

Technical solution description

Mobile operator

Content

Technical solution description.....	1
Mobile operator	1
1 Task	3
2 Used technologies.....	4
3 Database schema	5
4 Architecture.....	6
4.1 Main application.....	6
4.2 Advertising stand.....	10
5 Tests and quality of code	11
6 Deployment.....	13
7 GUI.....	14

1 Task

Main goal was to develop an application that models information system of mobile operator.

Application must provide following functionality:

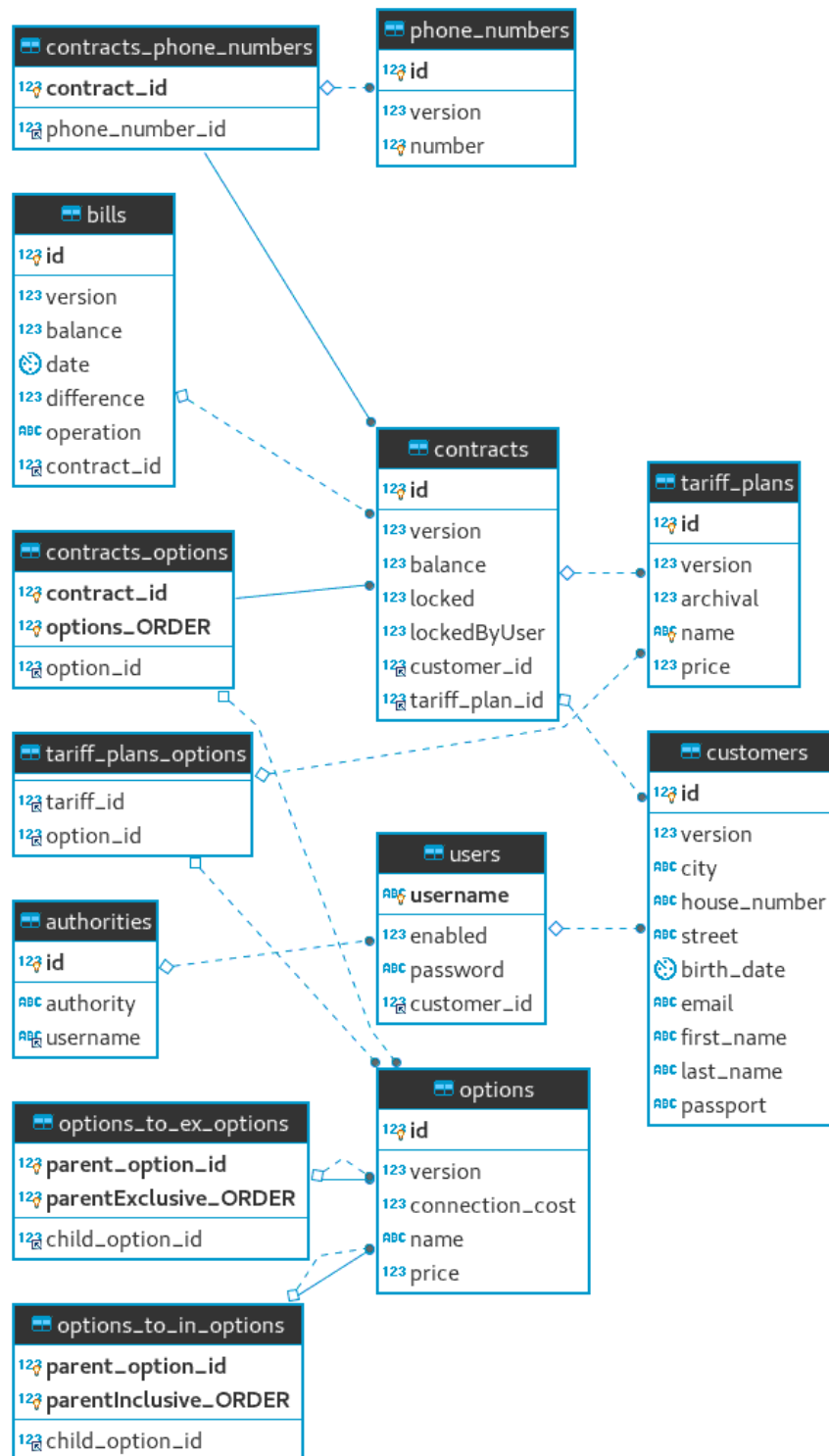
- For clients:
 - View contracts;
 - View all tariffs, change tariff;
 - View all tariff options, add new options.
 - Block / unblock contract.
- For managers:
 - Create new contracts with unique numbers/change old contracts;
 - View all clients and their contracts;
 - Block / unblock contracts;
 - Find client by number;
 - Manage tariffs and options.

Second goal was to develop an application for advertising stand, showing tariffs from first application. Applications should communicate by MQ.

2 Used technologies

Technology	Version
AspectJ	1.8.11
FasterXML Jackson	2.9.1
H2 Database	1.4.196
Hibernate	5.2.12
IntelliJ IDEA	2017.2.5
Java	8
JSP	2.2
JUnit	4.12
Log4J	1.2.17
Maven	3.5.0
MySQL	5.7
Primefaces	6.1
SonarQube	5.6.0
Spring and Spring security	5.1.0
Wildfly	14.0.1
Jenkins	2.150.1

3 Database schema



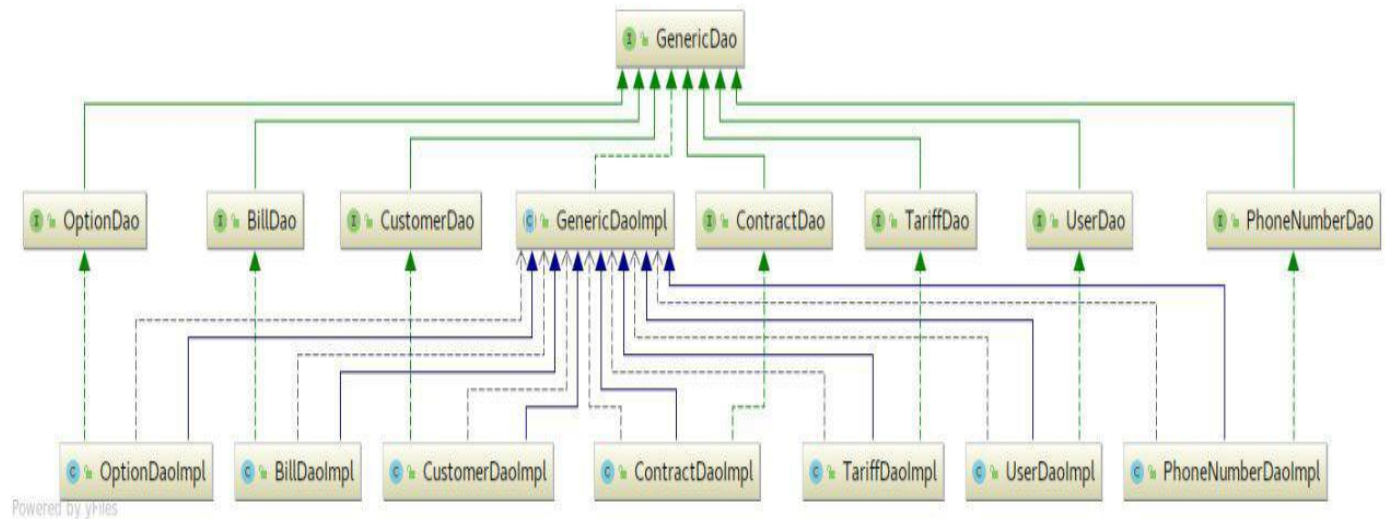
Picture 1 - database schema

4 Architecture

4.1 Main application

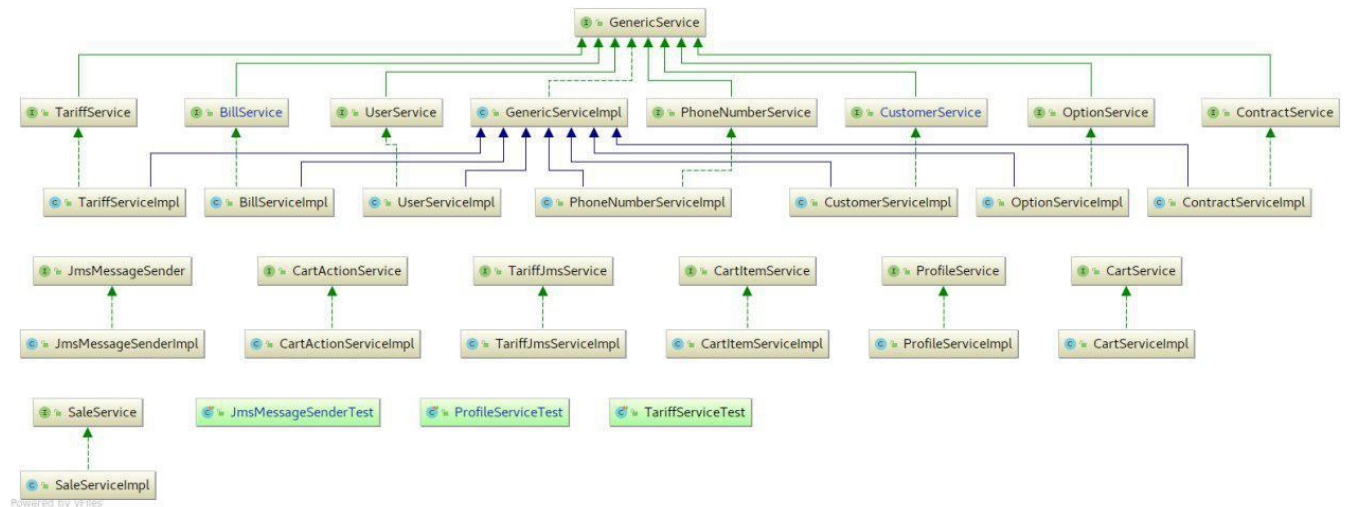
According to MVC, main application has next structure:

- Model level:



Picture 2 - model level diagram

- Controllers level:



Picture 3 - controllers level diagram

- View level:

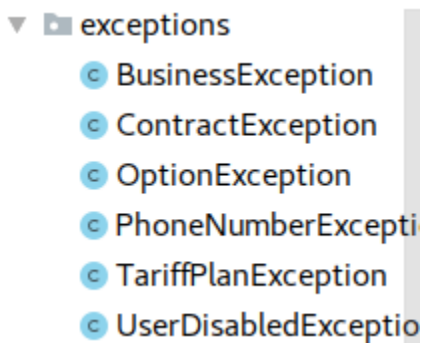


Picture 4 - server side view level



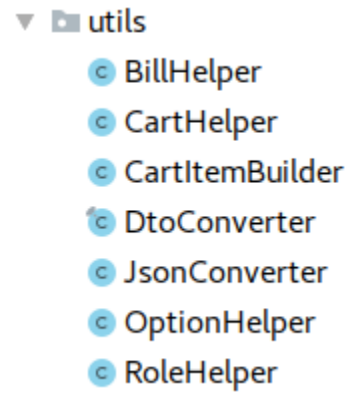
Picture 5 - client side view level

- Custom exceptions:



Picture 6 - custom exceptions list

- Utility classes:



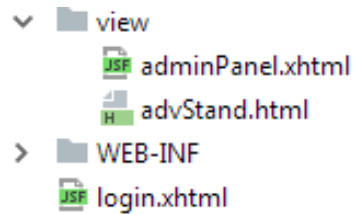
Picture 7 - utility classes

- For logging was used AOP.

4.2 Advertising stand

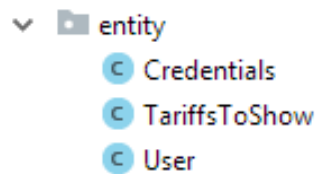
Advertising stand has next structure:

- Views:



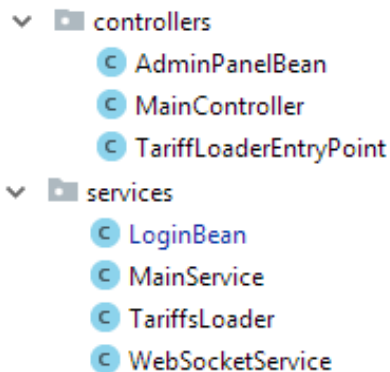
Picture 8 - advertising stand views

- Dtos:



Picture 9 - advertising stand entities

- Business logic:



Picture 10 - advertising stand views

5 Tests and quality of code

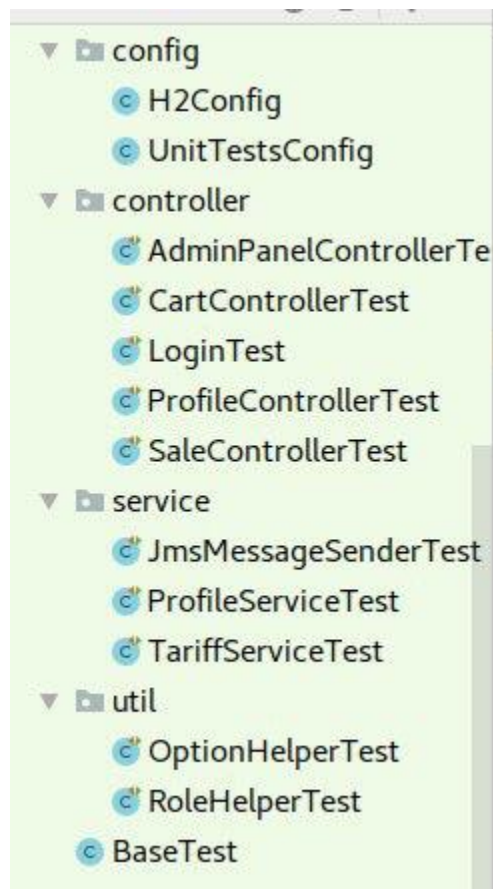
JUnit tests:

```
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.102 sec
```

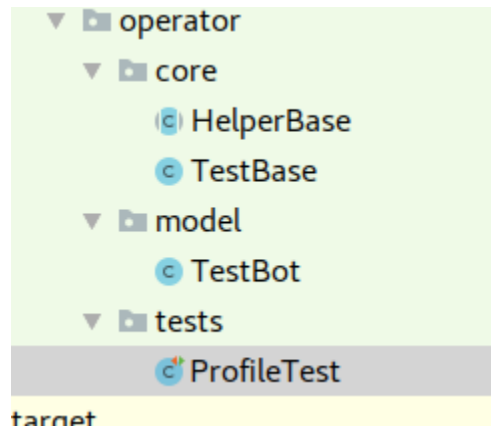
Results :

```
Tests run: 33, Failures: 0, Errors: 0, Skipped: 2
```

Picture 11 - JUnit tests result



Picture 12 - Unit tests structure



Picture 13 - UI tests structure

✓	ProfileTest (operator)	23 s 983 ms
✓	testAllContractsL	9 s 378 ms
✓	testLockButtons	3 s 630 ms
✓	testProfilePage	3 s 892 ms
✓	testInfoElement	3 s 767 ms
✓	testAdminPage	3 s 316 ms

14 – UI tests results

6 Deployment

Available deploy to wildfly via Jenkins.

Manual build and deploy with maven is also available:

1. In working directory enter in console: *mvn clean install*;
2. Then enter: *mvn wildfly:deploy*. (if wildfly is running)

7 GUI

MSS

Login

Username

Password

Login

MSS 2018

Picture 11 - login page

MSS Admin panel Sale contract Cart quantity: 0

Profile

Customers

Show 10 entries Search:

First Name	Last Name	Birth Date	Contract
asd	dsa	27.11.2018	79817549182
HH	Test	02.10.2018	79817549092
pam	pam	02.10.2018	79817549091 79817549181
Pam	a	02.10.2018	79817549094
User	Test	02.10.2018	79817549093

Picture 12 - managers index page

MSS Admin panel Sale contract Cart quantity: 0

Profile / 79817549091

79817549091

Confirm lock

Add to cart lock

Options: 1

Tariff plan:
Tariff sms
500

Balance:
-634

Option management

Change tariff plan

Current options

Show 10 entries

Search:

Name	Price
sms	0

Showing 1 to 1 of 1 entries

Previous
1
Next

79817549091

79817549181

Picture 17 - clients index page