

Санкт-Петербургский государственный политехнический университет
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Отчёт по курсовой работе
«Магазин продажи цветов «Flower Shop»
Дисциплина: «Архитектуры проектирования ПО»

Выполнила, студент гр. 13541/2: _____ О.А. Ивашкевич

Проверила: _____ А. Е. Зозуля

Санкт-Петербург
2019

Оглавление

1. Постановка задачи	3
1.2. Назначение проектируемой системы	3
1.3. Описание участников	3
1.4. Описание бизнес-процессов	3
1.3.1. Заказ	3
1.3.1. Доставка	4
1.3.1. Получение заказа	4
2. Варианты использования	4
2.2. Диаграмма вариантов использования для роли "Клиент"	4
2.3. Диаграмма вариантов использования для роли "Администратор"	5
2.4. Диаграмма вариантов использования для роли "Курьер"	6
3. Моделирование	6
3.2. Диаграмма для роли "Клиент"	6
3.3. Диаграмма для роли "Администратор"	7
3.4. Диаграмма для роли "Курьер"	7
3.5. Статическая модель предметной области	8
4. Слой бизнес логики	8
4.2. Проектирование	8
4.3. Реализация	8
5. Слой источников данных	9
6. Выводы	10

1. Постановка задачи

1.1. Назначение проектируемой системы

Система заказа цветов. Данная система предназначена для быстрого заказа цветов.

1.2. Описание участников

Клиент (Client): пользователь системы для заказа цветов.

- регистрация
- формирование заявки о покупке цветов

Представитель (Admin): работник магазина, который рассматривает заявки о покупке цветов, меняет описание магазина.

- регистрация
- формирование списка продуктов
- анализ поступающих заявок
- ответ на заявку
- подтверждение выполнения заказа

Курьер (Courier): работник, который занимается доставкой заказов.

- взятие заказа
- подтверждение доставки

1.3. Описание бизнес-процессов

1.3.1. Заказ (Ordering)

Участники:

- клиент
- представитель

Этапы:

- создание запроса на покупку
- рассмотрение запроса

- подтверждение запроса
- получение статуса заявки

1.3.2. Доставка (Ordering)

Участники:

- курьер

Этапы:

- взятие заказа на выполнение
- подтверждение доставки и оплаты заказа

1.3.3. Получение заказа

Участники:

- клиент
- представитель

Этапы:

- получение заказа
- подтверждение получения заказа клиентом
- подтверждение выдачи заказа представителем
- формированием отчета представителем

2. Варианты использования

2.1. Диаграмма вариантов использования для роли «Клиент»



Рис. 1. Диаграмма вариантов использования для роли «Клиент»

- Клиент регистрируется
- Клиент просматривает каталог и выбирает товар
- Клиент выбирает способ получения товара - самовывоз или доставка

2.2. Диаграмма вариантов использования для роли "Администратор"

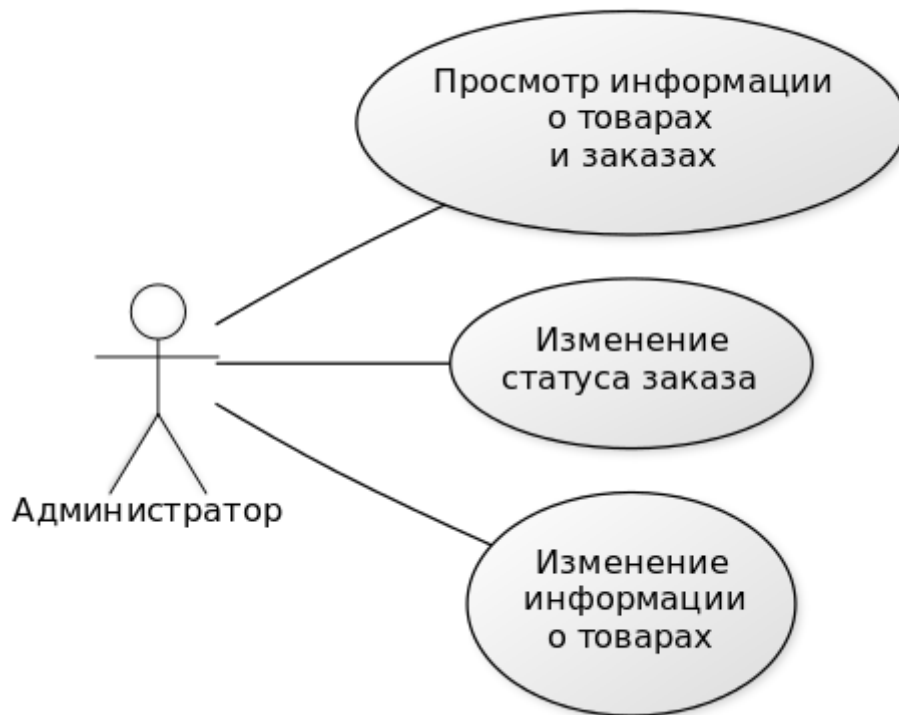


Рис. 2. Диаграмма вариантов использования для роли «Администратор»

- Администратор заходит в свою учетную запись
- Администратор просматривает наличие товара и редактирует товары
- Администратор обновляет статусы заказов

2.3. Диаграмма вариантов использования для роли "Курьер"



Рис. 3. Диаграмма вариантов использования для роли «Курьер»

- Курьер берет заказ и при осуществлении доставки изменяет статус заказа на "Доставлено" (Delivered)

3. Моделирование

3.1. Диаграмма для роли "Клиент"

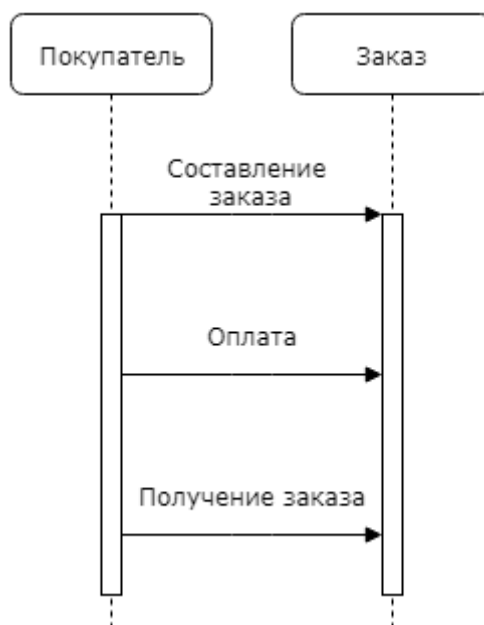


Рис. 4. Диаграмма для роли "Клиент"

3.2. Диаграмма для роли "Администратор"

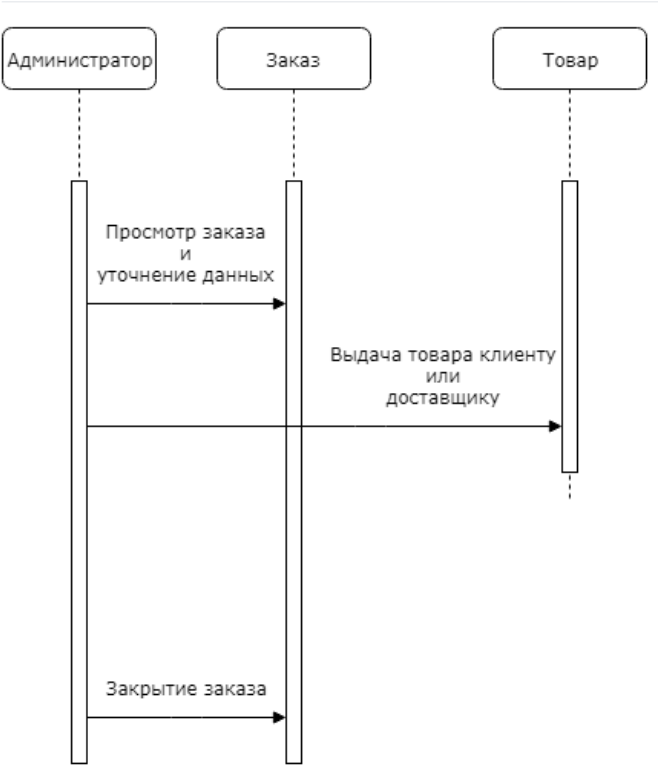


Рис. 5. Диаграмма для роли "Администратор"

3.3. Диаграмма для роли "Курьер"



Рис. 6. Диаграмма для роли "Курьер"

3.4. Статическая модель предметной области

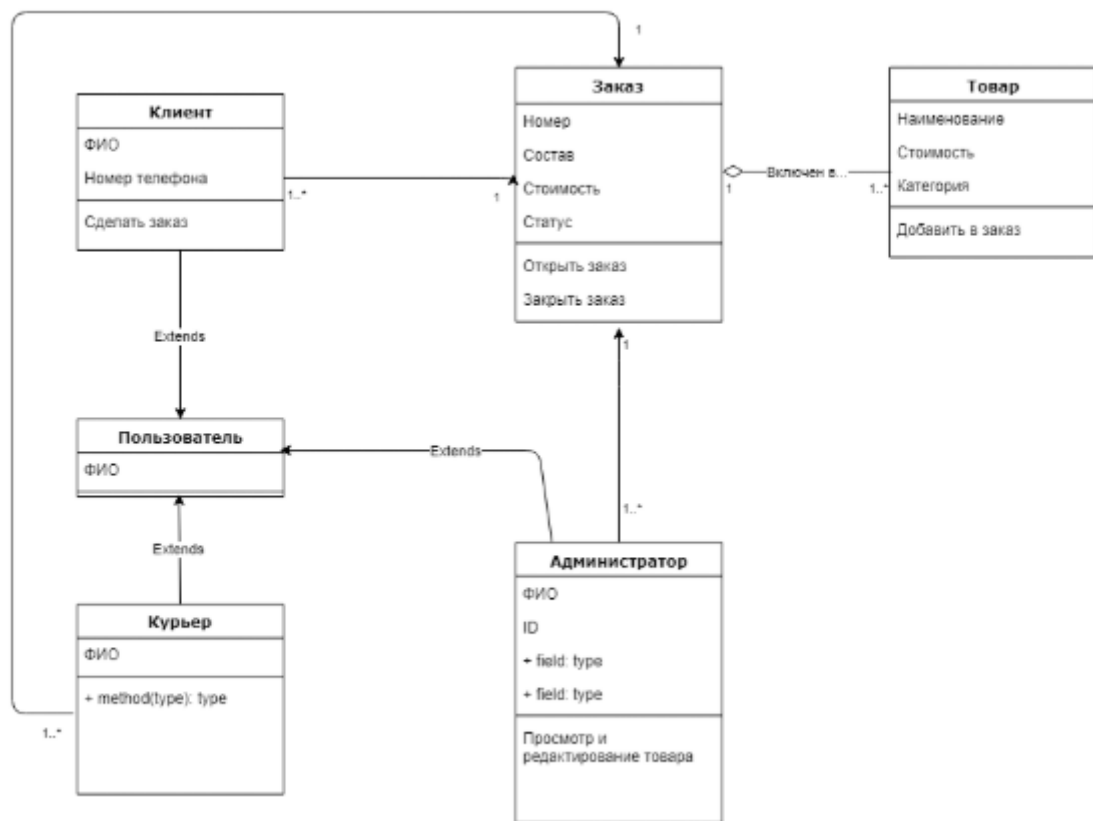


Рис.7. Статическая модель предметной области

4. Слой бизнес логики

4.1. Проектирование

Бизнес логика проектировалась в соответствии с шаблоном "Модель предметной области".

4.2. Реализация

- Слой бизнес логики состоит из базового класса **User**, и 3 классов наследуемых от него – **Administrator**, **Courier** и **Client**. Также существует класс **Order** представляющий собой сущность бизнес-процесса заказа.
- Класс **User** состоит из строковых полей **FirstName**, **Surname**, **password** и целочисленного поля **id**. Так же этот класс содержит абстрактные методы для получения и установки этих параметров.
- Класс **Client** помимо наследуемых полей и методов, содержит

строковые поля Email и TeleNumber, и метод CreateOrder - создать заказ, который принимает в качестве параметра список товаров и способ доставки.

- Класс Adminstator помимо наследуемых полей и методов, содержит методы по изменению статусов заказов в соответствии с вариантами использования, и методы по смене статусов процесса обмена и установки цены и категории на полученную игру.
- Класс Courier помимо наследуемых полей и методов, содержит методы по изменению статусов заказов в соответствии с вариантами использования.
- Класс Order содержит численные поля id, cost, поле списка товаров ItemList.
- Класс Order_Status содержит поле статуса заказа Status типа Order_status.

5. Слой источников данных

Применялся шаблон Репозиторий. Назначение временного хранилища: посредник между уровнями области определения (хранилище) и распределения данных. Для хранения элементов каждой сущности заводились коллекции (ArrayList). Репозиторий инкапсулирует набор объектов, сохраняемых в хранилище данных, и операции выполняемые над ними, обеспечивая более объектно-ориентированное представление реальных данных.

Классы - репозитории были реализованы для всех сущностей – участников основных бизнес - процессов: Админ, Клиент, Курьер, Товар, Заказ. Все репозитории были созданы с применением паттерна проектирования Singleton.

Для реализации слоя данных было принято решение использовать СУБД PostgreSQL. Подключение реализовано в классе db_utils, который реализовывал единую точку подключения к БД посредством соединения при помощи JDBC драйвера, тем самым предоставляя единые данные для всех преобразователей (“Data Mapper” - паттерн, который выступает в роли посредника для двунаправленной передачи данных между постоянным хранилищем данных (часто, реляционной базы данных) и представления данных в памяти (слой домена, то что уже загружено и

используется для логической обработки)).

В целях изучения работы мапперов для сущностей Item и Client мапперы были написаны самостоятельно.

6. Выводы

В ходе данной работы была разработана система продажи цветов в магазине. В процессе разработки были изучены архитектурные шаблоны и шаблоны проектирования для всех слоев программного обеспечения.

Рассматривая подробнее разработанную систему, можно сказать, что для проектирования слоя бизнес-логики был использован шаблон "Модель предметной области", для слоя источника данных были использованы шаблоны "Преобразователь данных" и "Репозиторий", для хранения данных используется реляционная СУБД PostgreSQL.