METROINFORMATOR

Technical solution description

BYSTROV ALEXEY

30 October, 2018

CONTENT

1. TASK……... 3
2. PROJECT GOALS…… 4
3. ADDITIONAL FEATURES……... 5
4. INSTRUMENTS AND TECHNOLOGIES……... 6
5. DATABASE MODEL…… 7
6. SYSTEM INFRASTRUCTURE…… 8
7. SYSTEM ARCHITECTURE…… 9
8. USER INTERFACE EXAMPLES…… 14
9. CODE QUALITY…… 17
10. DEVELOPMENT WORKFLOW…….. 20
11. BUILD AND DEPLOY……... 25
12. FUTURE IMPROVEMENTS……. 28

1. TASK

Develop application that represents information system for private metro company. The system provides the ability to purchase electronic ticket on the train in different directions.

User cases:

- search train chains passing through station A to station B in a given date
- schedule of trains on station;
- ticket purchase:
  - if train has empty seats;
  - before the departure not less than 10 minutes;
  - the trip will take no more than one day

Administrator cases:

- adding new trains;
- opening/closing stations;
- deleting trains;
- viewing of all registered passenger train;
- viewing all trains;
- viewing status of all stations

Additionally, to develop co-application for railway scoreboard retrieving data from main application.

2. PROJECT GOALS

- The useful, reliable, fast and automated system

- Cohesive data model

- User-friendly interface

- Separate access to different system’s part

3. ADDITIONAL FEATURES

- Search time-optimized path between stations on different branches taking into account closed transition stations
- Buying multiply tickets if begin and end stations located on different branches
- Automated schedule creation for the week ahead (can be changed)
- Automated cleaning old schedules
- Automated ticket-invalidation ticket if begin or end station are closed and before departure or arrival less than 10 minutes
- Sending SMS-messages:
    - when user was registered successfully
    - when ticket was invalidated (closed station or deleted train)
- Selenium UI – tests
- For tests used mini database in Docker
- JProfiler used for performance optimization and CPU-load control
- Continuous Integration with Jenkins

4. INSTRUMENTS AND TECHNOLOGIES

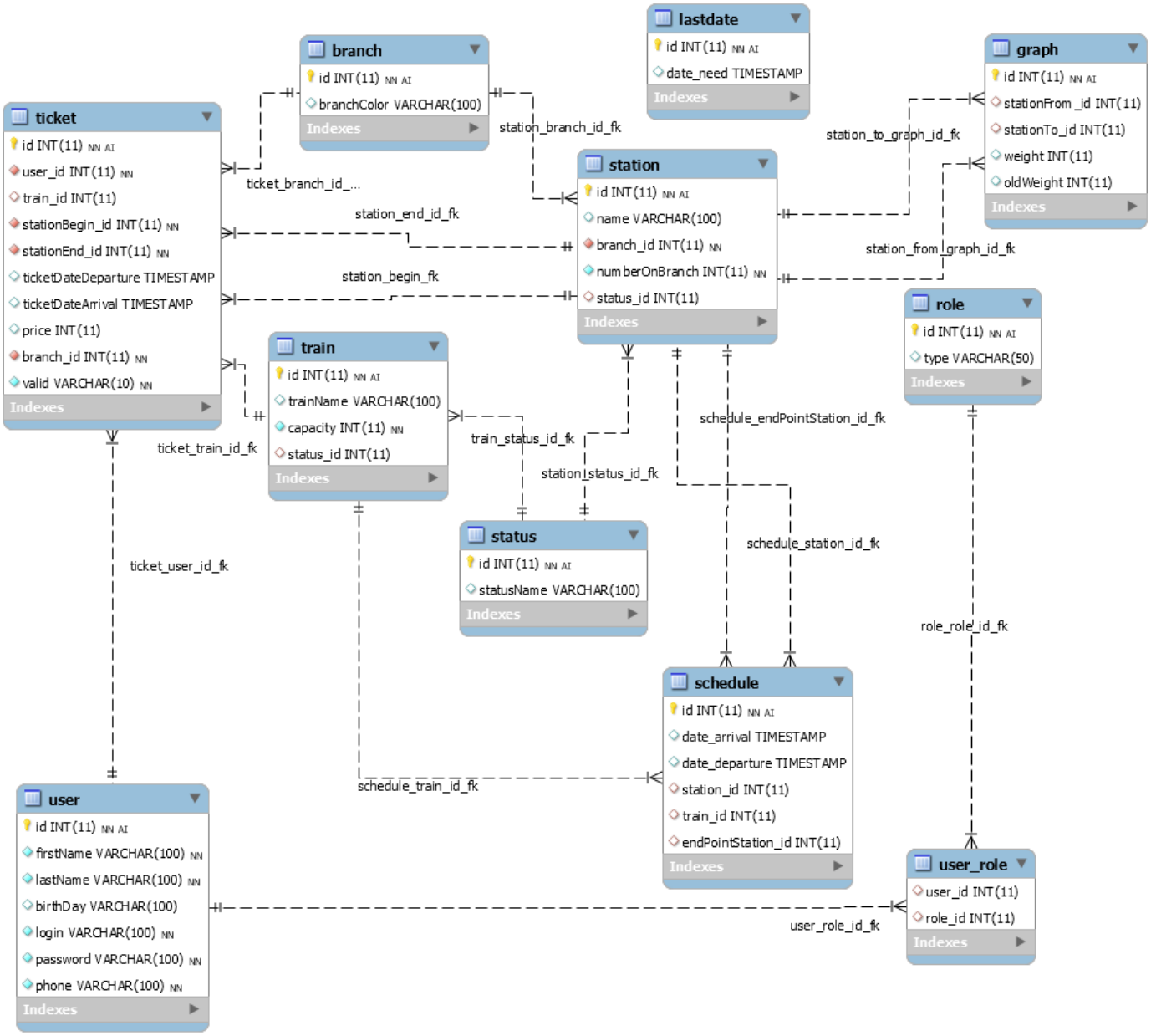
INSTRUMENTS

- IDE – Intelliji IDEA
- DB – MySQL 8.0.12
- Docker 18.06.1-ce-win73 (19507)
- Jenkins 2.146
- JProfiler 10.1.3
- MAINSMS
- Maven 3.5.4
- Tomcat 8.5.33
- Wildfly 14.0.1.Final

TECHNOLOGIES

- Ajax
- Bootstrap 4.0
- Hibernate 5.3.1.Final
- Java 8
- JavaScript
- JPA 2.0
- JQuery
- JSF 2.1.7
- JSP 2.3.1
- JUnit
- Log4j 1.2.17
- Lombok 1.16.20
- Mockito 1.10.19
- Primefaces 6.1
- RabbitMQ 3.7.8
- REST
- Selenium 2.53.1
- SonarQube 6.7.5
- Spring Framework 5.0.8
- Spring Security 5.0.6

5. DATABASE MODEL



6. SYSTEM INFRASTRUCTURE

Front-end (browser presentation level):

1) Web-page structure - HTML

2) Page-design - CSS

3) Dynamic content - JavaScript, JQuery, Ajax.

Back-end (server based level):

1) Application server - Tomcat

2) Database - MySQL

3) Server logic - Spring Framework

Client advertisement application:

1) Web-pages - JSF

2) JMS - RabbitMQ

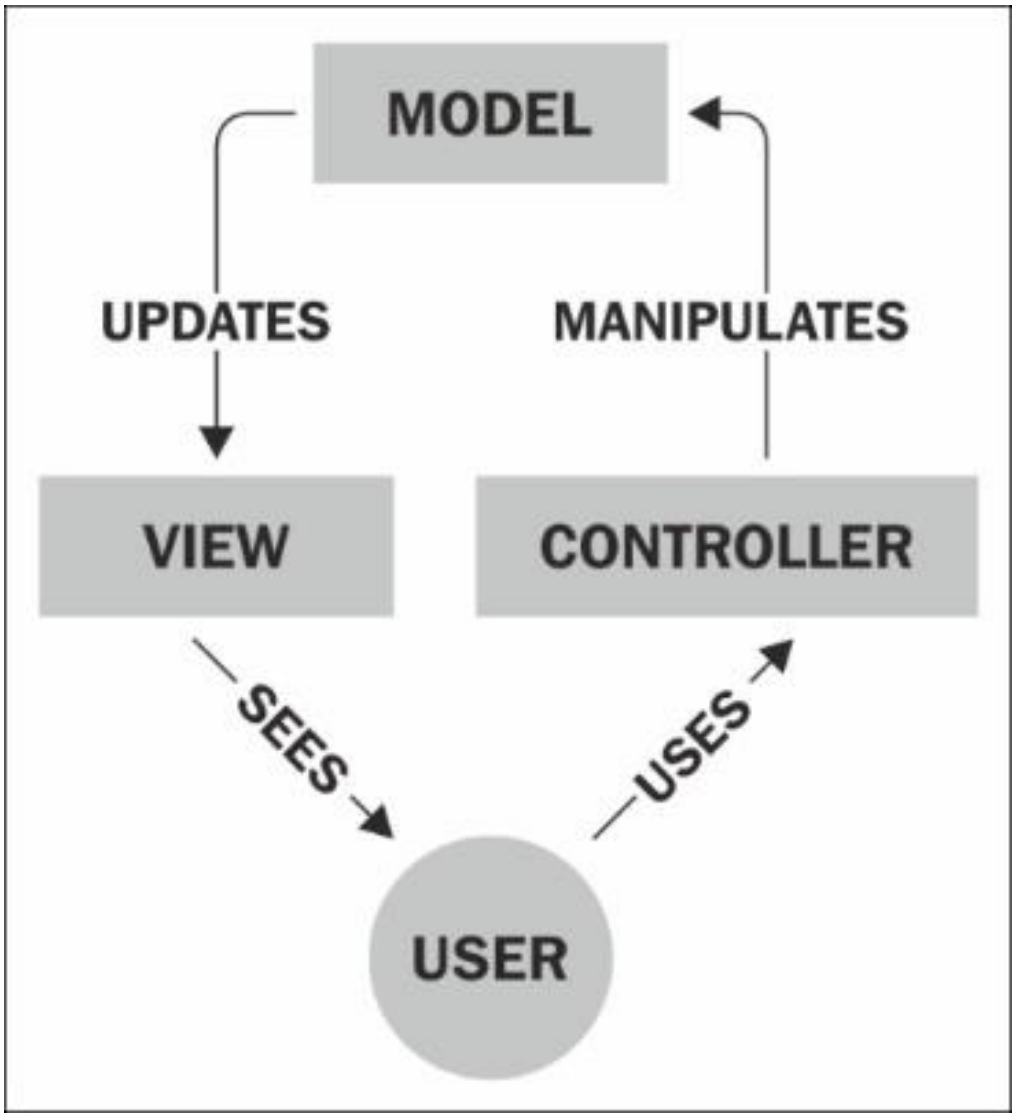
3) Application server - WildFly

4) Server logic - EJB

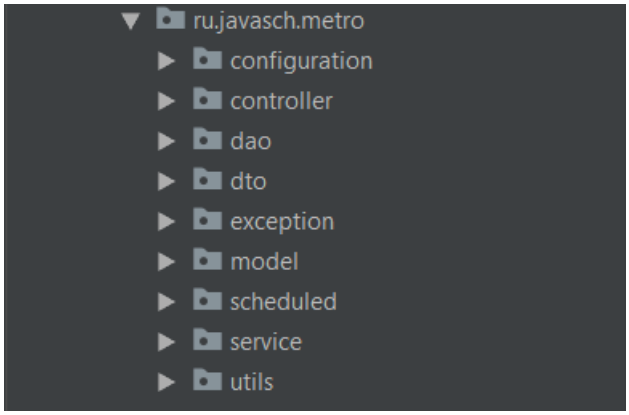
5) WS - REST

7. SYSTEM ARCHITECTURE

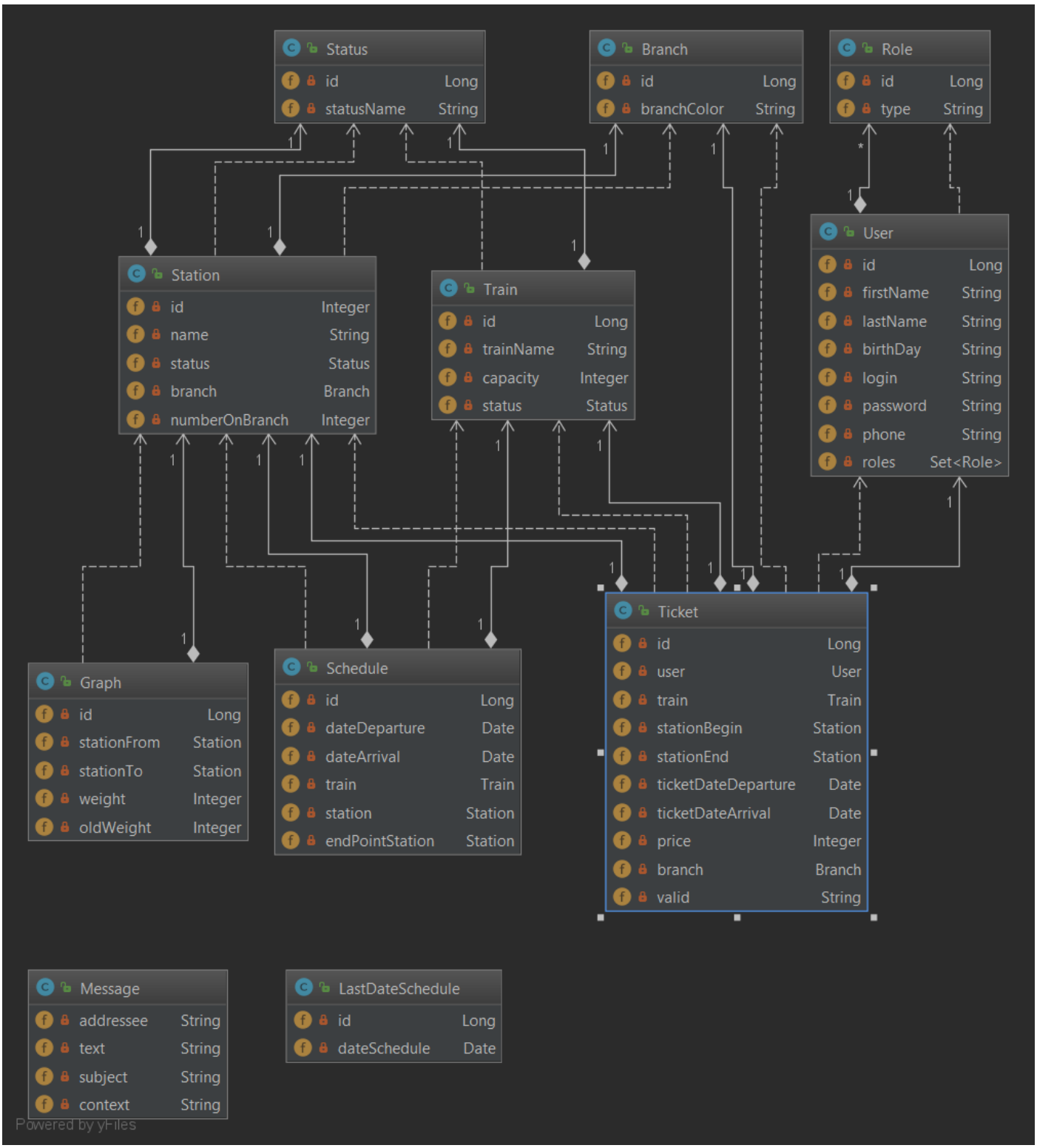
Architecture of server-based part presented by MVC - design pattern.



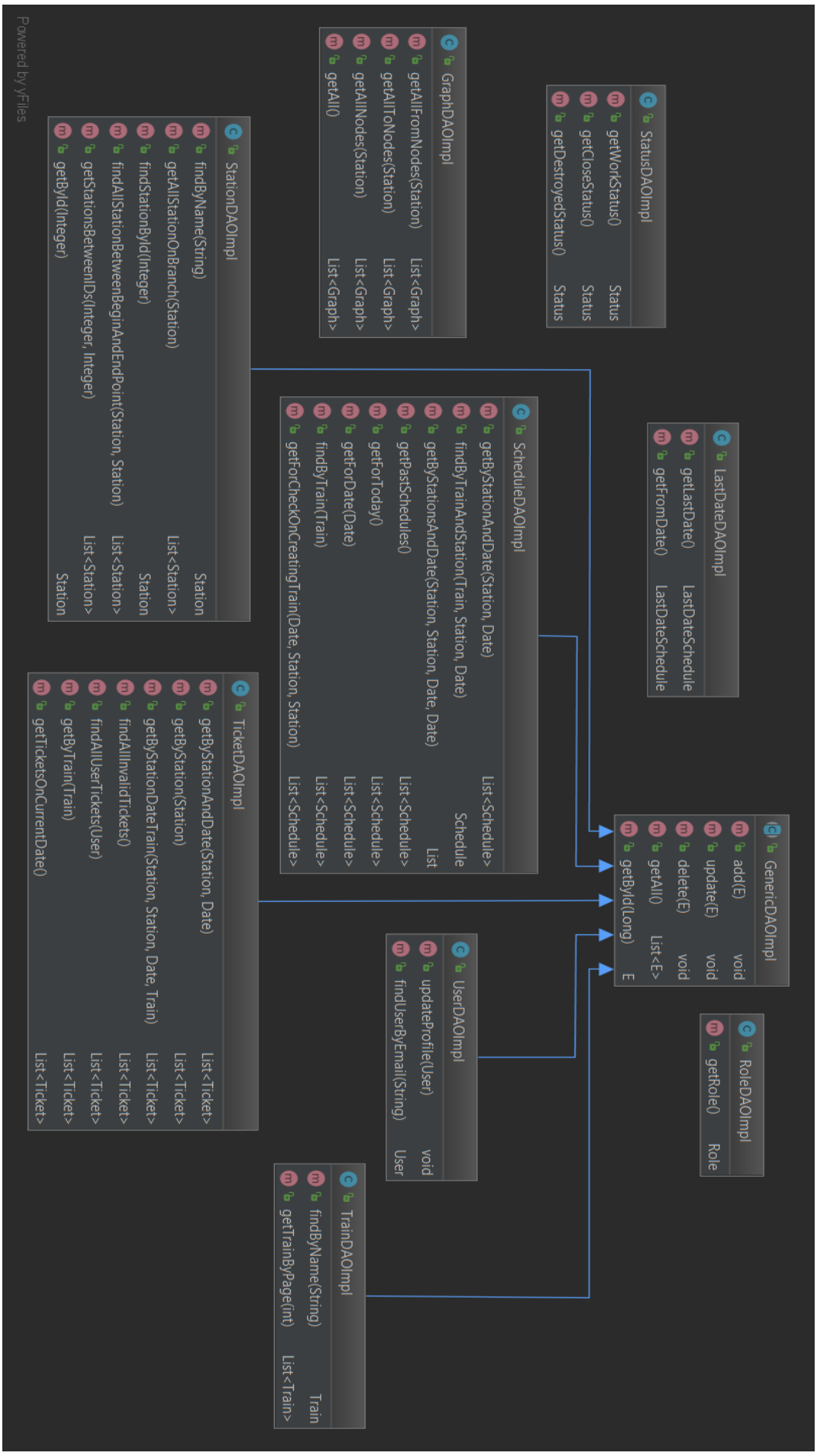
Class Structure:



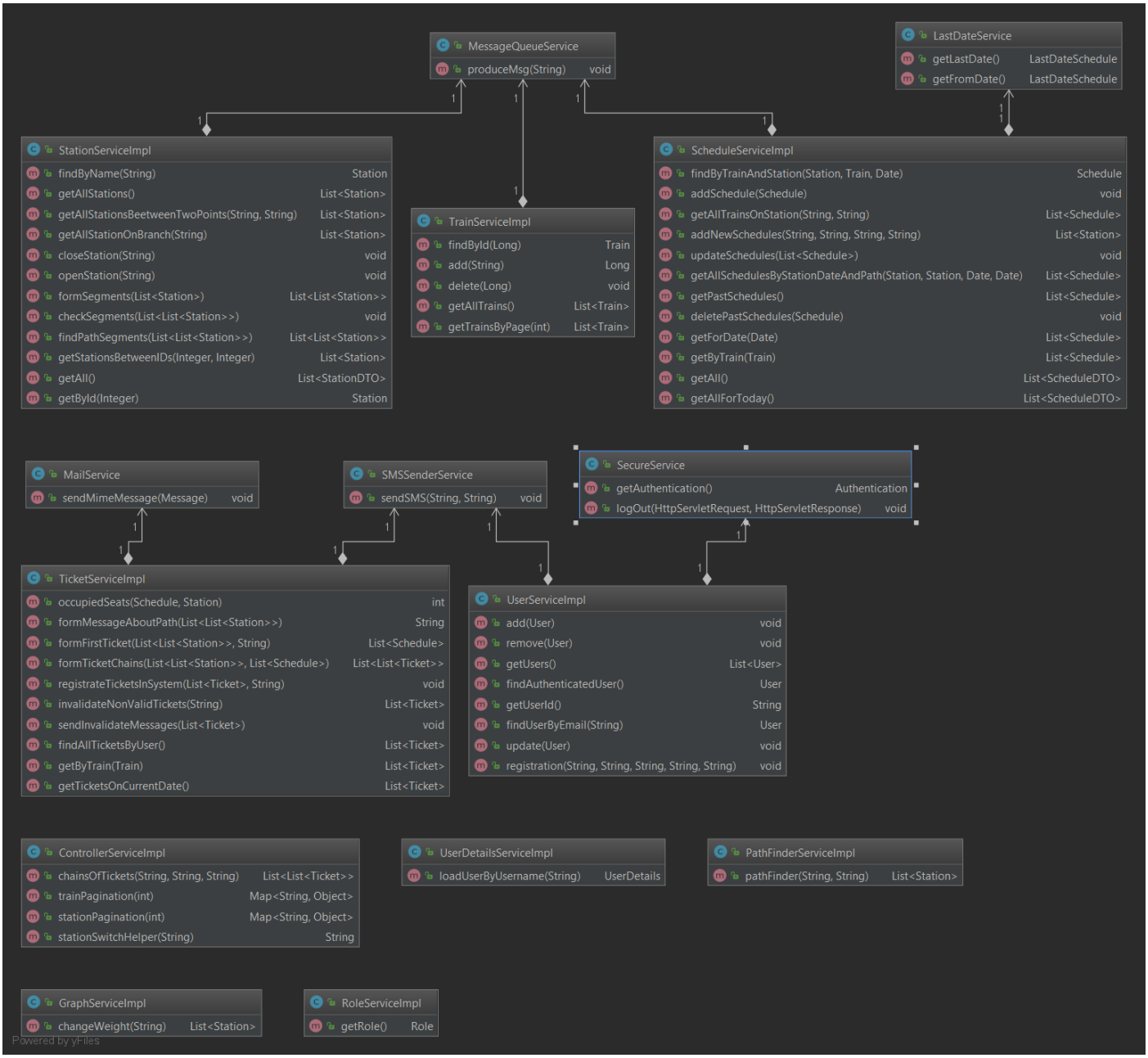
Model Level:


Status
id Long
statusName String
Branch
id Long
branchColor String
Role
id Long
type String
User
id Long
firstName String
lastName String
birthDay String
login String
password String
phone String
roles Set<Role>
Station
id Integer
name String
status Status
branch Branch
numberOnBranch Integer
Train
id Long
trainName String
capacity Integer
status Status
Ticket
id Long
user User
train Train
stationBegin Station
stationEnd Station
ticketDateDeparture Date
ticketDateArrival Date
price Integer
branch Branch
valid String
Graph
id Long
stationFrom Station
stationTo Station
weight Integer
oldWeight Integer
Schedule
id Long
dateDeparture Date
dateArrival Date
train Train
station Station
endPointStation Station
Message
addressee String
text String
subject String
context String
LastDateSchedule
id Long
dateSchedule Date
Powered by yFiles

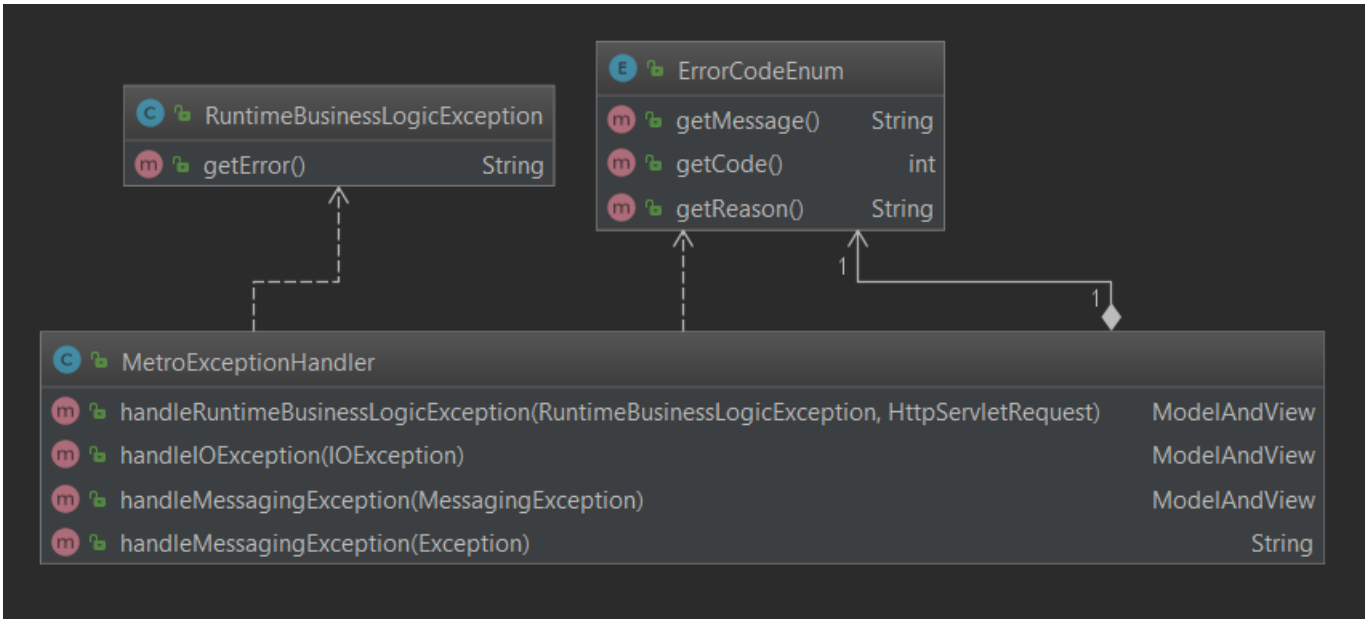
DAO Level:


StatusDAOImpl
getWorkStatus() Status
getCloseStatus() Status
getDestroyedStatus() Status
GraphDAOImpl
getAllFromNodes(Station) List<Graph>
getAllToNodes(Station) List<Graph>
getAllNodes(Station) List<Graph>
getAll() List<Graph>
LastDateDAOImpl
getLastDate() LastDateSchedule
getFromDate() LastDateSchedule
StationDAOImpl
findByName(String) Station
getAllStationOnBranch(Station) List<Station>
findStationById(Integer) Station
findAllStationBetweenBeginAndEndPoint(Station, Station) List<Station>
getStationsBetweenIDs(Integer, Integer) List<Station>
getById(Integer) Station
ScheduleDAOImpl
getByStationAndDate(Station, Date) List<Schedule>
findByTrainAndStation(Train, Station, Date) Schedule
getByStationsAndDate(Station, Station, Date, Date) List
getPastSchedules() List<Schedule>
getForToday() List<Schedule>
getForDate(Date) List<Schedule>
findByTrain(Train) List<Schedule>
getForCheckOnCreatingTrain(Date, Station, Station) List<Schedule>
GenericDAOImpl
add(E) void
update(E) void
delete(E) void
getAll() List<E>
getById(Long) E
RoleDAOImpl
getRole() Role
TicketDAOImpl
getByStationAndDate(Station, Date) List<Ticket>
getByStation(Station) List<Ticket>
getByStationDateTrain(Station, Station, Date, Train) List<Ticket>
findAllInvalidTickets() List<Ticket>
findAllUserTickets(User) List<Ticket>
getByTrain(Train) List<Ticket>
getTicketsOnCurrentDate() List<Ticket>
UserDAOImpl
updateProfile(User) void
findUserByEmail(String) User
TrainDAOImpl
findByName(String) Train
getTrainByPage(int) List<Train>
Powered by yFiles

Service Level:



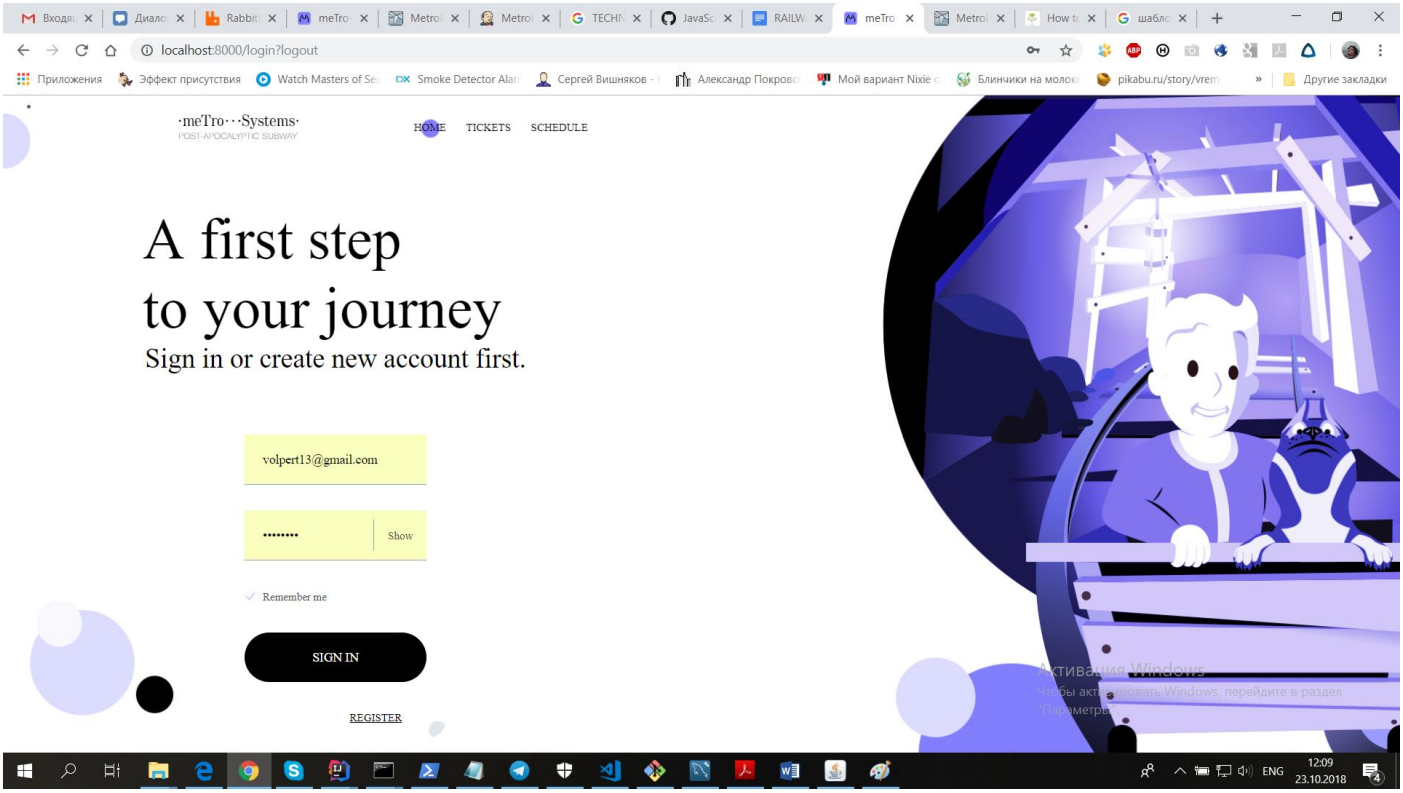
Exception Handler:

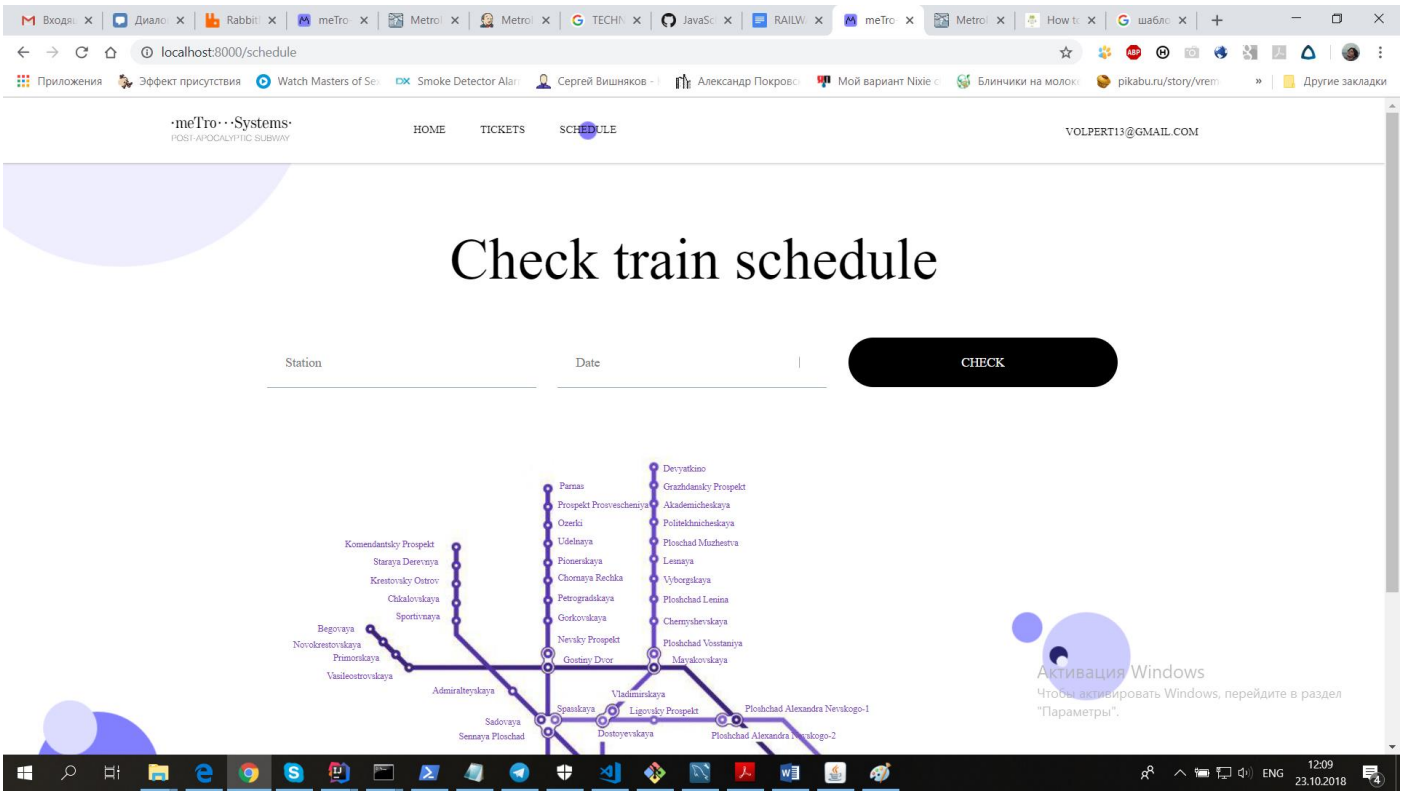


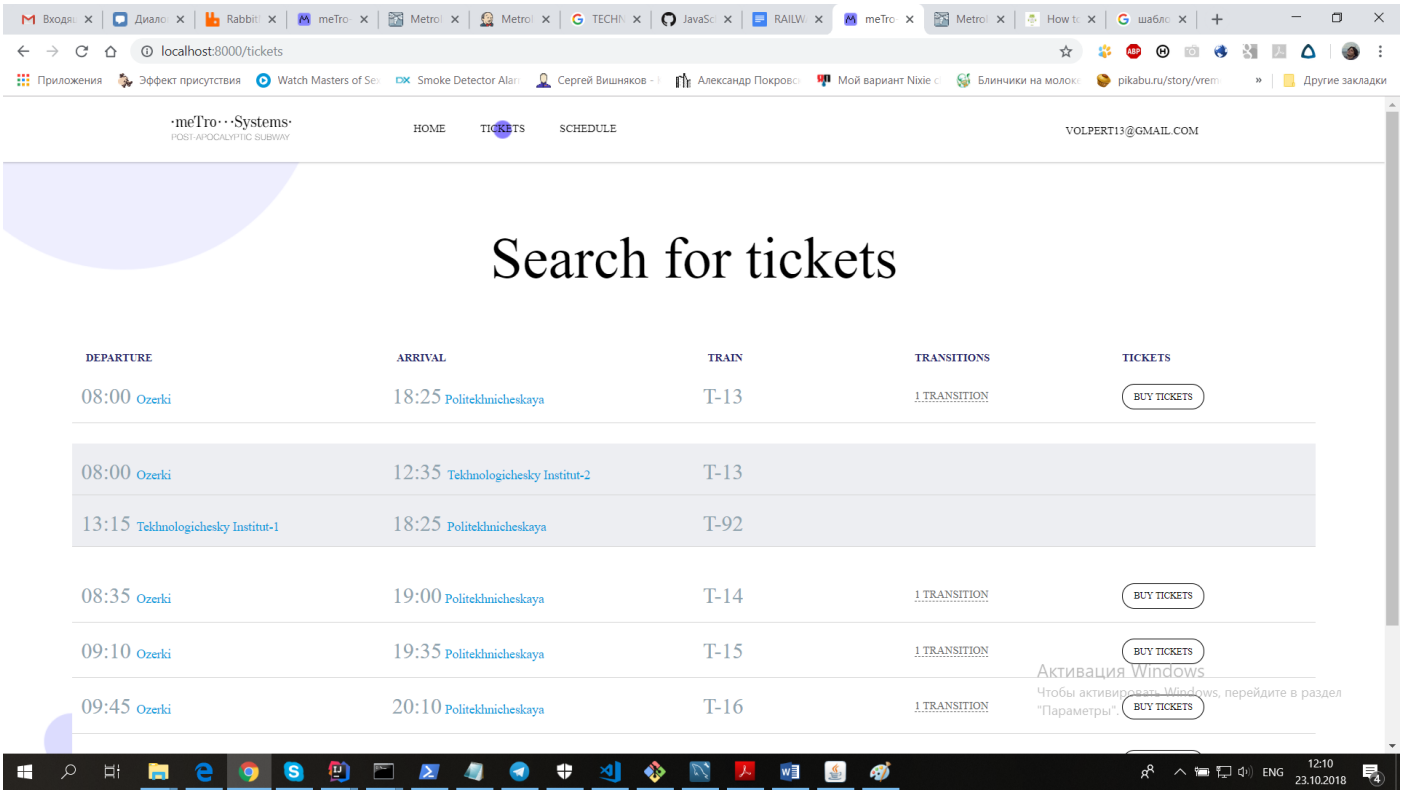
Error Codes:

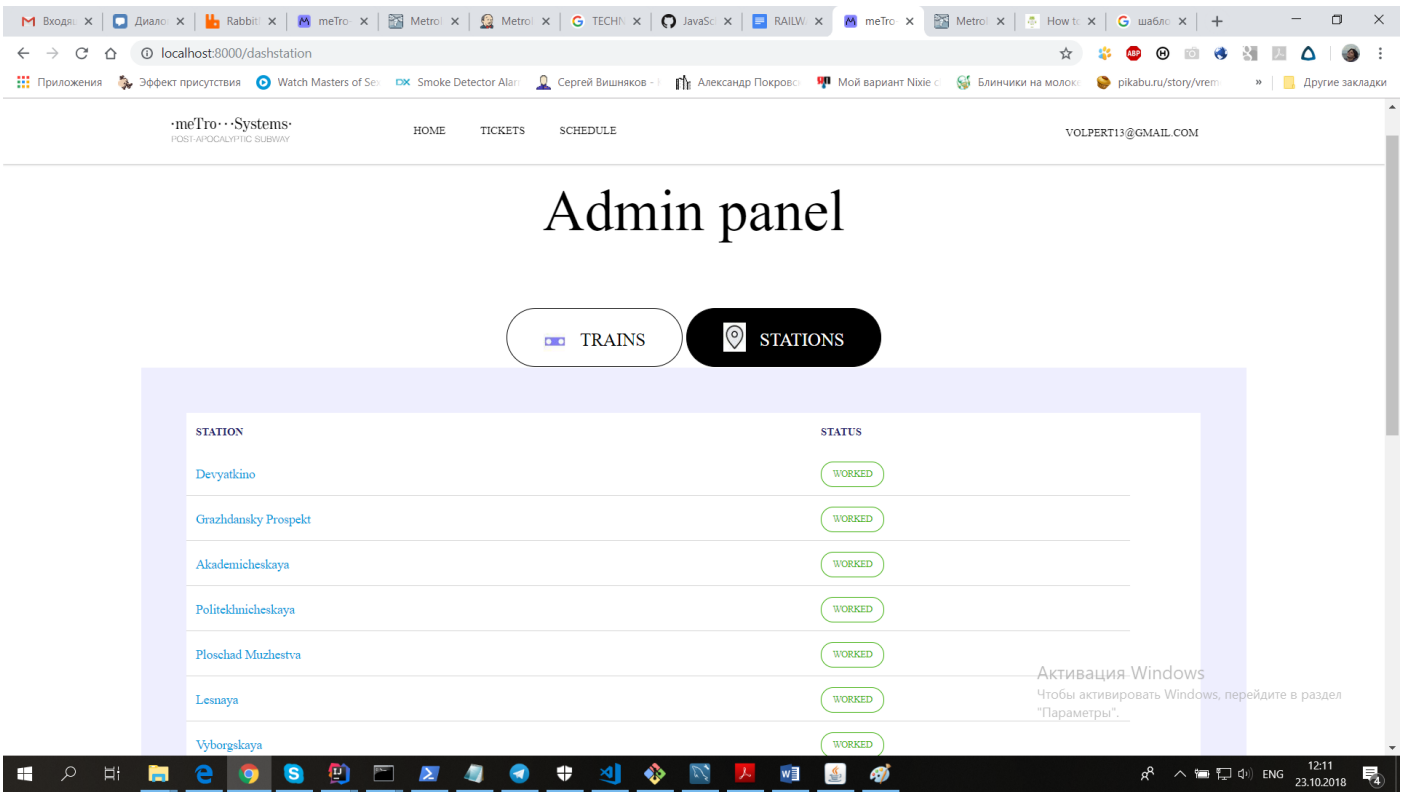
```
USER_EXIST(1, "User already exist"),
EMPTY_FIELDS(2, "One or two fields are empty"),
BEGIN_STATION_CLOSED(3, "Begin station is closed"),
END_STATION_CLOSED(4, "End station is closed"),
ATS_ARE_CLOSED(5, "All Transition Stations are Closed. Can't find the way"),
STATION_CLOSED(6, "Station closed for some reason"),
NO_MORE_TICKETS(7, "All tickets was already booked. Please try new search"),
TRAIN_EXIST(8, "Such train already exist"),
NO_TRAIN_ON_DATE(9, "No trains on that date"),
INCORRECT_CRED(10, "Password must contain not less 6 numbers or letters"),
INCORRECT_EMAIL(11, "Invalid email"),
TO_LATE_FOR_TRAIN(12, "To late for train"),
EMPTY_FIELDS_TRAIN_FORM(13, "One or two fields are empty in train form are empty"),
INCORRECT_DATE_SCHEDULE(14, "Your date is in past"),
INCORRECT_DATE_TICKETS(15, "You trying to get tickets in past"),
TRAIN_IN_PAST(16, "You trying to add train in past"),
DONT_KNOW_STATION (17, "No such station in that form"),
THAT_TIME_ALREADY_USED_BY_ANOTHER_TRAIN (18, "That time already used by another
train"),
NOT_AUTHORIZED_ADDING (19, "You trying to add train past schedules time border");
```

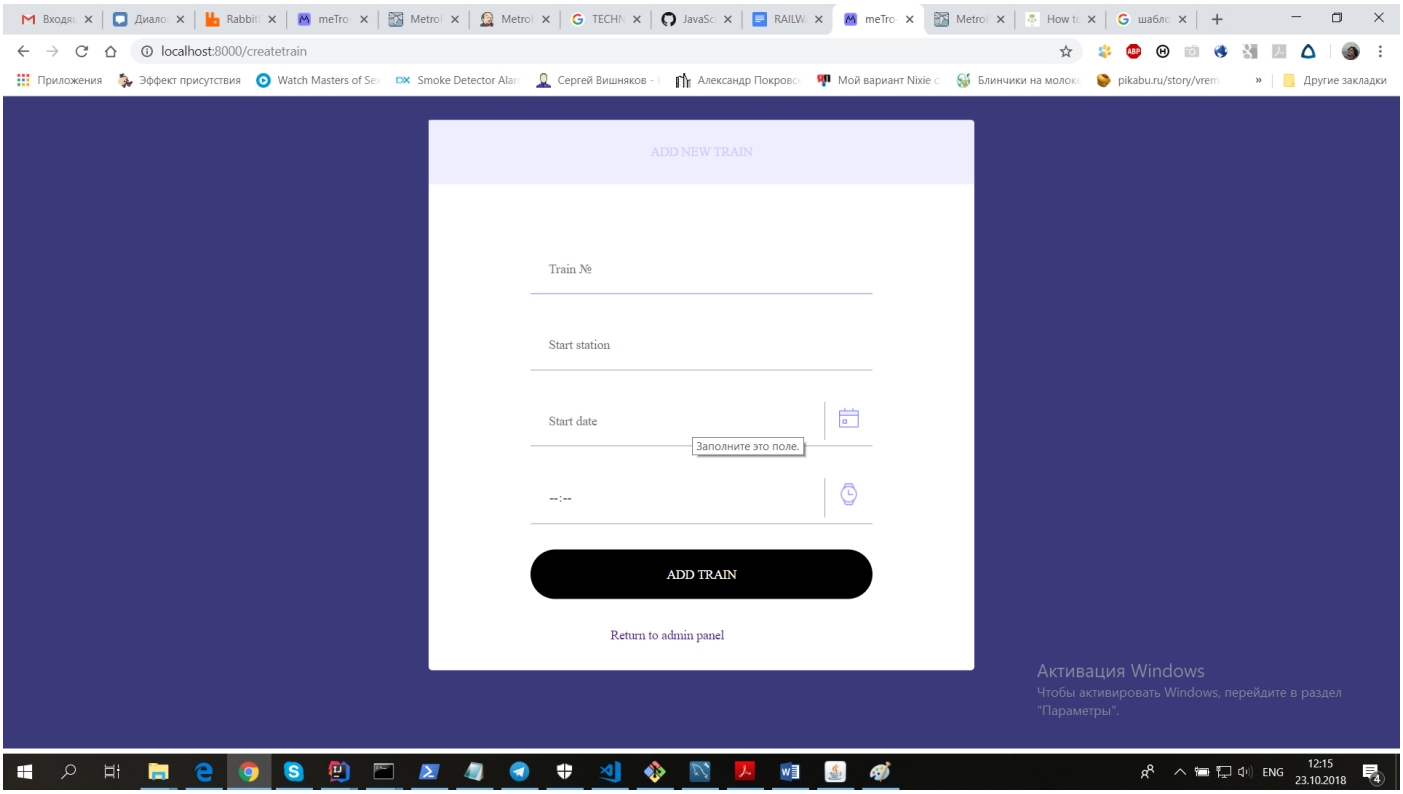
8. USER INTERFACE EXAMPLES

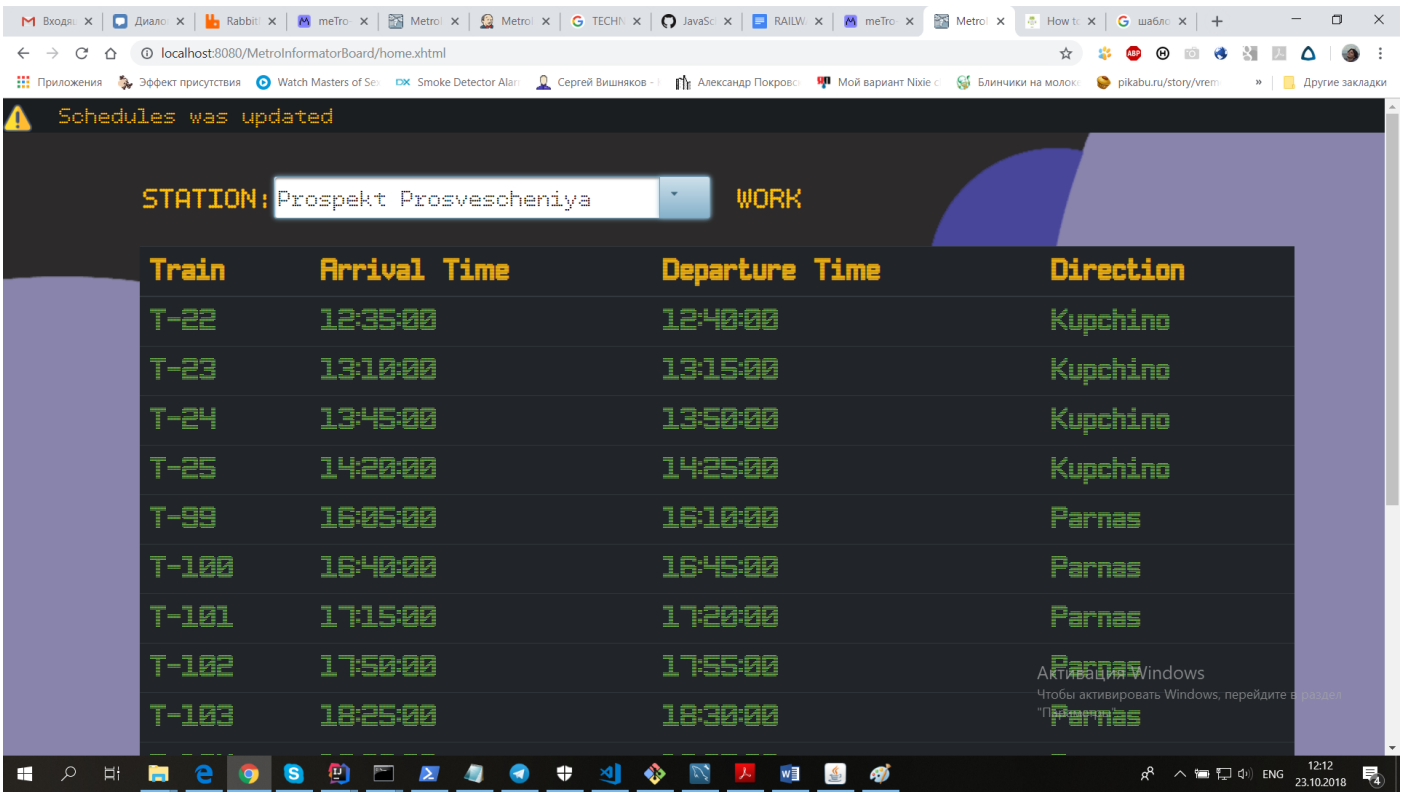





localhost:8000/tickets
·meTro···Systems·
POST-APOCALYPTIC SUBWAY
HOME
TICKETS
SCHEDULE
VOLPERT13@GMAIL.COM
Search for tickets
DEPARTURE
ARRIVAL
TRAIN
TRANSITIONS
TICKETS
08:00 Ozerki
18:25 Politekhnicheskaya
T-13
1 TRANSITION
BUY TICKETS
08:00 Ozerki
12:35 Tekhnologichesky Institut-2
T-13
13:15 Tekhnologichesky Institut-1
18:25 Politekhnicheskaya
T-92
08:35 Ozerki
19:00 Politekhnicheskaya
T-14
1 TRANSITION
BUY TICKETS
09:10 Ozerki
19:35 Politekhnicheskaya
T-15
1 TRANSITION
BUY TICKETS
09:45 Ozerki
20:10 Politekhnicheskaya
T-16
1 TRANSITION
BUY TICKETS
Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".
ENG
12:10
23.10.2018

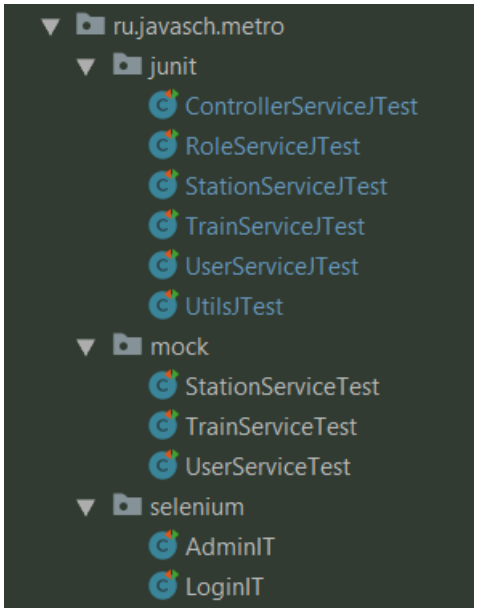

localhost:8000/dashstation
·meTro···Systems·
POST-APOCALYPTIC SUBWAY
HOME
TICKETS
SCHEDULE
VOLPERT13@GMAIL.COM
Admin panel
TRAINS
STATIONS
STATION
STATUS
Devyatkino
WORKED
Grazhdansky Prospekt
WORKED
Akademicheskaya
WORKED
Politekhnicheskaya
WORKED
Ploschad Muzhestva
WORKED
Lesnaya
WORKED
Vyborgskaya
WORKED
Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".
ENG
12:11
23.10.2018


localhost:8000/createtrain
ADD NEW TRAIN
Train №
Start station
Start date
Заполните это поле.
--:--
ADD TRAIN
Return to admin panel
Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".


localhost:8080/MetroInformatorBoard/home.xhtml
Schedules was updated
STATION: Prospekt Prosvescheniya
WORK
Train
Arrival Time
Departure Time
Direction
T-22 12:35:00 12:40:00 Kupchino
T-23 13:10:00 13:15:00 Kupchino
T-24 13:45:00 13:50:00 Kupchino
T-25 14:20:00 14:25:00 Kupchino
T-99 16:05:00 16:10:00 Parnas
T-100 16:40:00 16:45:00 Parnas
T-101 17:15:00 17:20:00 Parnas
T-102 17:50:00 17:55:00 Parnas
T-103 18:25:00 18:30:00 Parnas
Активация Windows

9. CODE QUALITY

Test Structure:



JUnit Tests:

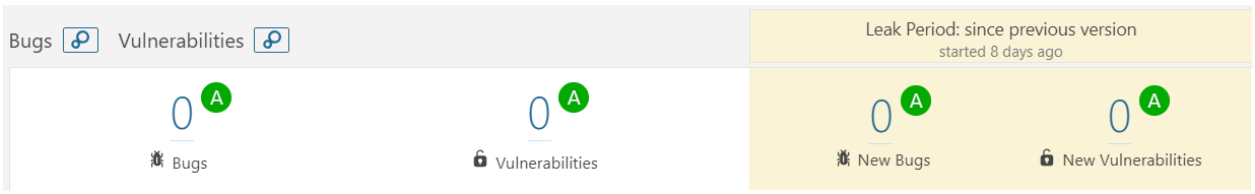
```
[INFO] Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 6.792 s - in ru.javasch.metro.junit.ControllerServiceJTest
[INFO] Running ru.javasch.metro.junit.RoleServiceJTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.005 s - in ru.javasch.metro.junit.RoleServiceJTest
[INFO] Running ru.javasch.metro.junit.StationServiceJTest
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.459 s - in ru.javasch.metro.junit.StationServiceJTest
[INFO] Running ru.javasch.metro.junit.TrainServiceJTest
INFO    2018-10-23 12:14:41,461 [main] ru.javasch.metro.service.implementations.TrainServiceImpl  - TRAIN ADDED
INFO    2018-10-23 12:14:41,464 [main] ru.javasch.metro.service.implementations.TrainServiceImpl  - TRAIN T-999 REMOVED
INFO    2018-10-23 12:14:41,602 [main] ru.javasch.metro.service.implementations.MessageQueueService  -  [x] Sent 'deletedtrain schedules []'
[INFO] Tests run: 5, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.378 s - in ru.javasch.metro.junit.TrainServiceJTest
[INFO] Running ru.javasch.metro.junit.UserServiceJTest
INFO    2018-10-23 12:14:41,820 [main] ru.javasch.metro.service.implementations.UserServiceImpl  - EXCEPTION: One or two fields are empty
[INFO] Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.179 s - in ru.javasch.metro.junit.UserServiceJTest
[INFO] Running ru.javasch.metro.junit.UtilsJTest
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.005 s - in ru.javasch.metro.junit.UtilsJTest
[INFO] Running ru.javasch.metro.mock.StationServiceTest
[INFO] Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.088 s - in ru.javasch.metro.mock.StationServiceTest
[INFO] Running ru.javasch.metro.mock.TrainServiceTest
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.005 s - in ru.javasch.metro.mock.TrainServiceTest
[INFO] Running ru.javasch.metro.mock.UserServiceTest
[INFO] Tests run: 5, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.004 s - in ru.javasch.metro.mock.UserServiceTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 31, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 14.401 s
[INFO] Finished at: 2018-10-23T12:14:42+03:00
[INFO] ------------------------------------------------------------------------
```

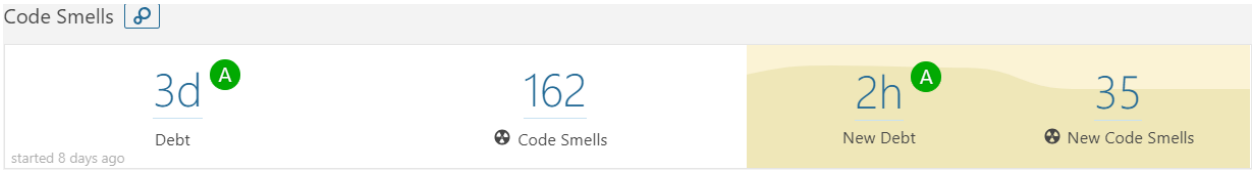
Selenium Tests:

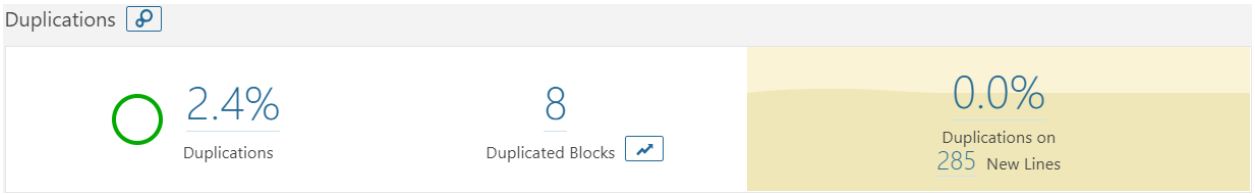
```
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running ru.javasch.metro.selenium.AdminIT
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 28063
Only local connections are allowed.
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 3941
Only local connections are allowed.
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 45296
Only local connections are allowed.
[INFO] Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 35.17 s - in ru.javasch.metro.selenium.AdminIT
[INFO] Running ru.javasch.metro.selenium.LoginIT
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 46622
Only local connections are allowed.
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 42516
Only local connections are allowed.
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 44509
Only local connections are allowed.
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 28756
Only local connections are allowed.
[INFO] Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 24.964 s - in ru.javasch.metro.selenium.LoginIT
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
```

Sonar Report:

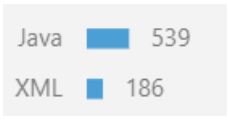
Main App:

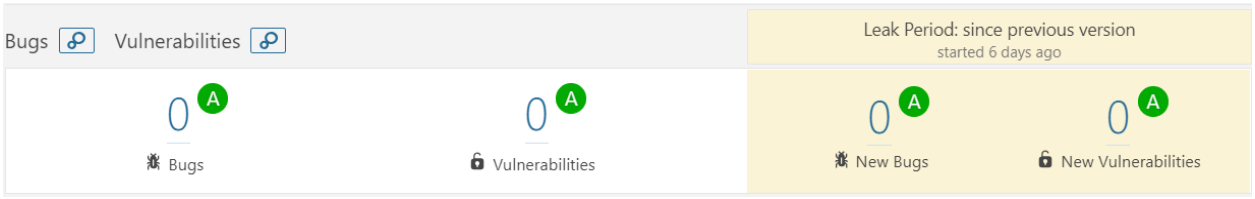


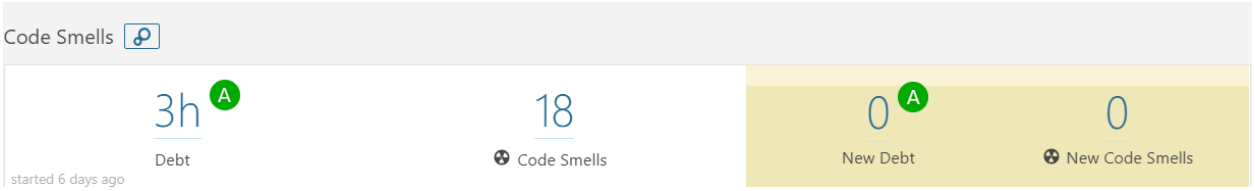


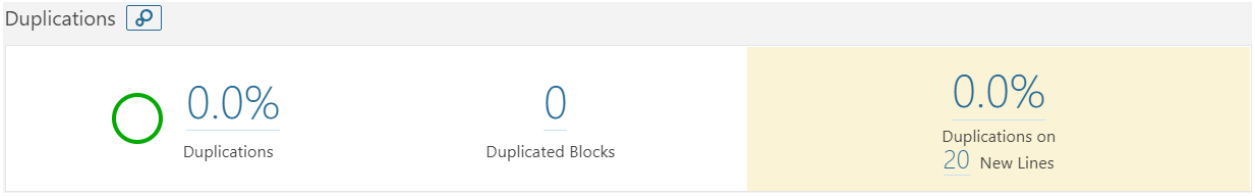


Scoreboard App:







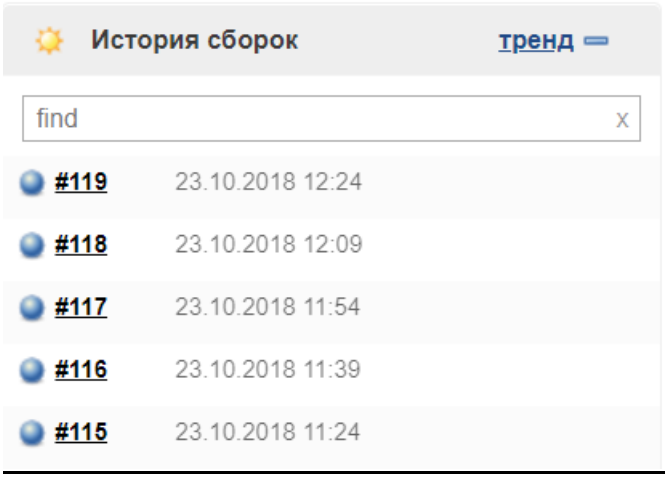


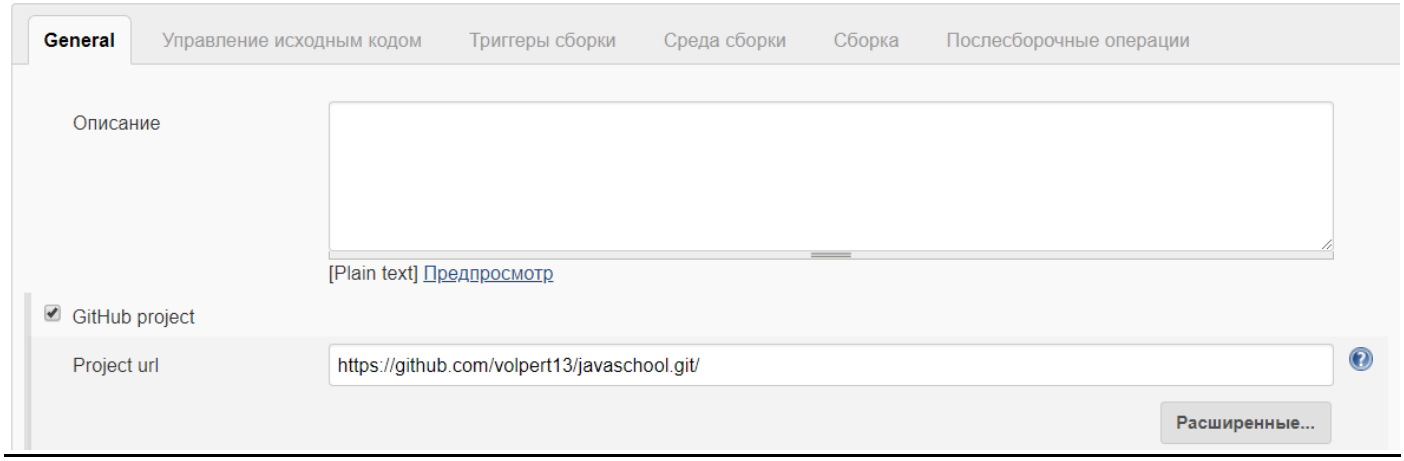
Logging:

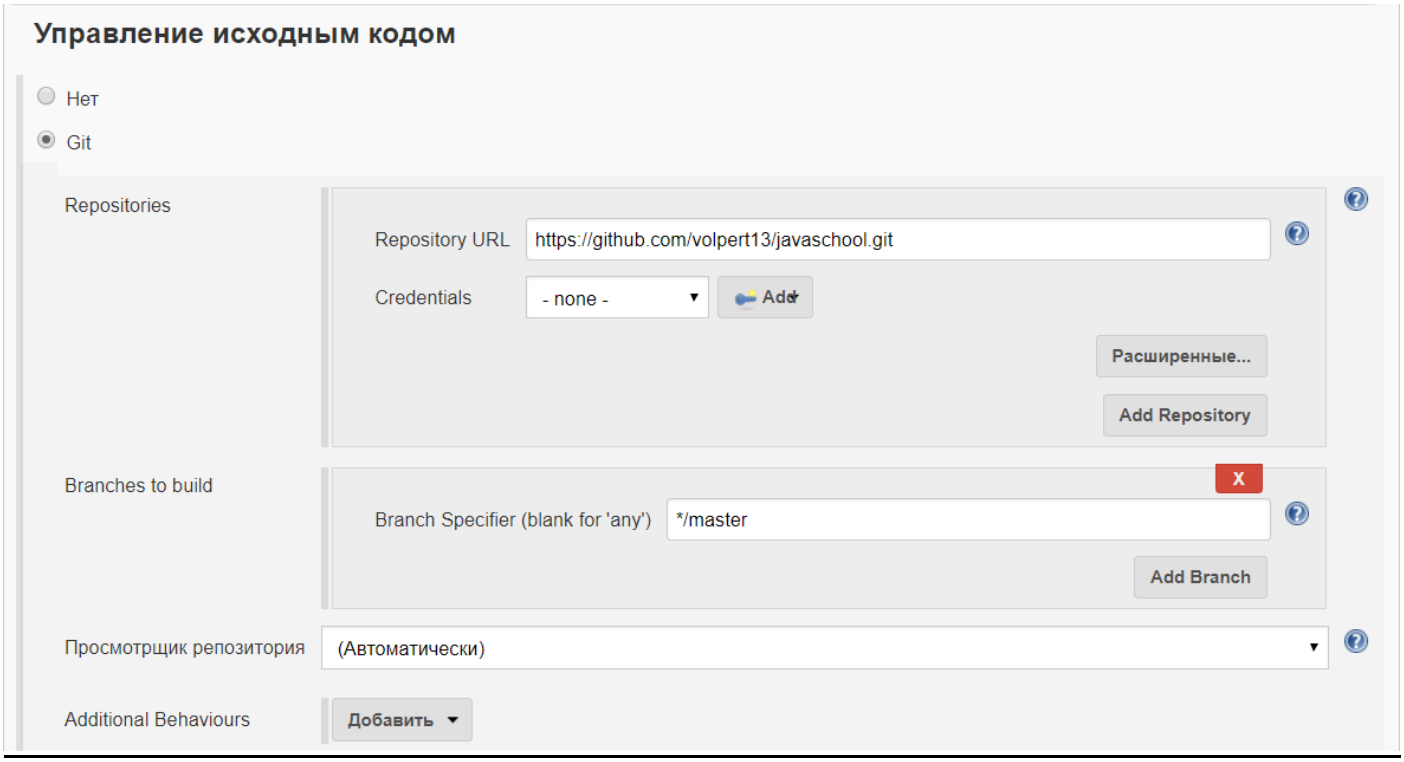
```
INFO    2018-10-23 12:30:06,137 [pool-1-thread-3] ru.javasch.metro.scheduled.ScheduledTasks  - deleted schedules[]
INFO    2018-10-23 12:30:06,138 [pool-1-thread-3] ru.javasch.metro.scheduled.ScheduledTasks  - MESSAGE SENDED: deleted schedules
INFO    2018-10-23 12:30:06,139 [pool-1-thread-3] ru.javasch.metro.scheduled.ScheduledTasks  - TIME SCHEDULE CLEANER END WORK. CLEANED 0 RECORDS
INFO    2018-10-23 12:30:56,046 [pool-1-thread-2] ru.javasch.metro.scheduled.ScheduledTasks  - TICKET INVALIDATOR CLEANER STARTED
INFO    2018-10-23 12:30:56,048 [pool-1-thread-2] ru.javasch.metro.scheduled.ScheduledTasks  - TICKET INVALIDATOR CLEANER END WORK. 0 WAS INVALIDATED
INFO    2018-10-23 12:31:06,148 [pool-1-thread-7] ru.javasch.metro.scheduled.ScheduledTasks  - TIME SCHEDULE CLEANER STARTED
INFO    2018-10-23 12:31:06,163 [pool-1-thread-7] ru.javasch.metro.service.implementations.MessageQueueService  -  [x] Sent 'deletedupdate schedules []'
INFO    2018-10-23 12:31:06,166 [pool-1-thread-7] ru.javasch.metro.scheduled.ScheduledTasks  - deleted schedules[]
INFO    2018-10-23 12:31:06,168 [pool-1-thread-7] ru.javasch.metro.scheduled.ScheduledTasks  - MESSAGE SENDED: deleted schedules
INFO    2018-10-23 12:31:06,170 [pool-1-thread-7] ru.javasch.metro.scheduled.ScheduledTasks  - TIME SCHEDULE CLEANER END WORK. CLEANED 0 RECORDS
INFO    2018-10-23 12:31:56,058 [pool-1-thread-8] ru.javasch.metro.scheduled.ScheduledTasks  - TICKET INVALIDATOR CLEANER STARTED
INFO    2018-10-23 12:31:56,059 [pool-1-thread-8] ru.javasch.metro.scheduled.ScheduledTasks  - TICKET INVALIDATOR CLEANER END WORK. 0 WAS INVALIDATED
INFO    2018-10-23 12:32:06,181 [pool-1-thread-1] ru.javasch.metro.scheduled.ScheduledTasks  - TIME SCHEDULE CLEANER STARTED
INFO    2018-10-23 12:32:06,189 [pool-1-thread-1] ru.javasch.metro.service.implementations.MessageQueueService  -  [x] Sent 'deletedupdate schedules []'
INFO    2018-10-23 12:32:06,192 [pool-1-thread-1] ru.javasch.metro.scheduled.ScheduledTasks  - deleted schedules[]
INFO    2018-10-23 12:32:06,193 [pool-1-thread-1] ru.javasch.metro.scheduled.ScheduledTasks  - MESSAGE SENDED: deleted schedules
INFO    2018-10-23 12:32:06,193 [pool-1-thread-1] ru.javasch.metro.scheduled.ScheduledTasks  - TIME SCHEDULE CLEANER END WORK. CLEANED 0 RECORDS
INFO    2018-10-23 12:32:56,065 [pool-1-thread-9] ru.javasch.metro.scheduled.ScheduledTasks  - TICKET INVALIDATOR CLEANER STARTED
INFO    2018-10-23 12:32:56,067 [pool-1-thread-9] ru.javasch.metro.scheduled.ScheduledTasks  - TICKET INVALIDATOR CLEANER END WORK. 0 WAS INVALIDATED
INFO    2018-10-23 12:33:06,204 [pool-1-thread-10] ru.javasch.metro.scheduled.ScheduledTasks  - TIME SCHEDULE CLEANER STARTED
INFO    2018-10-23 12:33:06,242 [pool-1-thread-10] ru.javasch.metro.service.implementations.MessageQueueService  -  [x] Sent 'deletedupdate schedules []'
INFO    2018-10-23 12:33:06,249 [pool-1-thread-10] ru.javasch.metro.scheduled.ScheduledTasks  - deleted schedules[]
INFO    2018-10-23 12:33:06,251 [pool-1-thread-10] ru.javasch.metro.scheduled.ScheduledTasks  - MESSAGE SENDED: deleted schedules
```

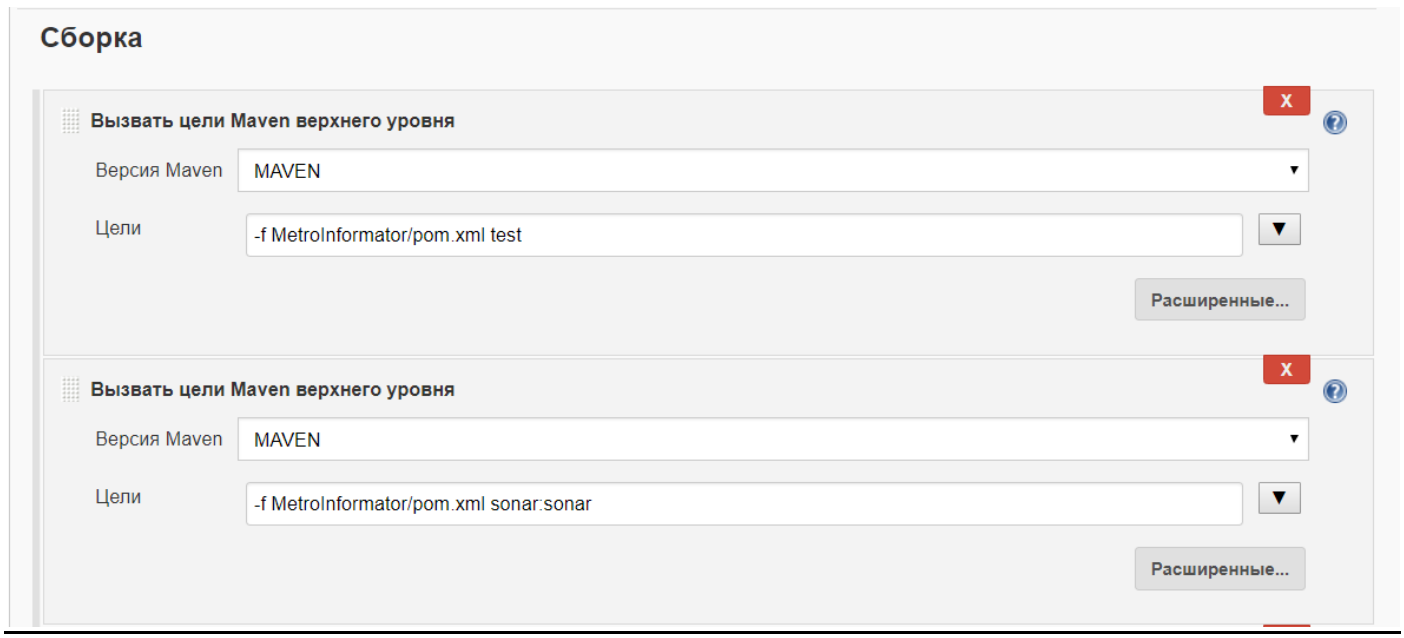
10. DEVELOPMENT WORKFLOW

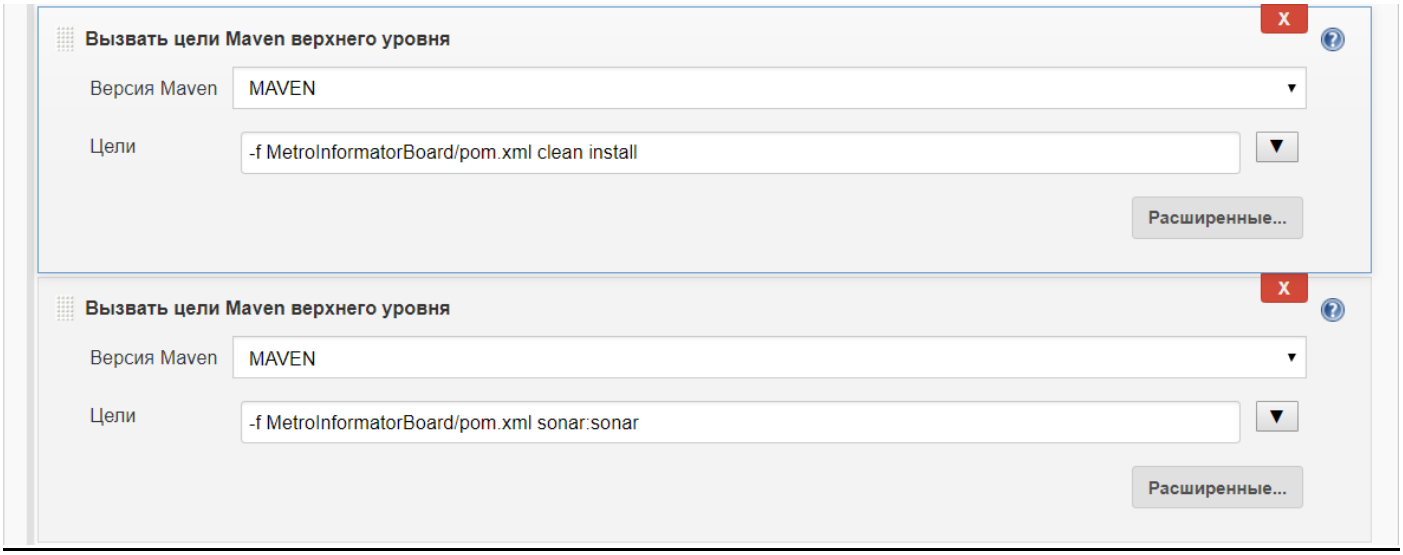
Jenkins (reports and settings):






Управление исходным кодом
Нет
Git
Repositories
Repository URL
https://github.com/volpert13/javaschool.git
Credentials
- none -
Add
Расширенные...
Add Repository
Branches to build
Branch Specifier (blank for 'any')
*/master
Add Branch
Просмотрщик репозитория
(Автоматически)
Additional Behaviours
Добавить

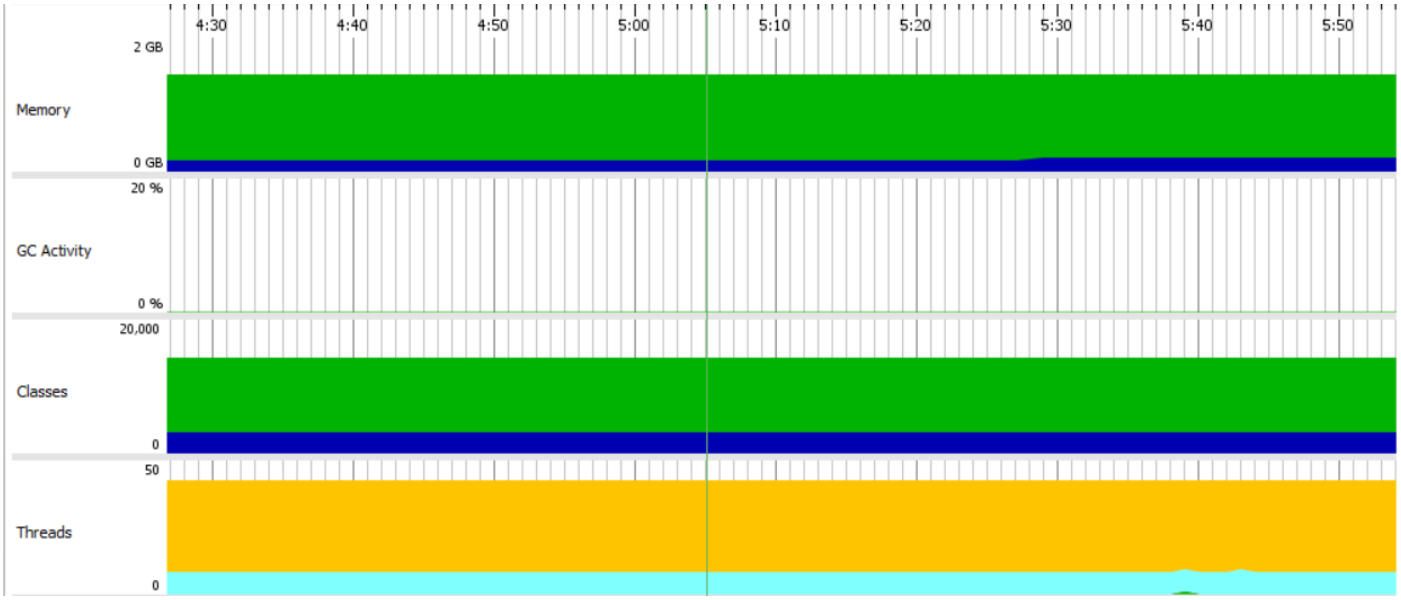

Сборка
Вызвать цели Maven верхнего уровня
Версия Maven
MAVEN
Цели
-f MetroInformator/pom.xml test
Расширенные...
Вызвать цели Maven верхнего уровня
Версия Maven
MAVEN
Цели
-f MetroInformator/pom.xml sonar:sonar
Расширенные...



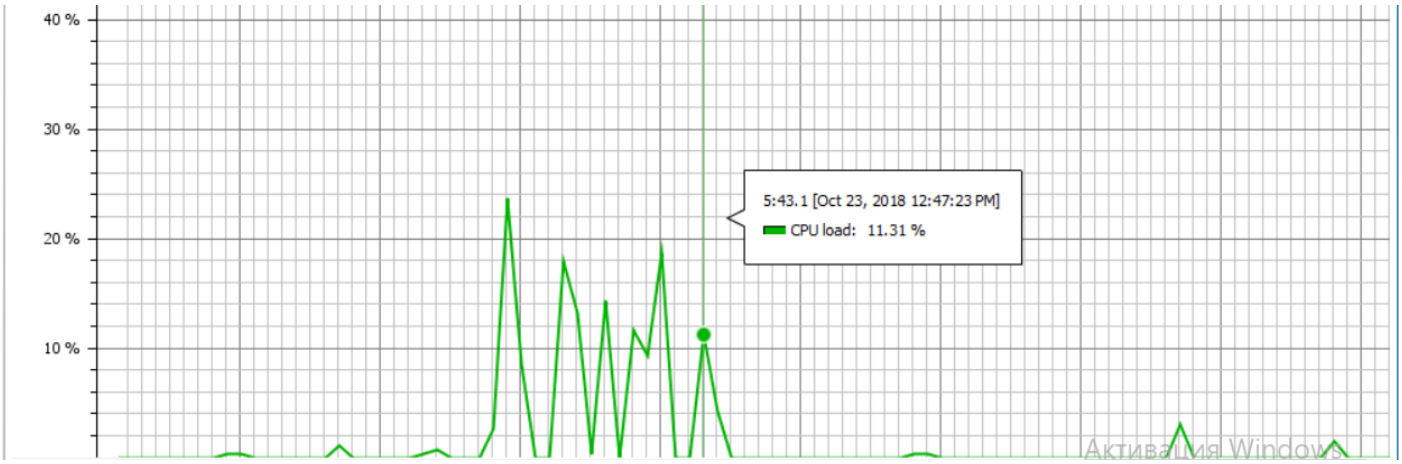
JProfiler metrics:

MAIN APP

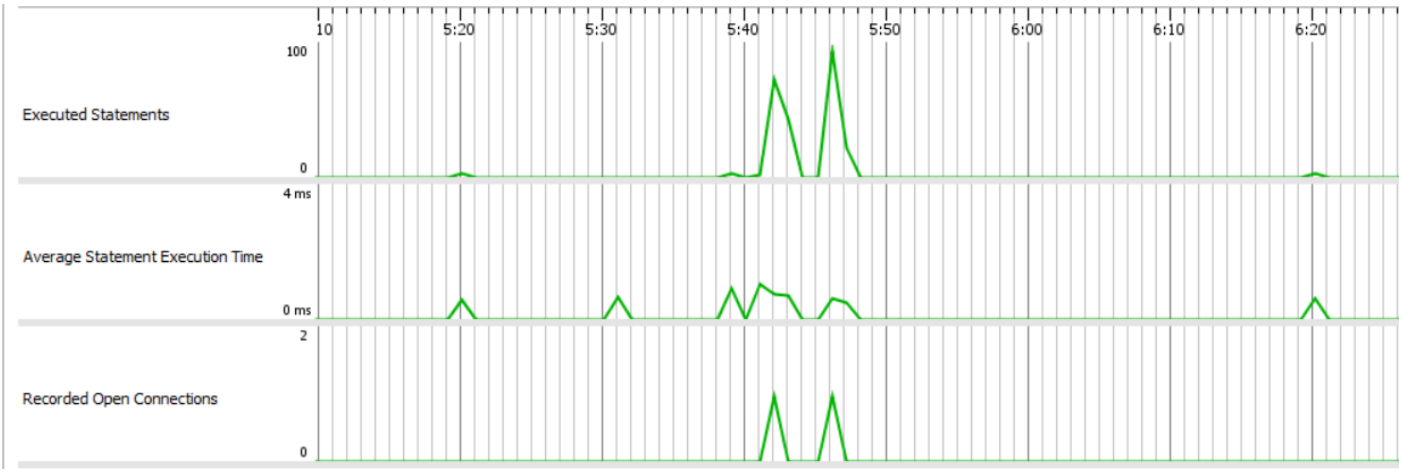
Common metrics



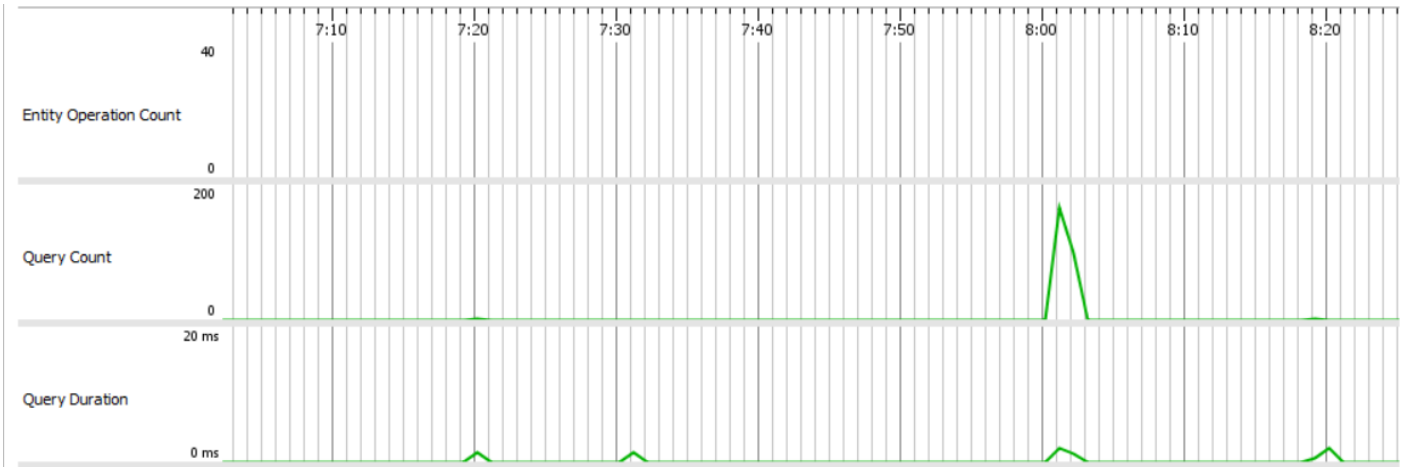
CPU Load



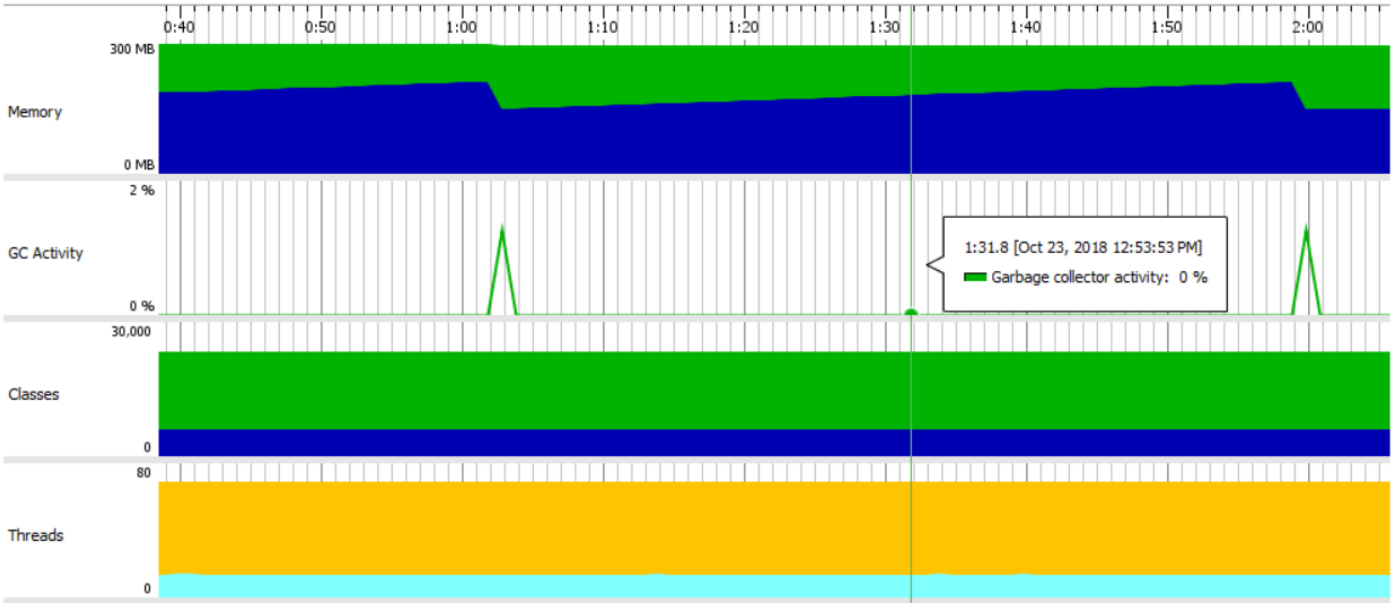
JDBC



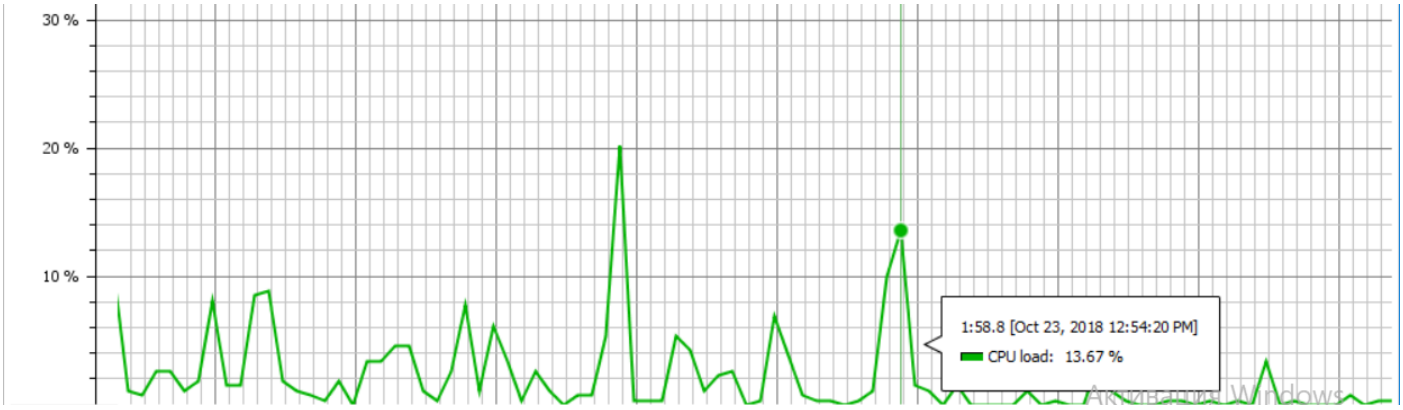
JPA/Hibernate



Common metrics



CPU Load



11. BUILD AND DEPLOY

Database:

mysql –u root –padmin

initialization script: fullcreator.sql

RabbitMQ:

rabbitmq-server start

Start Tomcat Application Server:

%CATALINA_HOME%/bin/startup.bat

Start Wildfly Application Server:

%WILDFLY_HOME%/bin/standalone.bat

Setup test database in Docker:

docker run -p 6161:3306 --name basename -e MYSQL_ROOT_PASSWORD=admin -d mysql:latest

docker cp ./docker-entrypoint-initdb.d/test.sql basename:docker-entrypoint-initdb.d/H2Script.sql

docker exec -it basename mysql -uroot -padmin

source ./docker-entrypoint-initdb.d/H2Script.sql

For Windows - be sure, that your account has password

Start/stop test database:

docker start basename

docker stop basename

Deploy main application:

mvn tomcat7:deploy

Test main application:

JUnit tests: *mvn test*

Selenium tests: *mvn failsafe:integration-test*

Deploy scoreboard application:

mvn wildfly:deploy

JProfiler (after application deployment):

Attach -> c:/WILDFLY/jboss-modules.jar –mp c:/WILDFLY/modules org.jboss.as.standalone –Djboss.home.dir=c:/WILDFLY -> Sampling

Attach -> org.apache.catalina.startup.Bootstrap start -> Sampling

Ports:

localhost:3306 – main DB

localhost:5673 – RabbitMQ

localhost:6161 – test DB in Docker

localhost:8000 – Tomcat

localhost:8080 – Wildfly

localhost:8282 – Jenkins

localhost:9000 – SonarQube

localhost:15672 – RabbitMQ Admin panel

12. FUTURE IMPROVEMENTS

- add QAuth2 authorization

- add payment system imitation

- improvement of user profiles system (trip history, editing information, etc.)

- additional application for management (branch load, ticket statistic, etc.)

- implementation of Testcontainers

- whole dockerization of system infrastructure

- refactoring code and performance improvement