

Федеральное государственное автономное образовательное  
учреждение высшего образования

Университет ИТМО

Дисциплина: Бизнес-логика программных систем

### **Лабораторная работа 3**

Вариант 3076

**Выполнили:**

Марков Петр Денисович  
Кривоносов Егор Дмитриевич

**Группа: Р33111**

**Преподаватель:**

Горбунов Михаил Витальевич

2022 г.

Санкт-Петербург

# 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

- 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.
- Асинхронное выполнение задач должно использовать модель доставки "очередь сообщений".
- В качестве провайдера сервиса асинхронного обмена сообщениями необходимо использовать очередь сообщений на базе Apache ActiveMQ.
- Для отправки сообщений необходимо использовать протокол STOMP. Библиотеку для реализации отправки сообщений можно взять любую на выбор студента.
- Для получения сообщений необходимо использовать слушателя сообщений JMS на базе Spring Boot (@JmsListener).

## Требования к реализации распределённой обработки

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

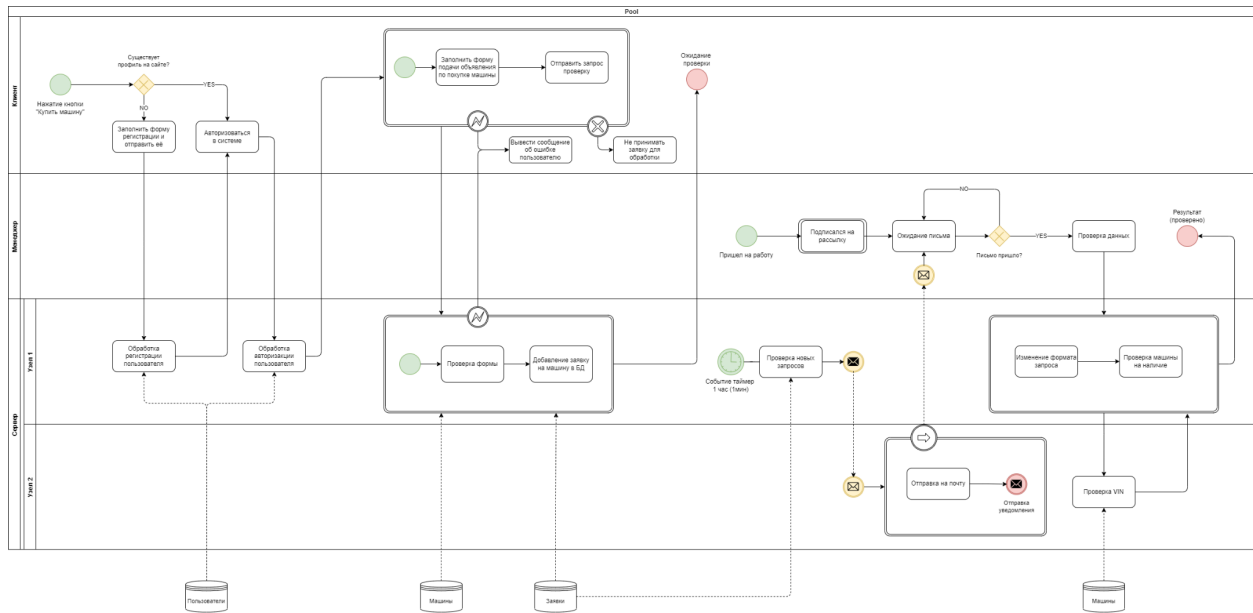
## Требования к реализации запуска периодических задач по расписанию

- 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.
- Реализовать утверждённые прецеденты с использованием планировщика задач Quartz.

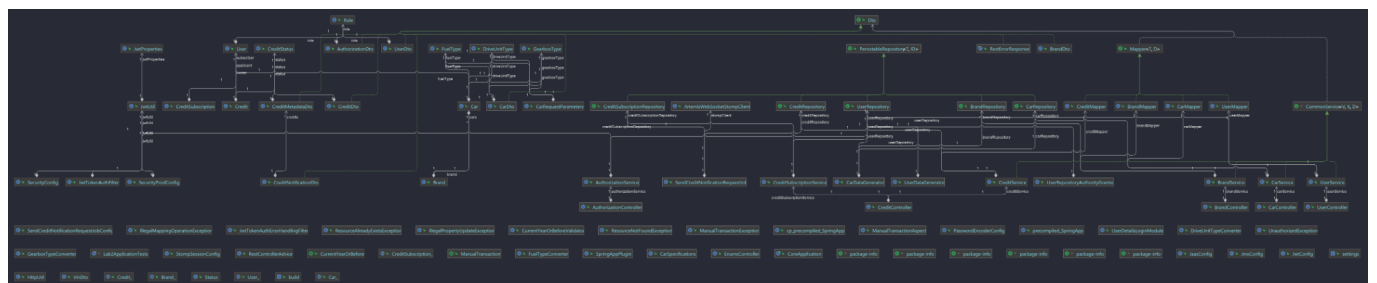
## Правила выполнения работы

- Все изменения, внесённые в реализуемый бизнес-процесс, должны быть учтены в описывающей его модели, REST API и наборе скриптов для тестирования публичных интерфейсов модуль
- Развернуть разработанное приложение на сервере helios на любом сервере с общедоступным URL для тестирования запросов.

# Модель потока управления



# UML-диаграмма



# REST API

drom.ru / Authorization / Login with User

Save

...

POST

http://localhost:8090/api/v1/login

S

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Query Params

KEY	VALUE	DESCRIPTION	...
Key	Value	Description	

Body

Cookies

Headers (14)

Test Results

⊕ Status: 200 OK Time: 877 ms Size: 730 B Save f

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

6

123456

```
"id": 1,
"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIjIjIyNyMUB1bWVpY20iLCJhdXRob3RpdGllcyI6WyJST0xFTX1VTRVlXSwiaWF0IjoxNjUyMzg4NTE4LCJleHAiOjE2NTIzOTU3MTh9.Scw1H51E0WF0IUVzW51l7g_UKInFYtQZzM_sgdvxUQZMDYavXoaxRyZkGF2Du-YG",
"expiresAt": "2022-05-12T22:48:38.00Z",
"role": "ROLE_USER"
```

GET

▼

http://localhost:8090/api/v1/cars

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

Query Params

	KEY	VALUE
	Key	Value

BodyCookiesHeaders (14)Test Results

PrettyRawPreviewVisualizeJSON ▼⌵

```
1  {
2    "content": [
3      {
4        "id": "24f94b24-fcca-4e33-90a7-71f4ed4e0103",
5        "vin": "4USBT53544LT26841",
6        "brandId": 1,
7        "modelName": "X3",
8        "year": 2015,
9        "city": "Saint-Petersburg",
10       "price": 1920000.0,
11       "mileage": 85450.0,
12       "generation": 2,
13       "driveUnitType": "AWD",
```

POST http://localhost:8090/api/v1/users/1/credits

Params Authorization Headers (9) Body Pre-request Script Tests Settings

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

```
1 {
2   ... "price": 2000000,
3   ... "vin": "7985d82f-8b6d-40f9-90a6-fc89ea73ada6"
4 }
```

Body Cookies Headers (12) Test Results

Status: 400 Bad Request Time: 110 ms

Pretty Raw Preview Visualize JSON

```
1 {
2   "timestamp": "2022-05-12T20:51:31.31Z",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Failed field validation with messages: wrong value for field vin: size must be between 17 and 17",
6   "requestUrl": "http://localhost:8090/api/v1/users/1/credits"
7 }
```

POST http://localhost:8090/api/v1/users/1/credits

Params Authorization Headers (9) Body Pre-request Script Tests S

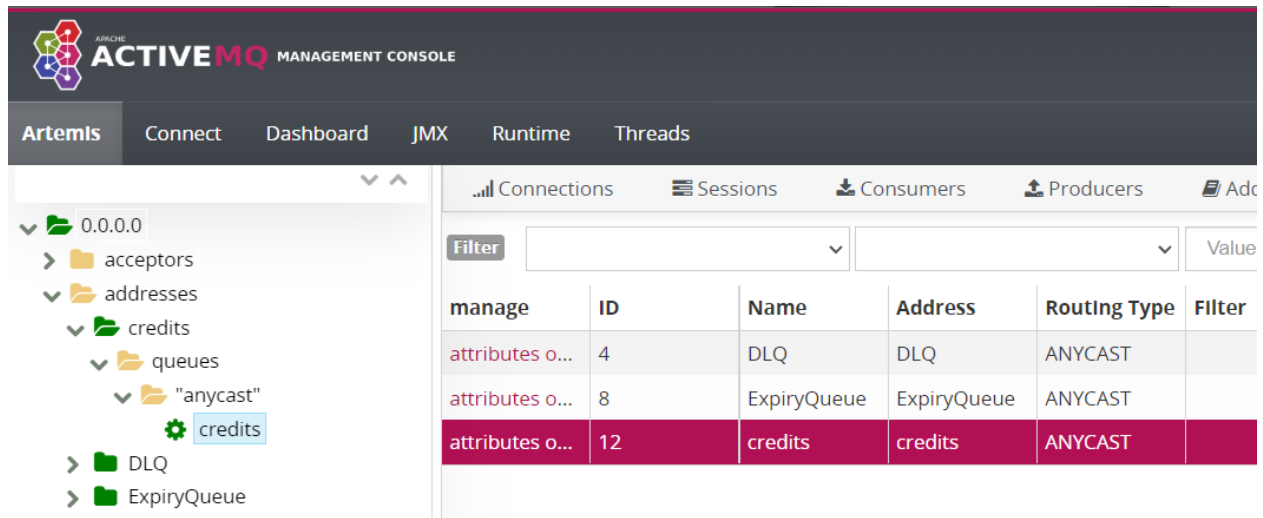
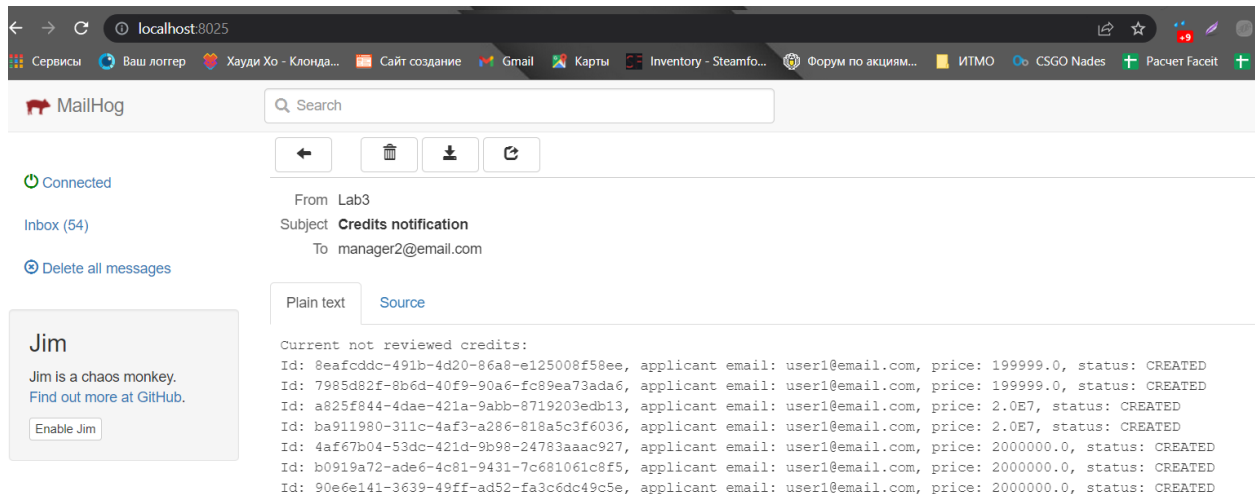
none form-data x-www-form-urlencoded raw binary GraphQL

```
1 {
2   ... "price": 2000000,
3   ... "vin": "4USBT53544LT26841",
4   ... "applicantId": 1,
5   ... "applicantEmail": "user1@email.com"
6 }
```

Body Cookies Headers (14) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": "90e6e141-3639-49ff-ad52-fa3c6dc49c5e",
3   "vin": "4USBT53544LT26841",
4   "price": 2000000.0,
5   "applicantId": 1,
6   "applicantEmail": "user1@email.com",
7   "managerId": null,
8   "managerEmail": null,
9   "status": "CREATED"
10 }
```



## Исходный код

Исходный код программы можно найти в репозитории на github:

[https://github.com/RedGry/blps\\_lab3](https://github.com/RedGry/blps_lab3)

## Вывод

В данной лабораторной работе мы познакомились с механизмом работы асинхронного выполнения задач, а также с выполнением периодических операций использованием планировщика задач. Помимо этого в процессе выполнения

данной лабораторной работы были разработаны REST API, а также была выполнена непосредственно реализация самого бизнес-процесса с помощью языка Java. Также были выявлены проблемы провайдера с Docker'ом при попытке запуска и настройки лабораторной работы.