

Отчет
по курсовой работе
Дисциплина
«Архитектуры проектирования ПО»

Тема:
«Система создания и обработки заявок интернет провайдера»

Выполнил студент группы: 13541/3: Покатило П.А.

Преподаватель: Зозуля А.В.

Оглавление

Постановка задачи	3
Задание.....	3
Функциональные требования	3
Бизнес-процессы.....	3
Варианты использования	4
Заказчик	4
Оператор Колл-центра	5
Оператор Технического отдела	6
Оператор отдела монтажа.....	6
Статическая объектная модель предметной области.....	7
Диаграмма классов	7
Динамическая объектная модель предметной области	8
Заказчик	8
Оператор КЦ	9
Оператор ТО	9
Оператор МО	10
Слой бизнес логики	10
Проектирование	10
Реализация.....	10
Сущности.....	10
Слой источников данных	10
Состав	10
База данных	11
Сервисный слой и слой представления	11
Реализация сервисного слоя	11
Интеграция	11
Реализация.....	11
Http-сервер.....	12
Реализация слоя представления	12
Тестирование.....	12
Выводы	12

Постановка задачи

Задание

Информационная система для обработки заявок провайдера интернета

В системе фигурируют следующие типы участников:

- Заказчик
- Колл-центр
- Технический отдел
- Монтажный отдел

Функциональные требования

Для каждого участника определены следующие требования:

- Заказчик
 - Создает запрос в личном кабинете и передает данные о подключении (адрес, тарификация)
 - Принимает информацию о заявке на подключение и осуществляет оплату или отмену в случае несогласия (двойная опция)
 - При оплате контролирует монтажные работы
- Колл-центр
 - Принимает заявки и информацию от заказчика
 - Создает виртуальный документ заявки
 - Назначает на обработку заявки сотрудников технического отдела
 - Создает документ с итоговой стоимостью и передает Заказчику
 - Создает заявку на монтажные работы
 - Назначает на монтажные работы сотрудников монтажного отдела
- Технический отдел
 - Вносит информацию о необходимом оборудовании и коммуникациях для осуществления подключения и сопутствующую информацию
- Монтажный отдел
 - Производит монтаж оборудования и вносит отчетную информацию в заявку

Бизнес-процессы

Расчет стоимости подключения интернета

1. Создание заявки в личном кабинете Заказчиком
2. Регистрация заявки в Колл-центре
3. Обработка заявки в техническом отделе

4. Передача заказчику расчетных документов на оплату
5. Оплата или отказ от заявки Заказчиком

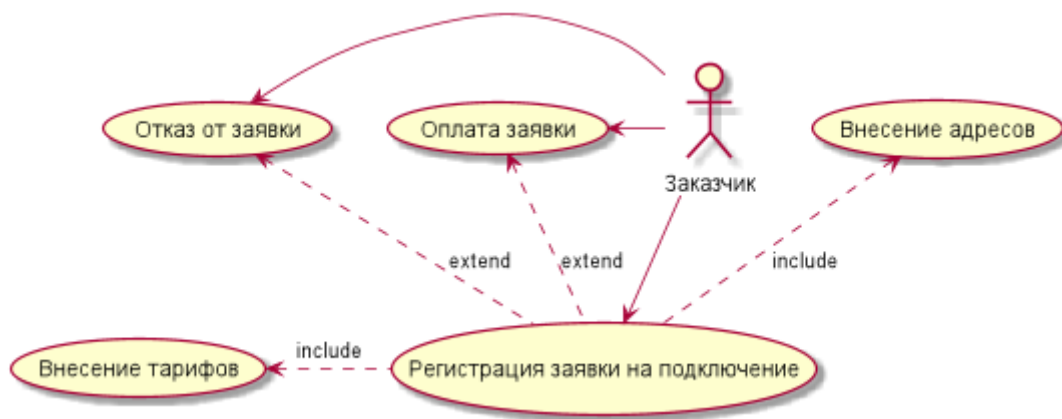
Монтажные работы

1. Назначение операторов МО заявки в Колл-центре
2. Установка оборудования
3. Внос информации о процессе установки в заявку
4. Подтверждение завершения обработки заявки в Колл-центре
5. Закрытие заявки

Варианты использования

Заказчик

В личном кабинете системы **заказчик** видит **заявки**, которые создал



Регистрация заявки

1. **Заказчик** авторизуется в **Личном кабинете**
2. **Заказчик** создает новую **заявку**
3. **Заказчик** вносит в заявку следующие данные:
 - **Адрес** подключения
 - **Тариф** подключения соответственно адресу
4. **Заказчик** регистрирует **заявку**, которая попадает в **Колл-центр**

Оплата заявки

1. **Заказчик** авторизуется в **Личном кабинете**
2. **Заказчик** оплачивает заявку на подключение к интернету
 - При условии, что заявка была обработана и итоговый платежный документ составлен

Отказ от заявки

1. **Заказчик** авторизуется в **Личном кабинете**

2. **Заказчик** на момент оплаты решает отказаться от **заявки** и производит **отказ**

Согласие на монтаж

1. **Заказчик** производит оплату заявки на подключение
2. **Заказчик** контролирует установку оборудования

Оператор Колл-центра

В личном кабинете системы **оператор** видит **заявки**, куратором которых является, а также **заявки** ожидающие назначения операторов



Регистрация заявки на подключение

1. **Оператор КЦ** авторизуется в **Личном кабинете**
2. **Оператор КЦ** регистрирует поступившую заявку, для этого он назначает для работы с ней следующих операторов из **базы**:
 - **Операторов ТО**
 - **Оператор КЦ** Автоматически назначается куратором **заявки**

Регистрация заявки на монтажные работы

1. **Оператор КЦ** авторизуется в **Личном кабинете**
2. **Оператор КЦ** создает **заявку на монтажные работы** в соответствии с оплаченной **заявкой на подключение интернета**
3. **Оператор КЦ** назначает для работы с ней следующих операторов из **базы**:
 - **Операторов МО**

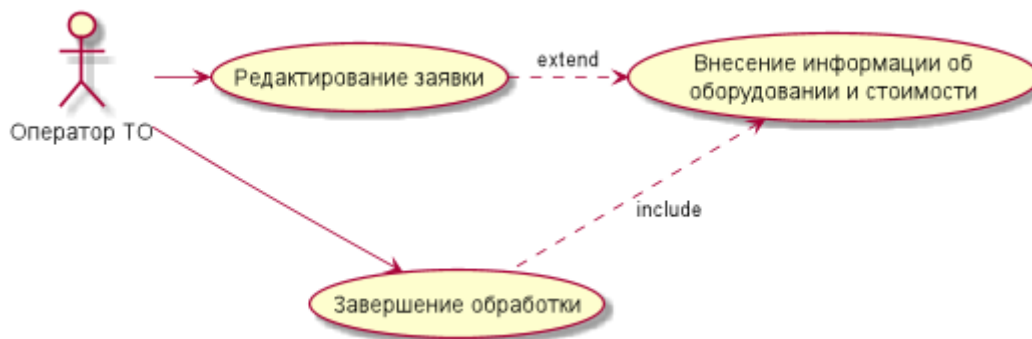
Заккрытие заявки

1. **Оператор КЦ** авторизуется в **Личном кабинете**
2. **Оператор КЦ** регистрирует завершение обработки **заявки на монтажные работы**

3. **Оператор КЦ** завершает обработку заявки и помещает ее в архив

Оператор Технического отдела

В **личном кабинете** системы **оператор** видит **заявки**, на обработку которых он был назначен

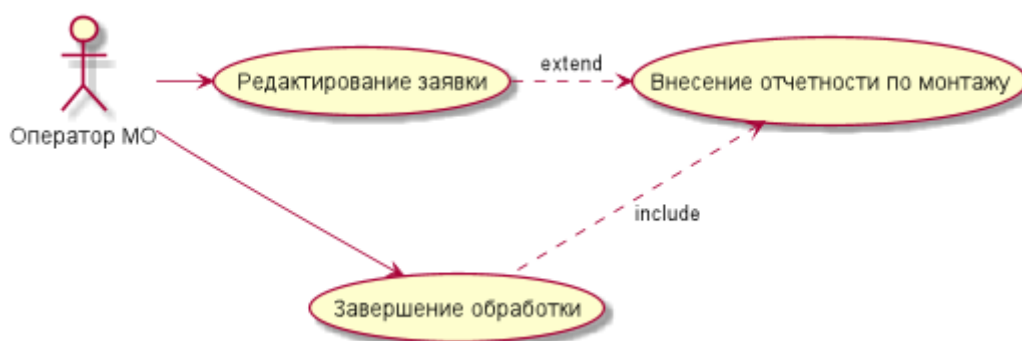


Редактирование и закрытие заявки

1. **Оператор ТО** авторизуется в **Личном кабинете**
2. **Оператор ТО** просматривает список **заявок**, на которые был назначен
3. **Оператор ТО** выбирает нужную **заявку**
4. **Оператор ТО** вносит информацию об оборудовании и его цене в необходимую **заявку**
5. **Оператор ТО** после подтверждения внесения изменений автоматически закрывает заявку

Оператор отдела монтажа

В **личном кабинете** системы **оператор** видит **заявки**, на обработку которых он был назначен



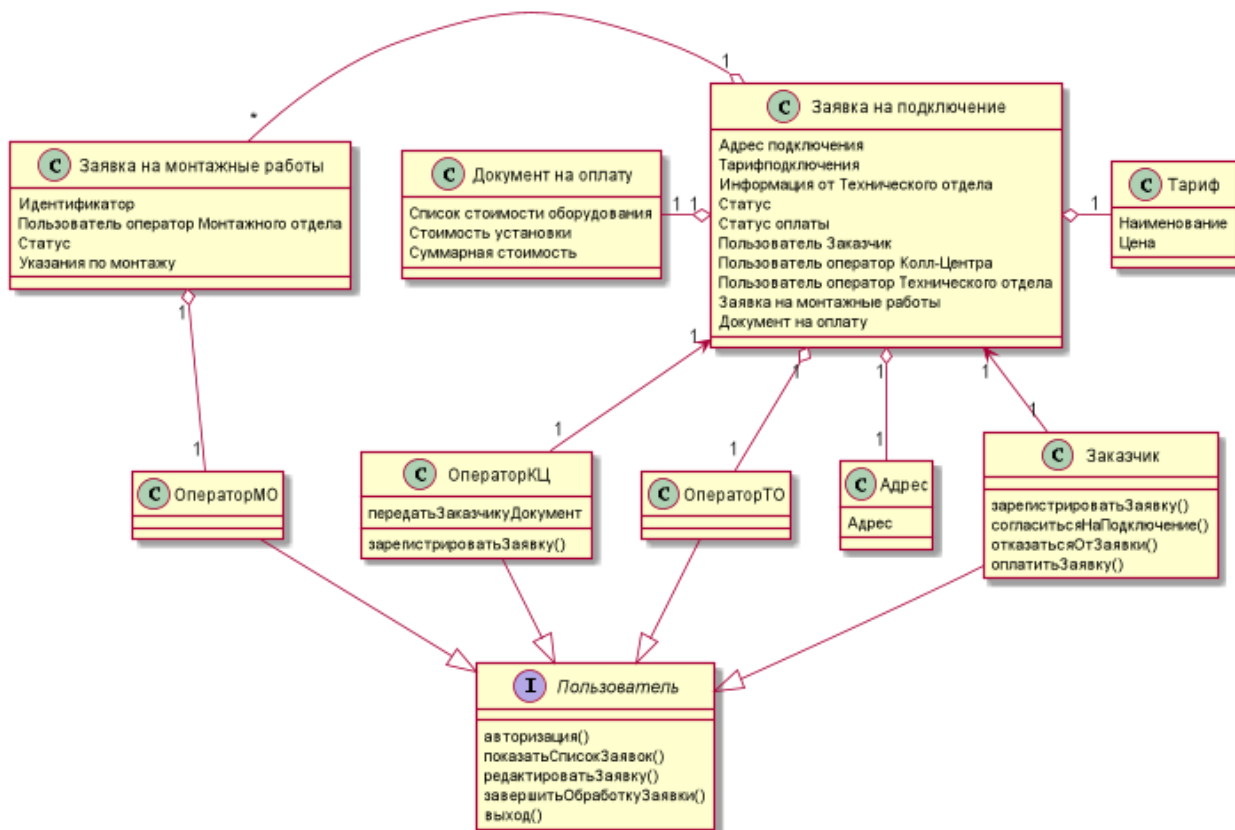
Редактирование и закрытие заявки

1. **Оператор МО** авторизуется в **Личном кабинете**
2. **Оператор МО** просматривает **список заявок**, на которые был назначен
3. **Оператор МО** выбирает нужную **заявку**
4. **Оператор МО** изучает заявку и производит монтаж
5. **Оператор МО** вносит отчетность по монтажу оборудования в необходимую **заявку**

6. **Оператор МО** после подтверждения внесения изменений автоматически закрывает заявку

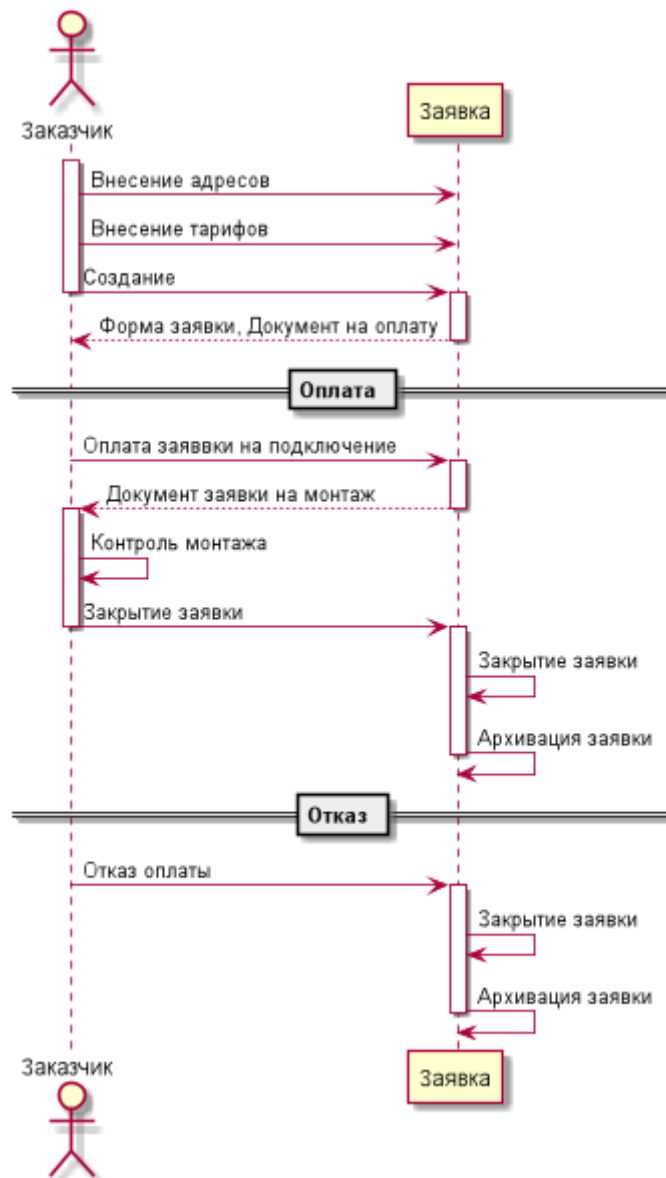
Статическая объектная модель предметной области

Диаграмма классов

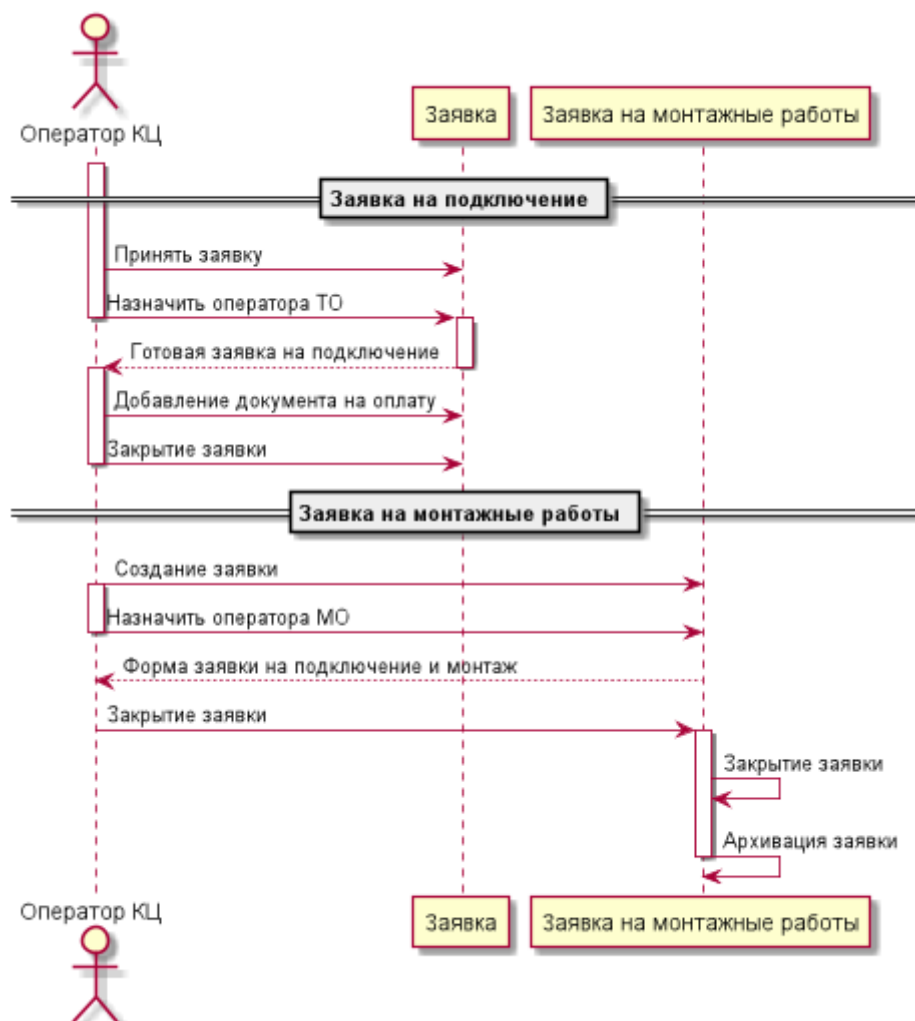


Динамическая объектная модель предметной области

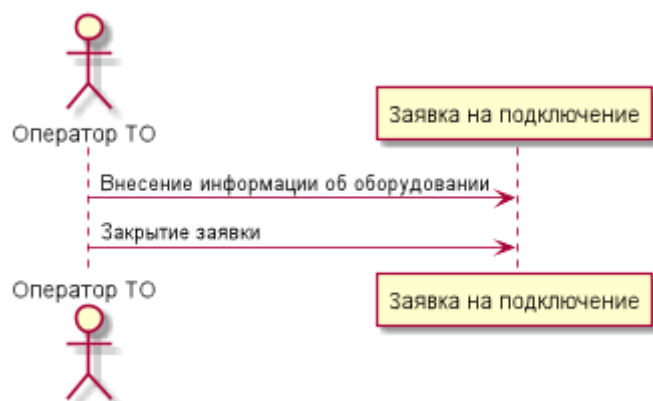
Заказчик



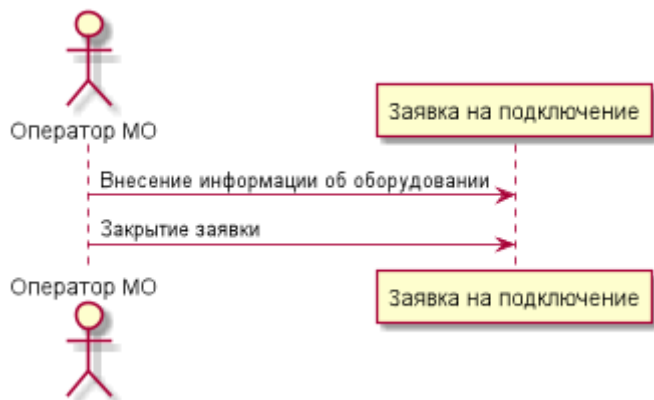
Оператор КЦ



Оператор ТО



Оператор МО



Слой бизнес логики

Проектирование

Сущности бизнес логики соответствуют шаблону "Модель предметной области"

Реализация

Реализация бизнес логики соответствует динамической объектной модели предметной области. Также присутствуют классы, упрощающие взаимодействие с состояниями системы типа ModelTypes

Сущности

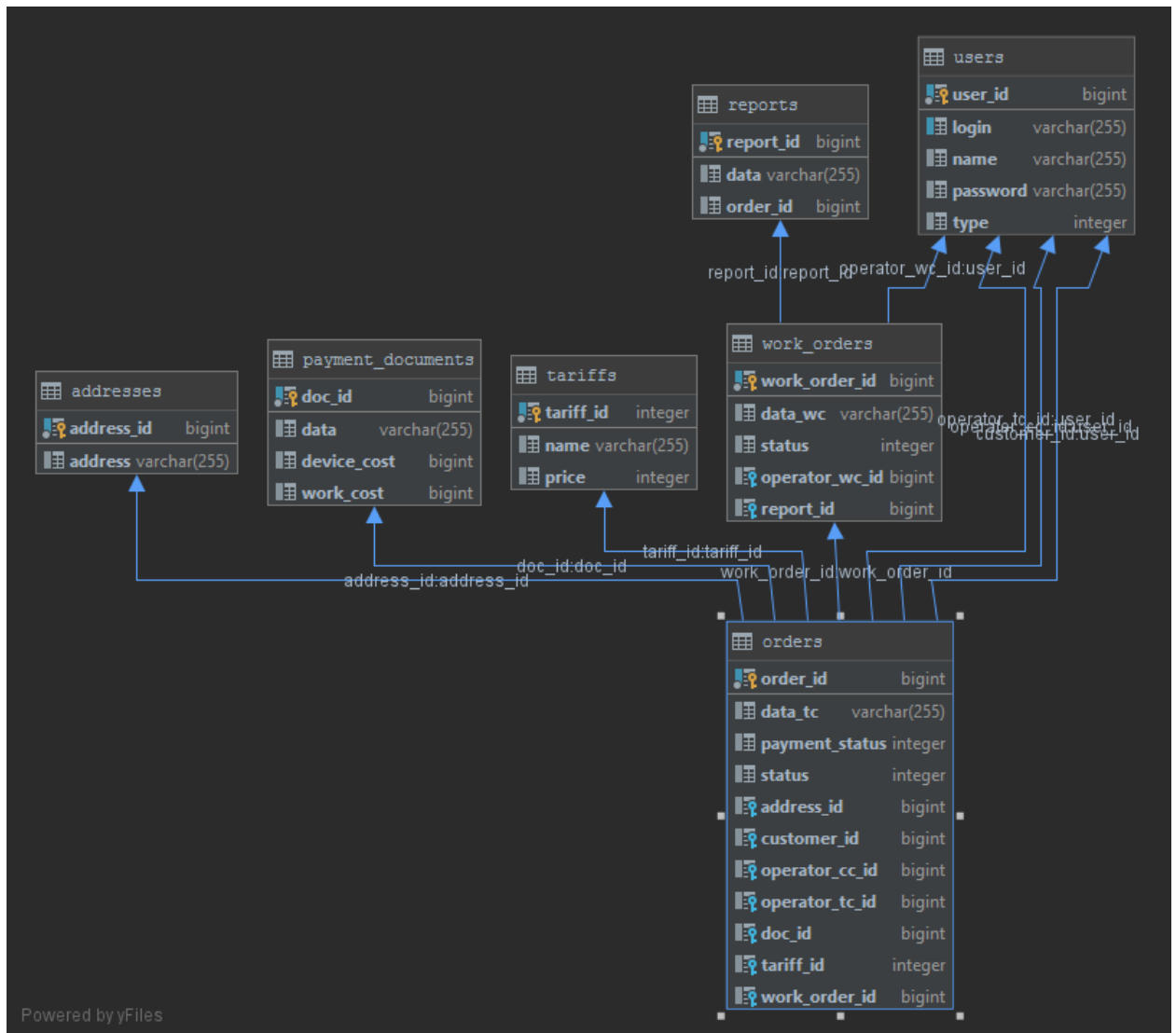
- Order - заявка, основной контейнер и документ
- WorkOrder - заявка на монтажные работы
- User - пользователь системы различного типа
- PaymentDocument - документ на оплату
- Tariff - тариф
- Addresss - адрес подключения
- ReportWC - отчет монтажного отдела

Слой источников данных

Состав

В промежуточной версии программы использовался паттерн Репозиторий, который был модифицирован для работы с базой данных и представляет собой прослойку между БД и бизнес-логикой. Для использования в собственном http сервере был написан mapper для существующих тарифов, для остальной части использовался ORM Hibernate с аннотациями JPA.

База данных



Сервисный слой и слой представления

Реализация сервисного слоя

В реализации сервисного слоя совместно с контроллерами использовался паттерн проектирования Адаптер. Используя унифицированный интерфейс `MarkedUser`, хранящийся в классе `CurrentUser`, удалось переиспользовать контроллеры и организовать единоместное хранение и унифицированное обращение к текущему пользователю и бизнес логике, независимо от типа пользователя.

Интеграция

Реализация

Конфиденциальные данные пользователя читаются из .json файла следующего формата и с их помощью реализуется подключение к бд

Http-сервер

Простейший http-сервер по адресу <http://localhost:8000/tariff> выдает json массив актуальных тарифов провайдера, которые получает из БД с помощью маппера

Реализация слоя представления

Разработанный на JavaFX 8 графический интерфейс содержит следующие формы:

- Login - форма авторизации пользователей
- CCOOrdersView - личный кабинет оператора КЦ
- TCOOrdersView - личный кабинет оператора ТО
- WCOOrdersView - личный кабинет оператора МО
- CustomerOrdersView - личный кабинет заказчика
- AssignUser - назначение операторов на заявку
- OrderForm - документ заявки и общий инструмент редактирования
- OrderCreationForm - форма создания заявки
- PayDocCreationForm - форма создания документа на оплату

Тестирование

Тестирование осуществлялось с помощью фреймворка JUnit. Тесты представляют из себя варианты прохождения бизнес-процессов. Графический интерфейс тестировался вручную.

Выводы

В ходе данной работы была разработана система обработки и регистрации заявок для некоторого абстрактного провайдера. Материал, полученный в ходе курса Архитектуры проектирования программного обеспечения, помог создать некоторую абстрактную, конфигурируемую систему, слои которой четко разделены и легко подвержены модификации без вреда для остальных слоев.

Итоговая система использует "Модель предметной области" как основной шаблон проектирования бизнес-логики. Источники данных используют шаблоны "Репозиторий" и "Преобразователь данных"(Data mapper), которые взаимодействуют с СУБД PostgreSQL. Сервисный слой реализует шаблон "Адаптер". Графический интерфейс использует xml верстку и создан с помощью библиотеки JavaFX.

Система работоспособна и протестирована. Однако, при дальнейшем развитии системы требуется более тщательное тестирование зависимостей в логике слоя представления на уровне языка Java. Для полноценного улучшения системы можно модифицировать слой представления, а также улучшить http сервер для выдачи полноценной версии API для работы с данной системой и интеграции ее в другие проекты.