

Университет ИТМО

Факультет программной инженерии и компьютерной техники

**Лабораторная работа №3**  
по «Бизнес-логика программных систем»

Выполнил:

Студент группы Р33101

Сударушкин Я. В.

Преподаватель:

Райла Мартин

Санкт-Петербург

2023

## **Задание**

Доработать приложение из лабораторной работы #2, реализовав в нём асинхронное выполнение задач с распределением бизнес-логики между несколькими вычислительными узлами и выполнением периодических операций с использованием планировщика задач.

### **Требования к реализации асинхронной обработки:**

1. Перед выполнением работы необходимо согласовать с преподавателем набор прецедентов, в реализации которых целесообразно использование асинхронного распределённого выполнения задач. Если таких прецедентов использования в имеющейся бизнес-процесса нет, нужно согласовать реализацию новых прецедентов, доработав таким образом модель бизнес-процесса из лабораторной работы #1.
2. Асинхронное выполнение задач должно использовать модель доставки "очередь сообщений".
3. В качестве провайдера сервиса асинхронного обмена сообщениями необходимо использовать очередь сообщений на базе Apache ActiveMQ.
4. Для отправки сообщений необходимо использовать протокол XMPP. Библиотеку для реализации отправки сообщений можно взять любую на выбор студента.
5. Для получения сообщений необходимо использовать слушателя сообщений JMS на базе Spring Boot (@JmsListener).

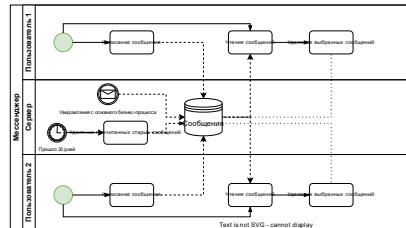
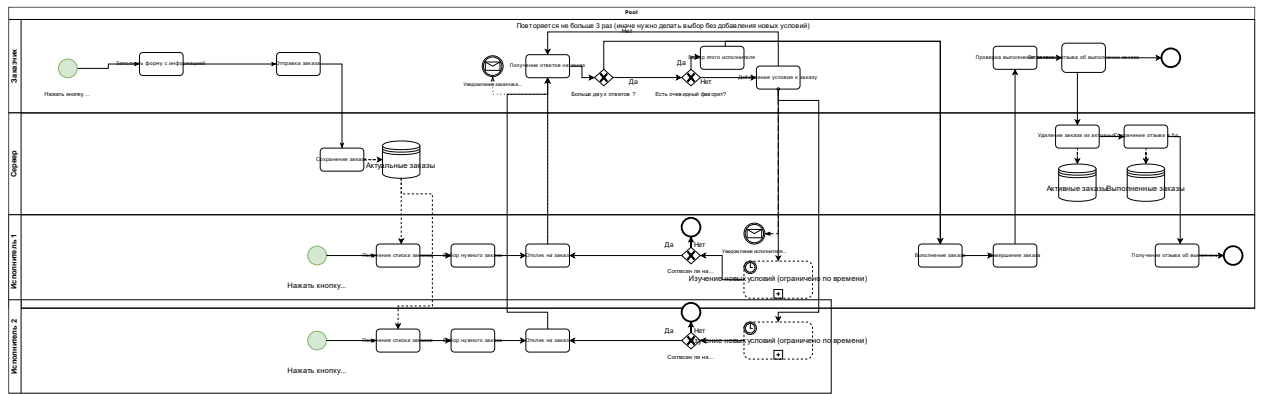
### **Требования к реализации распределённой обработки:**

1. Обработка сообщений должна осуществляться на двух независимых друг от друга узлах сервера приложений.
2. Если логика сценария распределённой обработки предполагает транзакционность выполняемых операций, они должны быть включены в состав распределённой транзакции.

### **Требования к реализации запуска периодических задач по расписанию:**

1. Согласовать с преподавателем прецедент или прецеденты, в рамках которых выглядит целесообразным использовать планировщик задач. Если такие прецеденты отсутствуют -- согласовать с преподавателем новые и добавить их в модель автоматизируемого бизнес-процесса.
2. Реализовать утверждённые прецеденты с использованием планировщика задач Spring (@Scheduled).

## **Модель**



[https://github.com/SudYar/BLPS\\_3/blob/master/BPMN.svg](https://github.com/SudYar/BLPS_3/blob/master/BPMN.svg)

## Спецификация пользовательских привилегий и ролей

Есть две роли: Заказчики и Исполнители.

Роль пользователь выбирает при регистрации

В зависимости от роли можно пользоваться Api- заказчика или исполнителя (у них свои контроллеры)

Вне зависимости от роли можно регистрироваться и заходить.

## REST API

Auth

POST sign-in Executor

POST sign-in Employer

POST sign-up

POST sign-up Copy

Employer

POST new\_order

DEL deleteOrder

DEL deleteAll

POST

http://localhost:21470/auth/sign-up

Params

Authorization

Headers

Body

Pre-request Script

Tests

Setting

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

```

1  POST
2  ... "login": "user3",
3  ... "password": "qwerty",
4  ... "role": "EXECUTOR"
5  
```

+

Auth

POST sign-in Executor

POST sign-in Employer

POST sign-up

POST sign-up Copy

Employer

POST new\_order

DEL deleteOrder

DEL deleteAll

GET GetNotice

Employer / new\_order

POST http://localhost:21470/employer/newOrder

Params Authorization Headers Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   ... "address": "SPB",
3   ... "description": "Tesssttt",
4   ... "price": 6
5 }
```

+

Auth

Employer

Executor

GET get\_orders

GET get\_info

POST chooseOrder

GET getNotice

POST finish

GET MyFeedbacks

Executor / MyFeedbacks

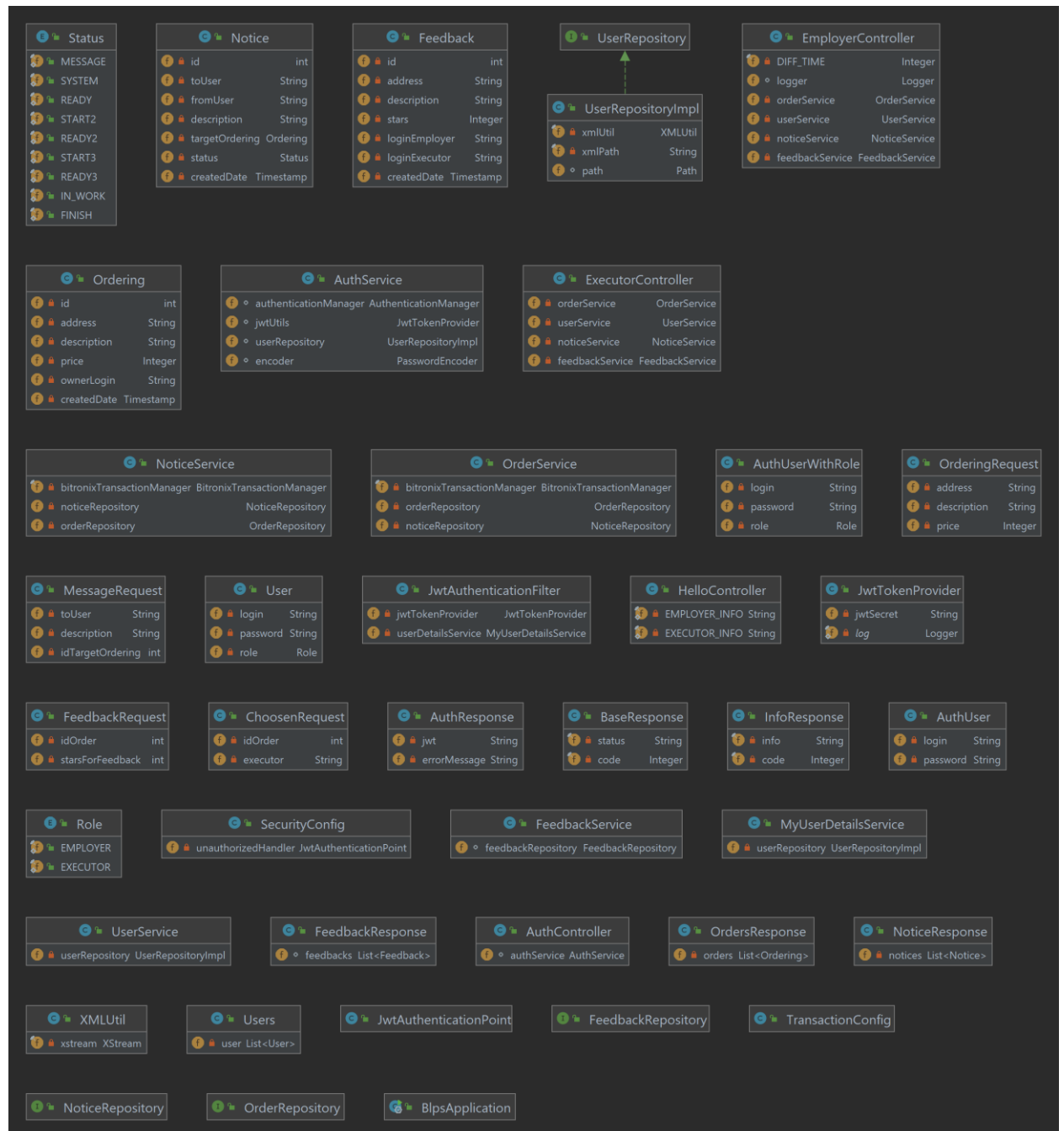
GET http://localhost:21470/executor/myFeedbacks

Params Authorization Headers Body Pre-request Script Tests Settings

Query Params

Key	Value
Key	Value

# UML



## Код

[https://github.com/SudYar/BLPS\\_3 Main](https://github.com/SudYar/BLPS_3_Main)

[https://github.com/SudYar/BLPS\\_3 Messenger](https://github.com/SudYar/BLPS_3_Messenger)

## Вывод

Теперь приложений 2