Technical Solution Description

eCare

Table of Contents

Table of Contents

1. Task

Part 1

The task is to develop an application that models the information system of the mobile operator.

Application must provide the following functionality:

For clients:
- View contracts
- Manage contracts
- Manage options

For managers:
- Create account for client
- Find clients by phone
- View all clients and tariffs
- Create and manage contracts
- Create and manage tariffs
- Create and manage options

Part 2

The task is to develop an application for advertising stand, showing tariffs from first application.

Requirements
- Applications should communicate through MQ

2. Used technologies

Technology	Version
ActiveMQ	5.15.8
MySQL	5.7
Spring	4.3.18.RELEASE
Spring-Security	4.2.5.RELEASE
ModelMapper	0.7.4
Java	8
Wildfly	14.0.1.Final
Hibernate	4.3.11.Final
JUnit	4.12
Log4j	1.2.17
Mockito	1.10.19
lombok	1.18.2
JSF	2.3
Maven	3.2.0
IntelliJ IDEA	2018.3
AspectJ	1.8.13
Jackson	2.4.1

3. Additional features

- Forgotten password recovery - First application can send a link (with a unique token) to the client by mail. Client can use this link to change password.

- Top up account - Clients can deposit money into their account using a credit card or Yandex.Money. First application receives notifications from Yandex, checks the transaction token and puts the data into the database.

- Speech recognition - frontend of the first application can recognize words and find the corresponding item in the menu.

- Continuous integration - Jenkins downloads sources from GitHub, builds and tests applications and deploy them to WildFly.
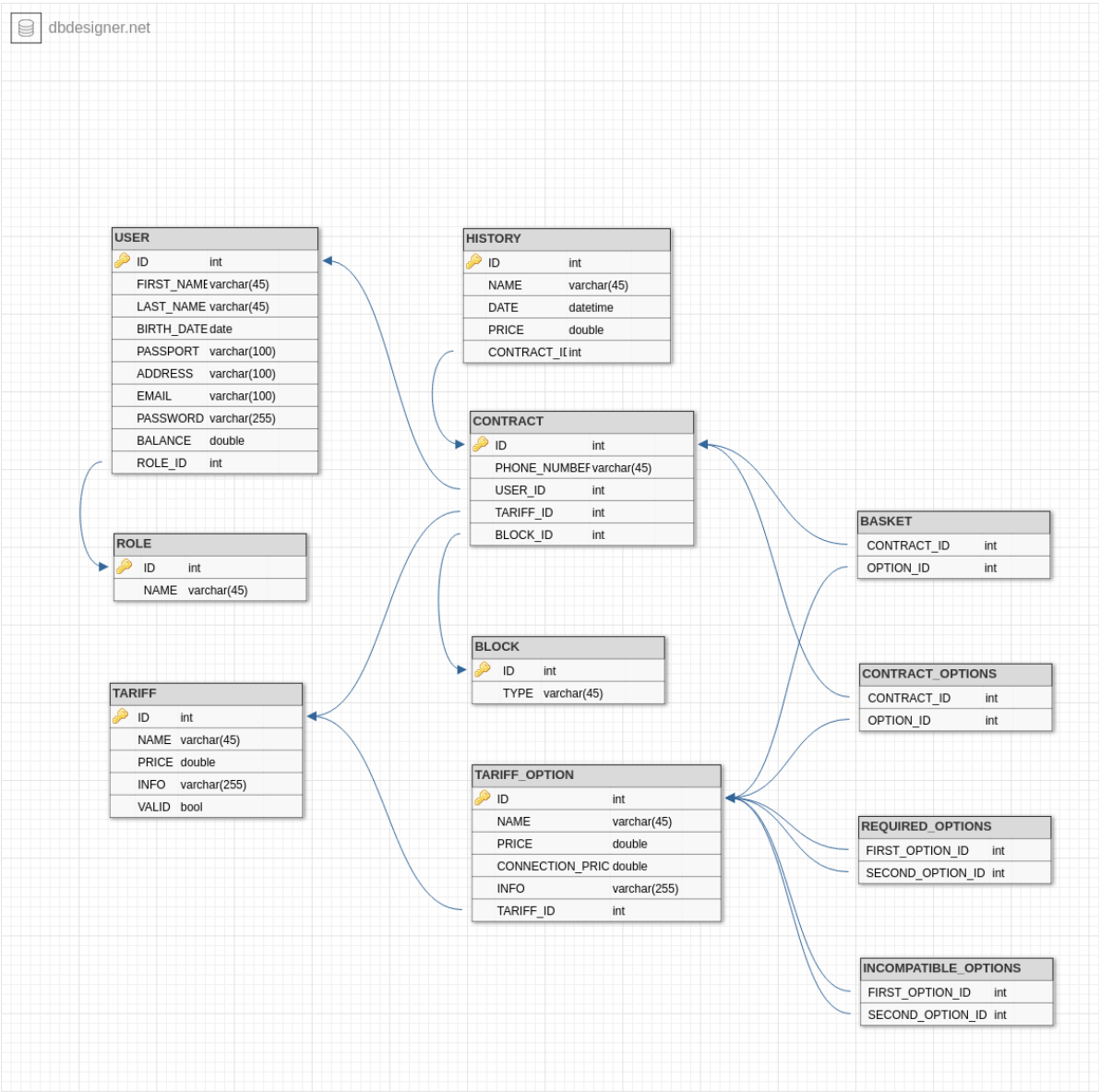
4. Database schema

Illustration 1: database

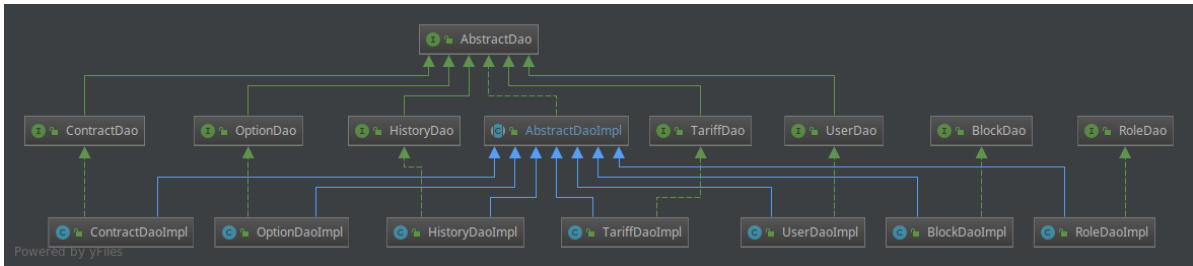# 5. Architecture

5.1 First application
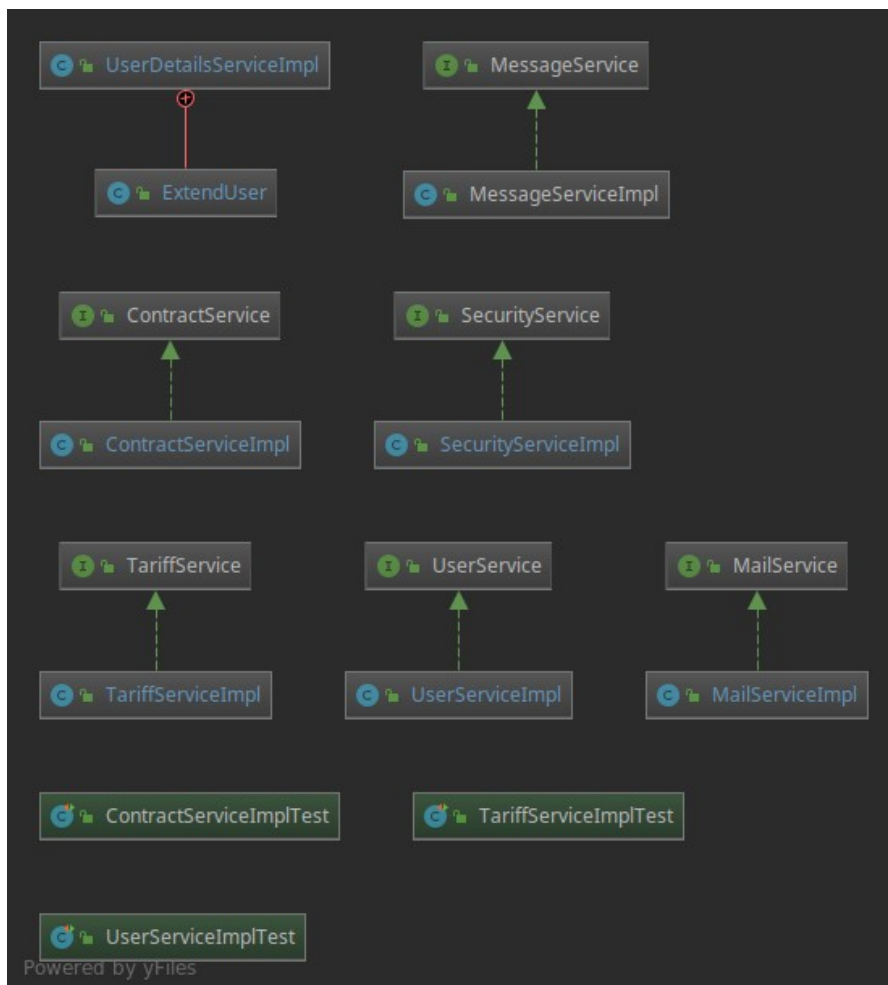
- Dao level

Illustration 2: Dao level

- Service level

Illustration 3: Service level

··T··Systems·····································
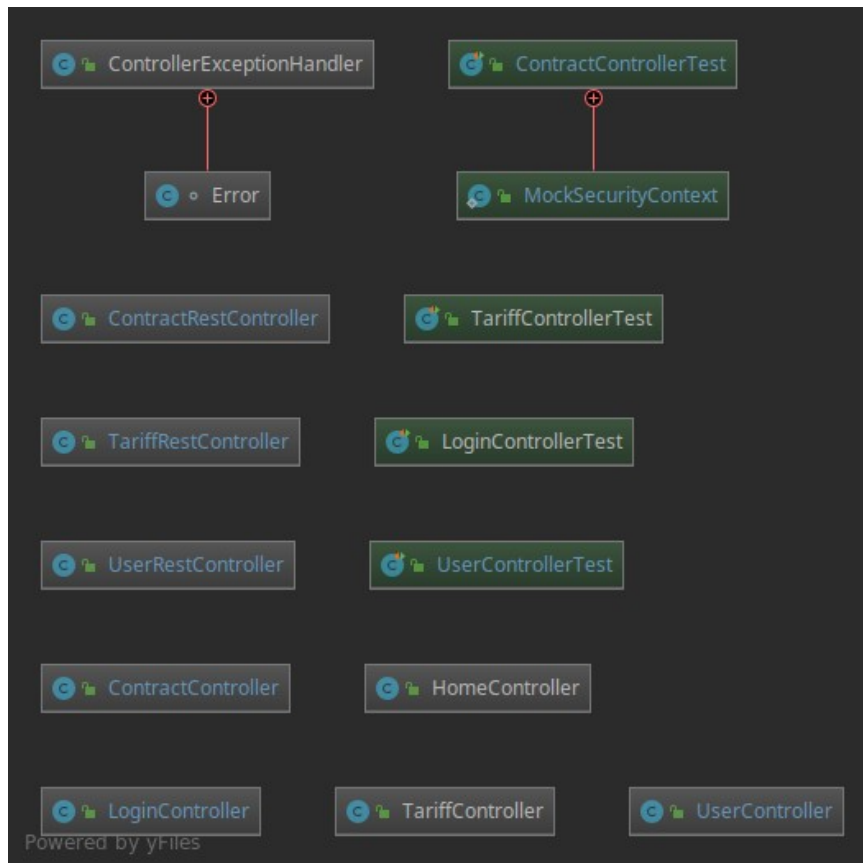
- Controllers level

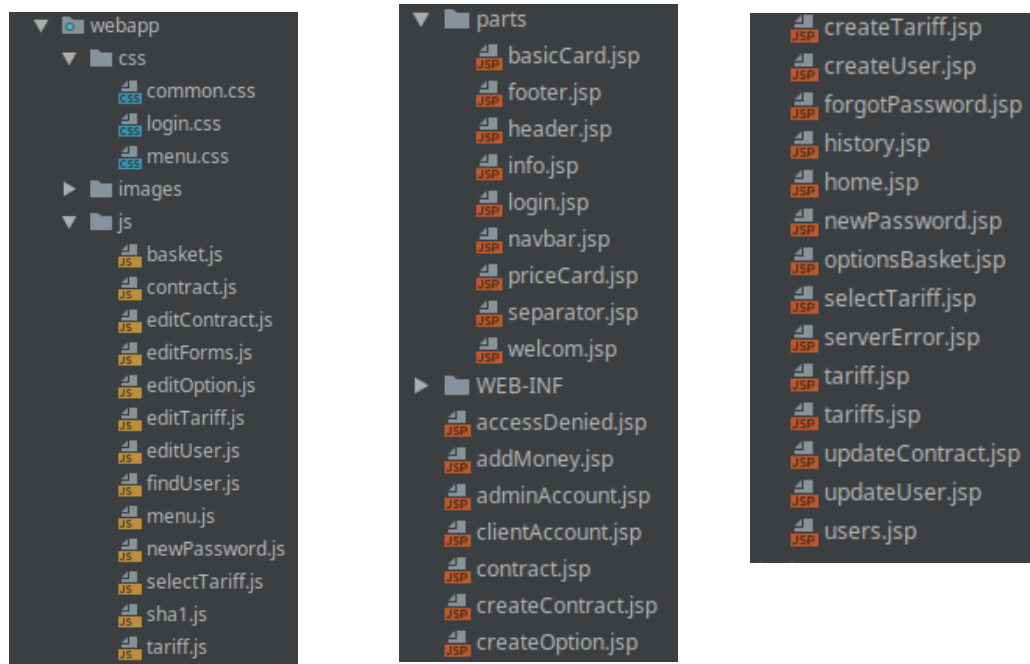*Illustration 4: Controllers level*

- View level

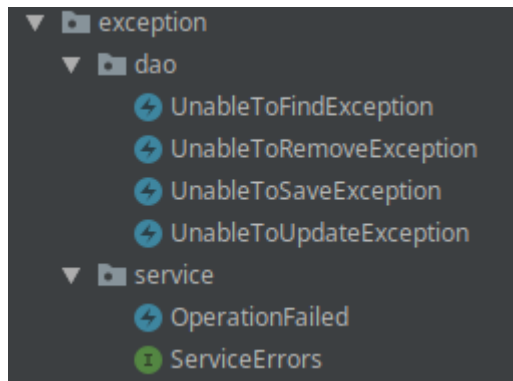*Illustration 5: View level*

- Exceptions level

Illustration 6: Exceptions level
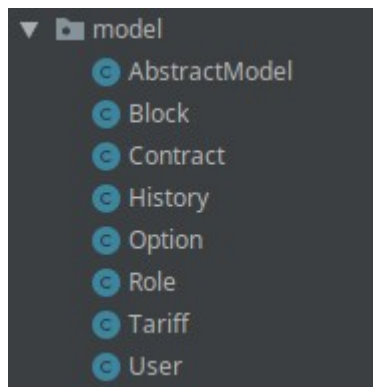
- Entity level

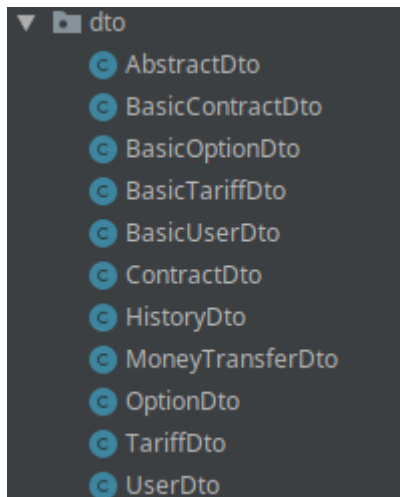*Illustration 7: Entity level*

- Dto level

Illustration 8: Dto level
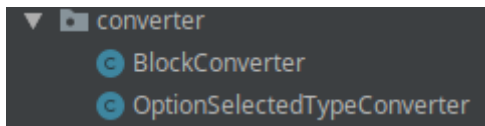
- Converters level

Illustration 9: Converters level

- For logging was used AOP

5.2 Second application

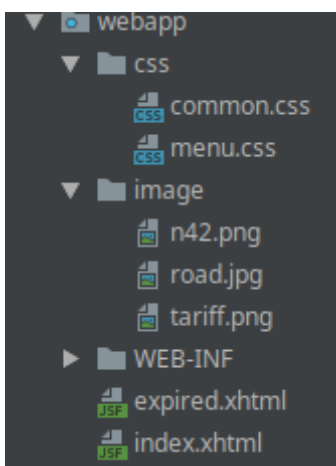Second application has next structure:

- View level

Illustration 10: View level
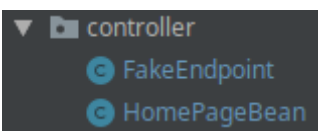
- Controllers level
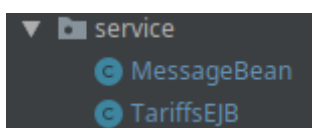

Illustration 11: Controllers level

- Service level

Illustration 12: Service level

6. Tests

JUnit tests:

```
        options0_.CONTRACT_ID=?
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.136 sec
Running com.bilev.service.impl.ContractServiceImplTest
Tests run: 65, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.436 sec
Running com.bilev.service.impl.TariffServiceImplTest
Tests run: 44, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.133 sec
Running com.bilev.service.impl.UserServiceImplTest
Tests run: 24, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.102 sec

Results :

Tests run: 150, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] Building ejb 1.0-SNAPSHOT
[INFO] ------------------------------------------------------------------------
[INFO]
```

Illustration 13: tests results

```
▼ ▩ test
  ▼ ▩ java
    ▼ ▩ com.bilev
      ▼ ▩ configuration
          ⓒ ServiceTestConfig
      ▼ ▩ controller
        ▼ ▩ jsp
            ⓒ ContractControllerTest
            ⓒ LoginControllerTest
            ⓒ TariffControllerTest
            ⓒ UserControllerTest
          ▩ rest
      ▼ ▩ service.impl
          ⓒ ContractServiceImplTest
          ⓒ TariffServiceImplTest
          ⓒ UserServiceImplTest
      ▼ ▩ tools
          ⓒ ContractCreator
          ⓘ Creator
          ⓒ OptionCreator
          ⓒ TariffCreator
          ⓒ UserCreator
  ▼ ▩ resources
        ⬛ 03-create-tables.sql
        ⬛ 04-insert-test-data.sql
        ⬛ 05-clear.sql
        ⬛ application.properties
```

Illustration 14: JUnit tests

··T··Systems········································

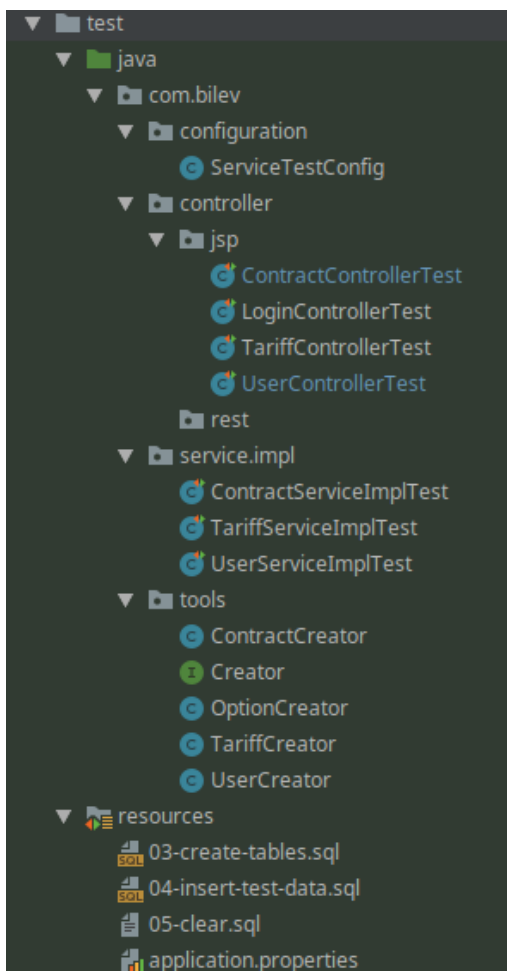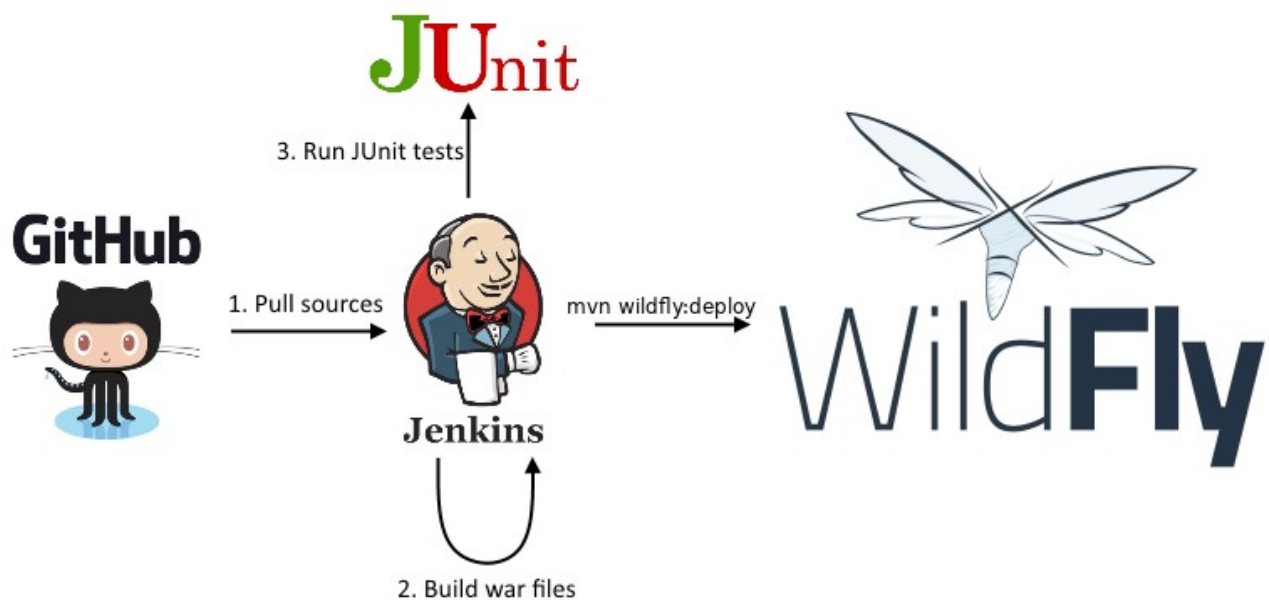## 7. Deployment

　　Project can be built with Jenkins, so all you need to build and deploy is commit changes to Github:

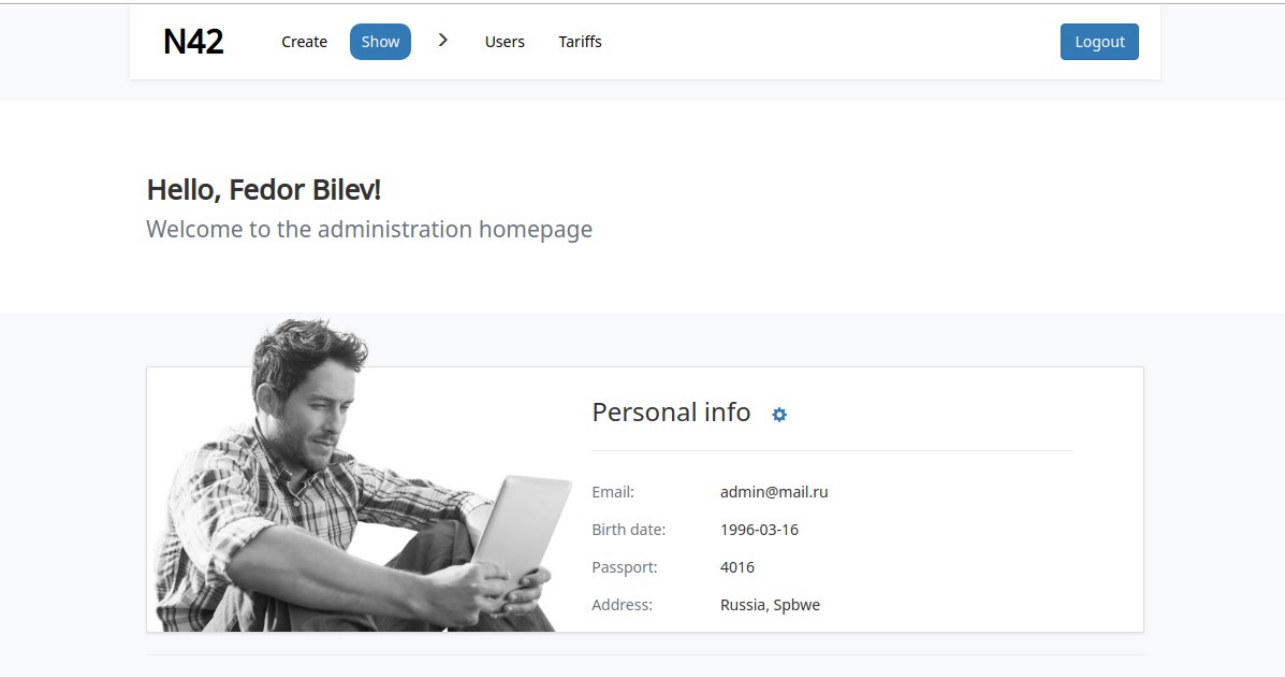Manual build

　　mvn clean install
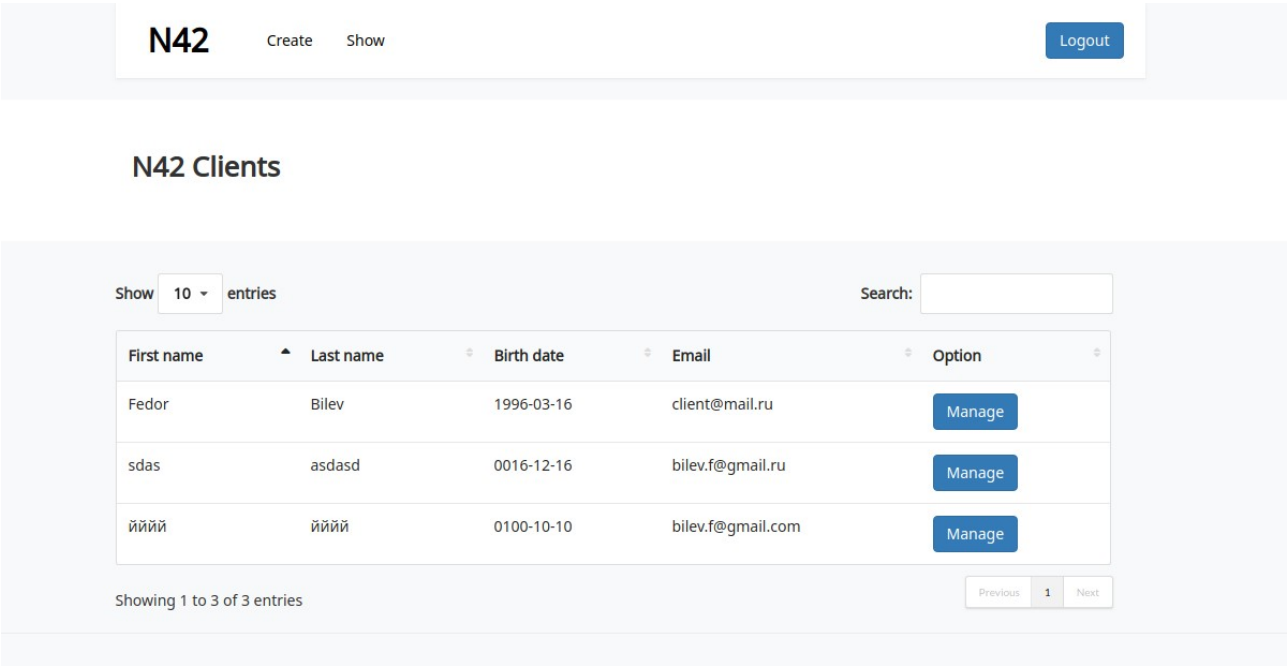　　mvn wildfly:deploy

8. GUI

Illustration 15: accoount

Illustration 16: list of clients

Illustration 17: list of tariffs

Illustration 18: tariff's home page

Option name

option

Option price

1

Option connection price

100.0

Connection price can not be blank.

Option info

info

Info must be between 2-255 characters long.

Name	Price	Connection price	Info	Option
22	2.0	2.0	22	Non INCOMPATIBLE REQUIRED
asdas	1.0	1.0	123	Non INCOMPATIBLE REQUIRED

Save option

Illustration 19: new option

Hello, Fedor Bilev!   Balance: ₽303.0

Welcome to the client's home page

Personal info

Email:	client@mail.ru
Birth date:	1996-03-16
Passport:	4016
Address:	Russia, Spbdd

Contracts

8(950)012-90-75

₽100.0 / mo

Tariff: Black
$5/Gb, $0.5/min, $0.5/sms
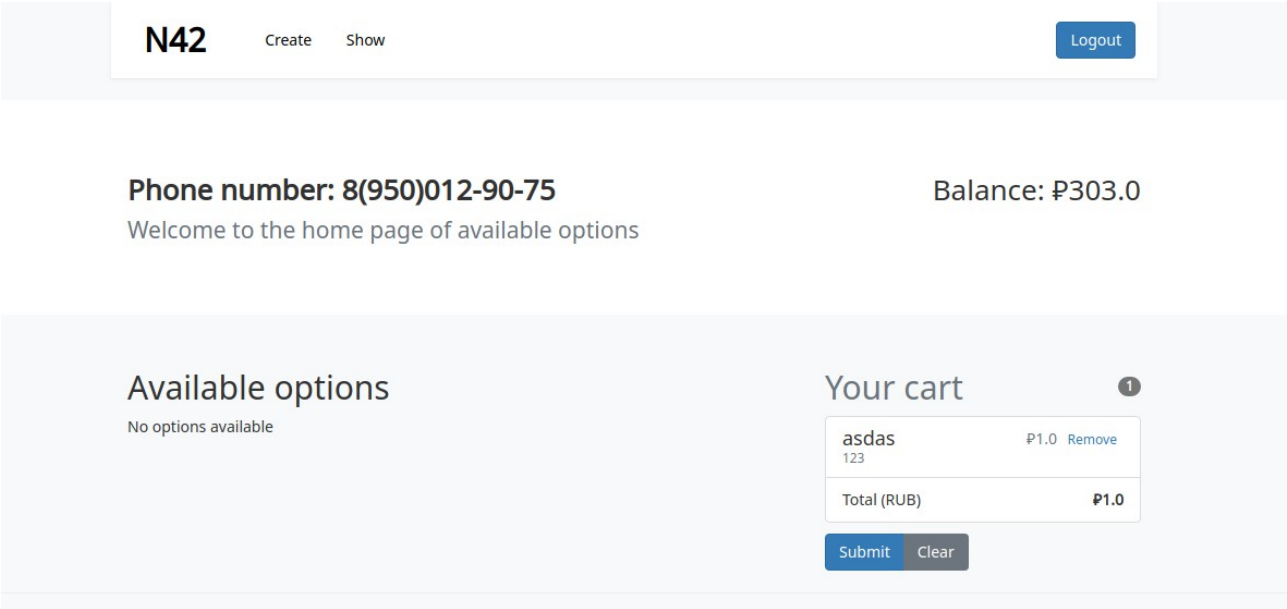
Manage

Illustration 20: client account

T··Systems

N42   Create   Show   Logout

Phone number: 8(950)012-90-75

Welcome to the home page of available options

Balance: ₽303.0

Available options

No options available

Your cart   ①

asdas 123	₽1.0	Remove
Total (RUB)		**₽1.0**

Submit   Clear

Illustration 21: basket

N42   Logout

Fedor Bilev

On this page you can add funds to your account
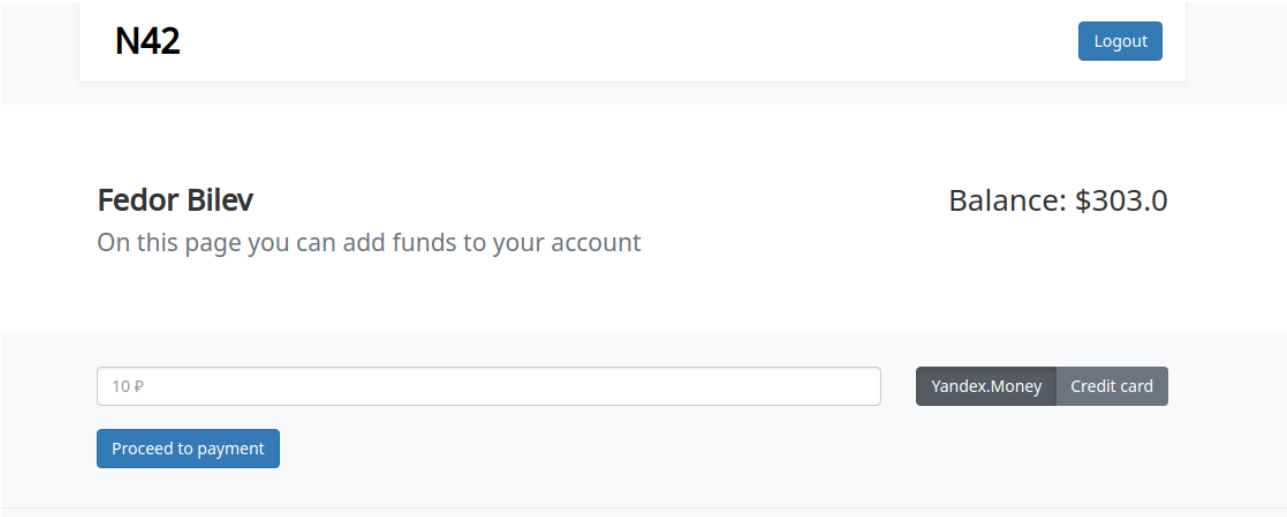
Balance: $303.0

| 10 ₽ | Yandex.Money | Credit card |

Proceed to payment

Illustration 22: add money