

Университет ИТМО

Бизнес-логика программных систем
Лабораторная работа №1

Выполнил:
Бавыкин Роман Алексеевич
Группа Р33091
Вариант 1914

г. Санкт-Петербург
2024

1. Текст задания:

Вариант №1914: Студопедия - Ваша Школопедия - <https://studopedia.ru>

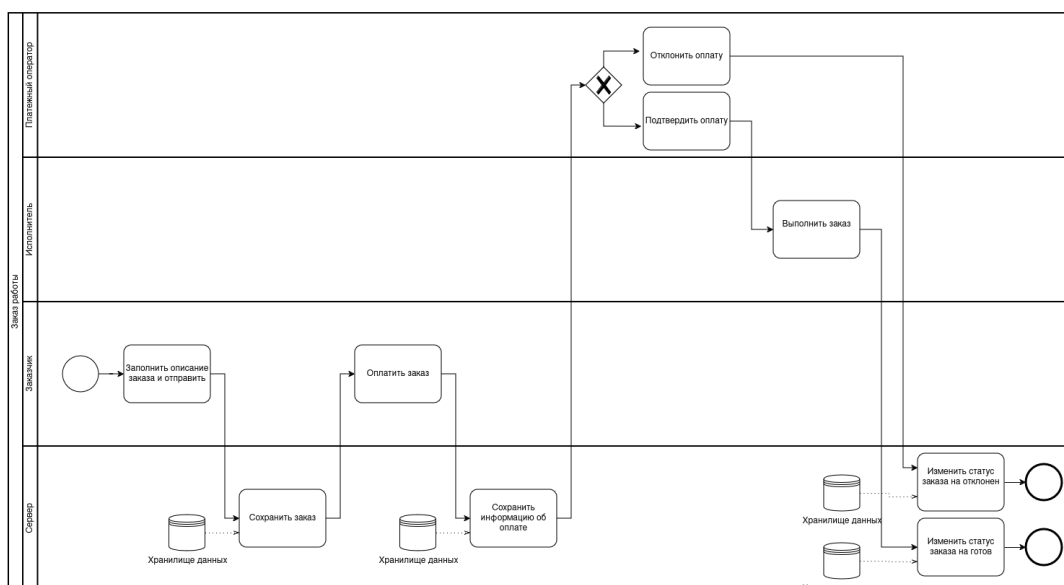
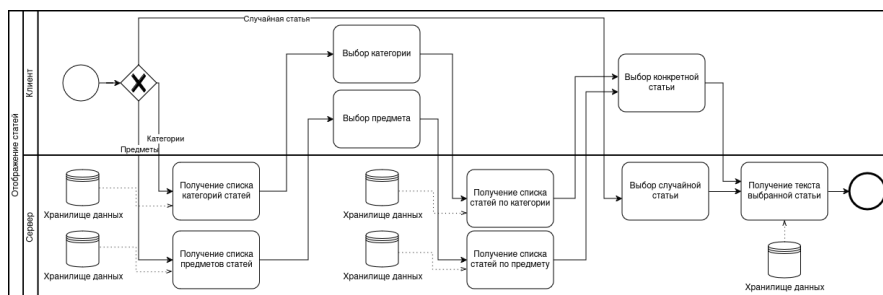
Описать бизнес-процесс в соответствии с нотацией BPMN 2.0, после чего реализовать его в виде приложения на базе Spring Boot.

Порядок выполнения работы:

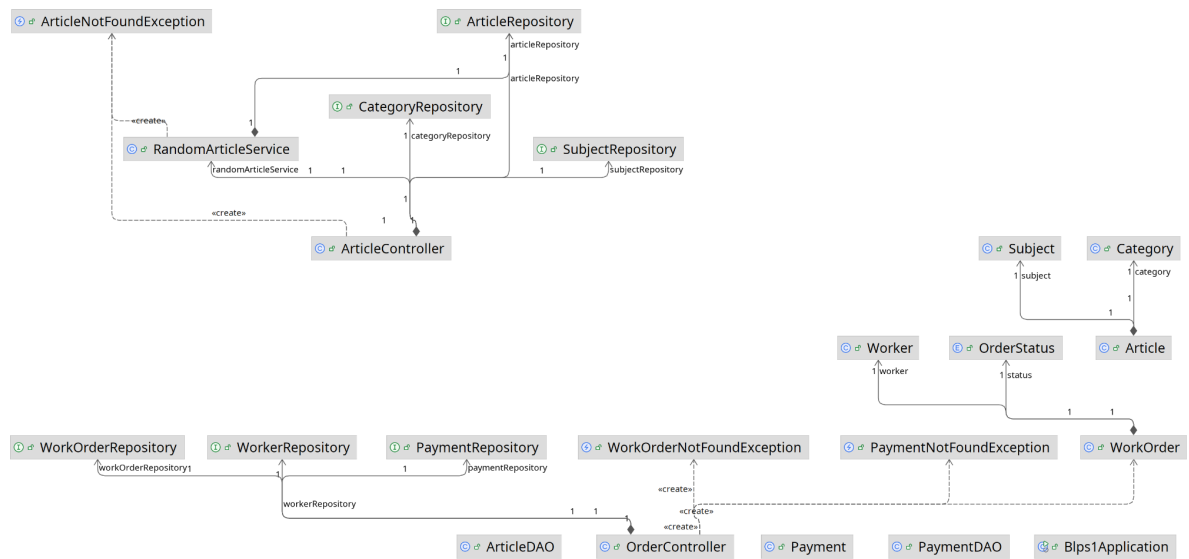
1. Выбрать один из бизнес-процессов, реализуемых сайтом из варианта задания.
2. Утвердить выбранный бизнес-процесс у преподавателя.
3. Специфицировать модель реализуемого бизнес-процесса в соответствии с требованиями BPMN 2.0.
4. Разработать приложение на базе Spring Boot, реализующее описанный на предыдущем шаге бизнес-процесс. Приложение должно использовать СУБД PostgreSQL для хранения данных, для всех публичных интерфейсов должны быть разработаны REST API.
5. Разработать набор curl-скриптов, либо набор запросов для REST клиента Insomnia для тестирования публичных интерфейсов разработанного программного модуля. Запросы Insomnia оформить в виде файла экспорта.
6. Развернуть разработанное приложение на сервере **helios**.

Активация Windows
Чтобы активировать Windows, перейдите к

2. Модель потока управления для автоматизируемого бизнес-процесса:



3. UML-диаграммы классов и пакетов разработанного приложения.



4. Спецификация REST API для всех публичных интерфейсов разработанного приложения.

GET /categories

- Response 200 (application/json)

```
[{"id":0,"name":"Авто"}, {"id":1,"name":"Биология"}, {"id":2,"name":"Генетика"}]
```

GET /subjects

- Response 200 (application/json)

```
[{"id":0,"name":"Авиадвигателестроение"}, {"id":1,"name":"Безопасность жизнедеятельности"}, {"id":2,"name":"Высшая математика"}]
```

GET /articles/0

- Response 200 (text/html)

GET /articles/category/0

- Response 200 (application/json)

```
[{"id":0,"name":"Я цифра – модификация"}, {"id":1,"name":"Жидкостей автомобильного транспорта"}]
```

GET /articles/subject/0

- Response 200 (application/json)

```
[{"id":4,"name":"Общие сведения о технологии авиадвигателестроения"}]
```

GET /articles/random

- Response 200 (text/html)

POST /order/new

- Request (text/plain)

Пример ТЗ

- Response 200 (application/json)

```
{"id": 502, "description": "Пример ТЗ", "status": "UNPAID", "deadline": null, "worker": null}
```

POST /payment

- Request (application/json)

```
{ "cardNumber": "1234 1234 1234 1234", "cardExpiration": "09.2025", "cardName": "Name Surname", "orderId": 502, "sum": 2000 }
```

- Response 200

```
{ "id": 102, "cardNumber": "1234 1234 1234 1234", "cardExpiration": "09.2025", "cardName": "Name Surname", "orderId": 502, "sum": 2000.0, "resolved": false }
```

GET /order/502

- Response 200

```
{ "id": 502, "description": "Пример ТЗ", "status": "UNPAID", "deadline": null, "worker": null }
```

GET /payment/102

- Response 200

```
{ "id": 102, "cardNumber": "1234 1234 1234 1234", "cardExpiration": "09.2025", "cardName": "Name Surname", "orderId": 502, "sum": 2000.0, "resolved": false }
```

PUT /payment/accept?paymentId=152

- Response 200

```
{ "id": 1, "description": "Пример ТЗ", "status": "AT_WORK", "deadline": "2024-06-22", "worker": { "id": 3, "name": "Worker3", "busy": true } }
```

PUT /payment/decline?paymentId=153

- Response 200

```
{ "id": 2, "description": "Пример ТЗ", "status": "DECLINED", "deadline": null, "worker": null }
```

PUT /order/ready?orderId=1

- Response 200

```
{ "id": 1, "description": "Пример ТЗ", "status": "READY", "deadline": "2024-06-22", "worker": { "id": 3, "name": "Worker3", "busy": false } }
```

5. Ссылка на исходный код

<https://github.com/robqqq/blps1>

6. Выводы по работе

Во время выполнения лабораторной работы ознакомился со спецификацией BPMN 2.0, на практическом примере специфицировал бизнес-процесс с помощью данной модели и разработал по нему приложение. По готовой спецификации удобно писать код, поскольку все уже спланировано и “разложено по полочкам” остается буквально “перевести” спецификацию на язык программирования.