headers (11) Test Results Status: 200 OK Time: 101 ms Size: 824

Pretty Raw Preview JSON

```

1 {
2   "content": [
3     {
4       "id": 3,
5       "name": "Timofey Timofeyev",
6       "birthdate": "2000-03-03",
7       "password": "passTimofeyev"
8     },
9     {
10      "id": 4,
11      "name": "Andy",
12      "birthdate": "2005-11-07",
13      "password": "new_client_pass"
14    }
15  ],
16  "pageable": {
17    "sort": {
18      "empty": true,
19      "sorted": false,
20      "unsorted": true
21    },
22    "offset": 0,
23    "pageNumber": 0,
24    "pageSize": 2,
25    "paged": true,
26    "unpaged": false
27  },
28  "last": true,
29  "totalPages": 1,
30  "totalElements": 2,
31  "number": 0,
32  "size": 2,
33  "sort": {
34    "empty": true,
35    "sorted": false,
36    "unsorted": true
37  },
38  "first": true,
39  "numberOfElements": 2,
40  "empty": false
41 }

```

- api/client/registerclient; new client registration

register client

POST localhost:8080/api/client/registerclient Send

Params Authorization Headers (2) Body Pre-request Script Tests Cookies Code

none form-data x-www-form-urlencoded raw binary JSON (application/json)

```

1 {
2   "name": "Notan",
3   "password": "passNotanoww4",
4   "birthdate": "06.06.2020"
5 }

```

Body Cookies (1) Headers (11) Test Results Status: 201 Created Time: 87 ms Size: 440 B Save

Pretty Raw Preview JSON

```

1 {
2   "id": 5,
3   "name": "Notan",
4   "birthdate": "2020-06-06",
5   "password": "passNotanoww4"
6 }

```

- api/client/moneytransfer; money transfer between client accounts

Critical messages during money transfer between accounts are generated in the following cases:

- wrong sender id
- wrong recipient id
- insufficient funds at the sender account
- money amount is either infinite or NAN what raises NumberFormatException during BigDecimal conversion

for instance ('sender has not enough funds'):

money transfer

PUTlocalhost:8080/api/client/moneytransfer?sourceId=1&recipientId=2&amount=50.3Send

ParamsAuthorizationHeaders (1)BodyPre-request ScriptTestsCookiesCode

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> sourceId	1	
<input checked="" type="checkbox"/> recipientId	2	
<input checked="" type="checkbox"/> amount	50.3	
Key	Value	Description

BodyCookies (1)Headers (11)Test ResultsStatus: 404 Not FoundTime: 143 msSize: 432 BSave

PrettyRawPreviewJSON

```
1 {
2   "message": "у отправителя недостаточно средств"
3 }
```

JWT structure:

JWT

DebuggerLibrariesIntroductionAsk

Crafted by auth0

AlgorithmHS256

EncodedPASTE A TOKEN HERE

DecodedEDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

{ "alg": "HS256" }

PAYLOAD: DATA

{ "sub": "test@gmail.com", "exp": 1679908386, "iat": 1679872386 }

VERIFY SIGNATURE

HMACHA256(base64UrlEncode(header) + "." + base64UrlEncode(payload), your-256-bit-secret)

☐ secret base64 encoded

Signature Verified

SHARE JWT

Sorry for using email instead of USER_ID in the payload. Token is assigned a lifetime (10 hrs in our case).

Caching

Caching at the service level has been implemented for the 'find a client by phone number' request.

Caching is implemented in the following form: the request authenticated with JWT writes the found client into the cache. Cache has a lifetime (one minute in the example). The subsequent request, if it is executed within a minute, will not be executed (accordingly the log message from the service method will not be displayed) The request executed later than the lifetime will get executed, resets the previous cache and forms a new temporary value, equal to the current one returned by the service.

51	2023-03-31 01:34:31.405	INFO	7628	---	[BRS-2]	com.bks.service.AccountServiceImpl	: client ID - 3, current balance - 18126.80, changed at - 2023/03/31 01:34:31
52	2023-03-31 01:34:31.406	INFO	7628	---	[BRS-2]	com.bks.service.AccountServiceImpl	: client ID - 2, current balance - 19024.39, changed at - 2023/03/31 01:34:31
53	2023-03-31 01:34:31.407	INFO	7628	---	[BRS-2]	com.bks.service.AccountServiceImpl	: client ID - 1, current balance - 4511.14, changed at - 2023/03/31 01:34:31
54	2023-03-31 01:35:00.517	INFO	7628	---	[BRS-1]	com.bks.service.AccountServiceImpl	: client ID - 3, current balance - 19939.48, changed at - 2023/03/31 01:35:00
55	2023-03-31 01:35:00.518	INFO	7628	---	[BRS-1]	com.bks.service.AccountServiceImpl	: client ID - 2, current balance - 20926.83, changed at - 2023/03/31 01:35:00
56	2023-03-31 01:35:00.518	INFO	7628	---	[BRS-1]	com.bks.service.AccountServiceImpl	: client ID - 1, current balance - 4962.25, changed at - 2023/03/31 01:35:00
57	2023-03-31 01:35:04.570	INFO	7628	---	[Ehcache [default]-0]	c.bks.service.support.CacheEventLogger	: Зафиксировано событие кэша CREATED для телефона 79207865434. Прежнее значение null. Новое значение Client(id=3, name='Timofey Timofeyev', birthdate=2000-03-03)
58	2023-03-31 01:35:29.261	INFO	7628	---	[BRS-2]	com.bks.service.AccountServiceImpl	: client ID - 3, current balance - 21933.43, changed at - 2023/03/31 01:35:29
59	2023-03-31 01:35:29.261	INFO	7628	---	[BRS-2]	com.bks.service.AccountServiceImpl	: client ID - 2, current balance - 23019.51, changed at - 2023/03/31 01:35:29
60	2023-03-31 01:35:29.261	INFO	7628	---	[BRS-2]	com.bks.service.AccountServiceImpl	: client ID - 1, current balance - 5458.48, changed at - 2023/03/31 01:35:29
61	2023-03-31 01:35:58.811	INFO	7628	---	[BRS-1]	com.bks.service.AccountServiceImpl	: client ID - 3, current balance - 24126.77, changed at - 2023/03/31 01:35:58
62	2023-03-31 01:35:58.812	INFO	7628	---	[BRS-1]	com.bks.service.AccountServiceImpl	: client ID - 2, current balance - 25321.46, changed at - 2023/03/31 01:35:58
63	2023-03-31 01:35:58.812	INFO	7628	---	[BRS-1]	com.bks.service.AccountServiceImpl	: client ID - 1, current balance - 6004.33, changed at - 2023/03/31 01:35:58
64	2023-03-31 01:36:07.503	INFO	7628	---	[Ehcache [default]-1]	c.bks.service.support.CacheEventLogger	: Зафиксировано событие кэша EXPIRED для телефона 79207865434. Прежнее значение Client(id=3, name='Timofey Timofeyev', birthdate=2000-03-03). Новое значение null
65	2023-03-31 01:36:07.507	INFO	7628	---	[Ehcache [default]-1]	c.bks.service.support.CacheEventLogger	: Зафиксировано событие кэша CREATED для телефона 79207865434. Прежнее значение null. Новое значение Client(id=3, name='Timofey Timofeyev', birthdate=2000-03-03)
66	2023-03-31 01:37:14.660	INFO	7628	---	[Ehcache [default]-2]	c.bks.service.support.CacheEventLogger	: Зафиксировано событие кэша EXPIRED для телефона 79207865434. Прежнее значение Client(id=3, name='Timofey Timofeyev', birthdate=2000-03-03). Новое значение null
67							

Additionally project contains:

- REST requests collection along with the parameters to be imported into Postman. Do not forget to authenticate, copy JWT and fix Authorization headers accordingly
- log example

Project is located at

<https://github.com/hyperborean72/bks>