

МИНОБРНАУКИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.
ЧЕРНЫШЕВСКОГО»

Кафедра теории функций и стохастического анализа

Проектирование и разработка информационной системы для ателье

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ (ДИПЛОМНОЙ) РАБОТЫ

студентки _____ 4 _____ курса _____ 451 _____ группы
направления _____ 38.03.05 - Бизнес -информатика _____
код и наименование направления
_____ механико-математического факультета _____
наименование факультета, института, колледжа
_____ Шведчиковой Александры Вадимовны _____
фамилия, имя, отчество

Научный руководитель

_____ доцент, д. ф-м. наук _____
должность, уч. степень, уч. звание

_____ дата, подпись

_____ Сидоров С. П. _____
инициалы, фамилия

Заведующий кафедрой

_____ доцент, д. ф-м. наук _____
должность, уч. степень, уч. звание

_____ дата, подпись

_____ Сидоров С. П. _____
инициалы, фамилия

Саратов 2020

ВВЕДЕНИЕ

В современном мире развитие сфер жизнедеятельности человека напрямую зависит от развития информационных систем. Уже невозможно представить нашу жизнь без использования информационных технологий.

Под информационной системой понимается хранилище информации, снабженное процедурами ввода, поиска, размещения, вывода информации и предназначенное для того, чтобы своевременно обеспечивать людей необходимой информацией. Главная задача информационной системы — обслуживание клиентов, поэтому её основная работа заключается в обслуживании двух встречных потоков новой информации: ввод новой информации и выдачи текущей информации по запросам. Информационные системы являются основными помощниками человека при работе с огромным потоком информации.

Цель данной работы заключается в разработке информационной системы ателье и проектирование графического интерфейса пользователя с применением средств высокоуровневого языка программирования Python и библиотеки PySide.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ предметной области «Ателье».
2. Построить UML-диаграммы такие, как диаграмма прецедентов, диаграмма деятельности и диаграмма последовательности, а также ER-диаграмму.
3. Спроектировать базу данных в SQLite.
4. Разработать графический интерфейс с помощью языка программирования Python и библиотеки PySide.

1 ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Работа состоит из трех разделов. Некоторые разделы содержат несколько подразделов:

1. Проектирование информационной системы средствами UML
 - а) Описание предметной области;
 - б) Описание технологии UML;
 - в) Диаграмма вариантов использования;
 - г) Поток событий;
 - д) Диаграмма последовательности;
 - е) Диаграмма деятельности.
2. Разработка базы данных для информационной системы
 - а) Система управления базами данных SQLite;
 - б) ER-диаграмма;
 - в) Описание таблиц базы данных.
3. Разработка графического интерфейса для информационной системы
 - а) Язык программирования Python и библиотека PySide;
 - б) Реализация графического интерфейса.

В первом разделе приводится описание предметной области, которая представляет собой ателье. Также описано средство проектирование информационной системы UML и построение диаграмм вариантов использования, последовательности и деятельности для предметной области «Ателье».

Информационная система «Ателье» позволяет упорядочить работу фирмы: информация о заказах, данные клиентов и т.д. хранятся в электронном виде на компьютере и легко могут быть найдены и воспроизведены на бумаге.

Ателье предоставляет два вида услуг: пошив с нуля и ремонт одежды. Ателье работает с разными видами изделий: брюки, пиджаки, юбки, платья и другими изделиями. Информационная система «Ателье» должна хранить информацию о клиентах и их заказах. Каждый заказ может иметь в себе несколько позиций. Для каждой позиции в заказе указывается вид услуги, вид изделий, цена, количество, а также дополнительная информация. При оформлении заказа указывается дата заказа и клиент. Каждому заказу назначается мастер, который будет его выполнять. В ателье имеется система скидок для постоянных клиентов в 5%, 10% и 15%. Для удобства в информа-

ционной системе «Ателье» можно отслеживать статус выполнения заказа.

Специфической особенностью ателье является индивидуальная работа с каждым клиентом. С одним клиентом работает один мастер от начала — с момента передачи раскроенного материала от закройщика, до выдачи готового изделия клиенту. Это позволяет контролировать качество пошива, повышает заинтересованность в получении конечного результата. У ателье есть постоянные клиенты, которые предпочитают делать заказы у конкретных мастеров.

Во втором разделе приводится описание СУБД SQLite, построение ER-диаграммы для информационной системы «Ателье» и разработка базы данных.

SQLite — компактная встраиваемая СУБД. Исходный код библиотеки передан в общественное достояние. Слово «встраиваемый» означает, что SQLite не использует парадигму клиент-сервер, то есть движок SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а представляет собой библиотеку, с которой программа компонуется, и движок становится составной частью программы. Таким образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite. Такой подход уменьшает накладные расходы, время отклика и упрощает программу. SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, на котором исполняется программа. Простота реализации достигается за счёт того, что перед началом исполнения транзакции записи весь файл, хранящий базу данных, блокируется.

Преимуществами SQLite являются:

1. Файловая структура - вся база данных состоит из одного файла, поэтому её очень легко переносить на разные машины
2. Используемые стандарты - хотя может показаться, что эта СУБД примитивная, но она использует SQL. Некоторые особенности опущены (RIGHT OUTER JOIN или FOR EACH STATEMENT), но основные все-таки поддерживаются.
3. Отличная при разработке и тестировании - в процессе разработки приложений часто появляется необходимость масштабирования. SQLite пред-

лагает всё что необходимо для этих целей, так как состоит всего из одного файла и библиотеки написанной на языке С.

К недостаткам SQLite можно отнести следующее:

1. Отсутствие системы пользователей - более крупные СУБД включают в свой состав системы управления правами доступа пользователей. Обычно применения этой функции не так критично, так как эта СУБД используется в небольших приложениях.
2. Отсутствие возможности увеличения производительности - опять, исходя из проектирования, довольно сложно выжать что-то более производительное из этой СУБД.

В проектировании структуры базы данных применяется семантическое моделирование. Семантическое моделирование представляет собой моделирование структуры данных, опираясь на смысл этих данных. В качестве инструмента семантического моделирования используются различные варианты диаграмм сущность-связь (ER - Entity-Relationship).

ER-диаграмма содержит информацию о сущностях системы и способах их взаимодействия, включает идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их связей с другими объектами (отношений).

Для наглядности связей между таблицами базы была создана ER-диаграмма, на которой изображены сущности и их атрибуты. Сущность изображается в виде прямоугольника, вверху которого располагается имя сущности, набранные полужирным шрифтом (например, Заказы). В прямоугольнике ниже перечислены атрибуты сущности; атрибуты ER-диаграмм, отмеченные ключом, являются ключевыми (так Код заказа — ключевой атрибут сущности Заказы, остальные атрибуты ключевыми не являются).

ER-диаграмма разрабатываемой информационной системы состоит из следующих сущностей:

1. Виды услуг;
2. Виды изделий;
3. Скидки;
4. Клиенты;
5. Категории;

6. Мастера;
7. Заказы;
8. Содержимое заказа;
9. Виды статусов;
10. Статус-содержимое.

ER-диаграмма «Ателье» создана с помощью интернет-ресурса DbDesigner . В соответствии с рисунком 1 построена ER-диаграмма «Ателье».

dbdesigner.net

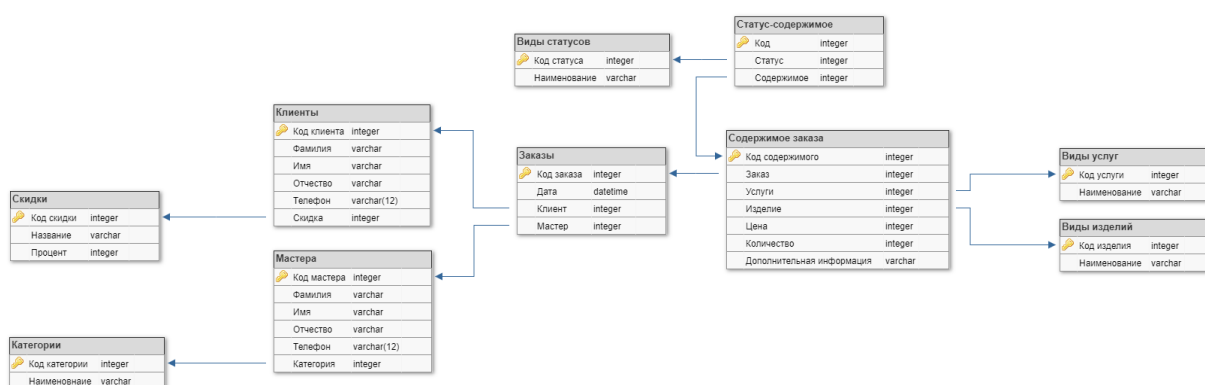


Рисунок 1 – ER-диаграмма «Ателье»

В третьем разделе приводится описание языка программирования Python с библиотекой PySide и реализация графического интерфейса. Python — высокоуровневый язык программирования с динамической типизацией, поддерживающий объектно-ориентированный, функциональный и императивный стили программирования. Это язык общего назначения, на котором можно одинаково успешно разрабатывать системные приложения с графическим интерфейсом, утилиты командной строки, научные приложения, игры, web-приложения и многое другое. Он является одним из наиболее популярных современных языков программирования. В 2019 году Python стал самым популярным языком программирования, обогнав Java на 10%.

Для реализации графического интерфейса к информационной системе был выбран именно данный язык программирования, также и за простоту

его освоения. Для реализации графического интерфейса будет использован язык программирования Python совместно с библиотекой PySide.

PySide — привязка языка Python к инструментарию Qt, совместимая на уровне API с PyQt . Проект возник в результате нежелания создателей PyQt менять лицензионную политику для своего проекта. PySide появился в августе 2009 года. Основными разработчиками PySide являются программисты Digia.

Qt является кроссплатформенным фреймворком для приложений и пользовательских интерфейсов . Он позволяет разработчикам писать приложения однократно и разворачивать их на операционных системах, не переписывая исходный код, в то время как Python является современным, динамическим языком программирования с большим сообществом разработчиков. PySide обеспечивает богатство фреймворка Qt-разработчикам благодаря объединению мощностей Qt и Python. Таким образом, Qt представляет собой первоклассную платформу для быстрой разработки программного обеспечения на языке Python, доступную на всех основных операционных системах.

PySide направлен на реализацию поддержки всех возможностей Qt, в том числе:

- QtQuick — современной технологией создания пользовательского интерфейса, особенностью которой является разделение декларативного описания дизайна интерфейса и императивной логики программирования;
- QtMobility — набора API (интерфейса программирования приложений) и фреймворков (программных платформ, определяющих структуру программной системы), ориентированных на разработку приложений для мобильных платформ.

В данной работе был сделан анализ предметной области «Ателье». На основе построенных диаграмм UML, ER-диаграммы и базы данных, написанной на языке SQL для СУБД SQLite, был разработан программный код графического интерфейса пользователя на языке программирования Python с помощью библиотеки PySide для работы с информационной системой «Ателье» .

Данные таблицы «Виды услуг» можно редактировать в соответствии с

рисунком 2.

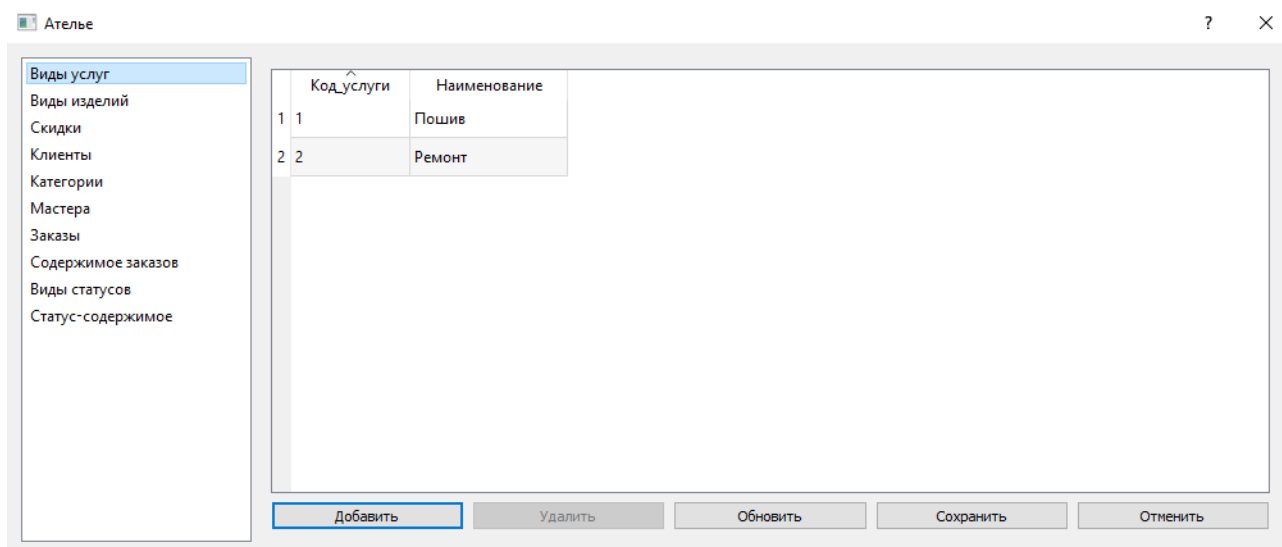


Рисунок 2 – Редактирование таблицы «Виды услуг»

Данные таблицы «Виды изделий» можно редактировать в соответствии с рисунком 3.

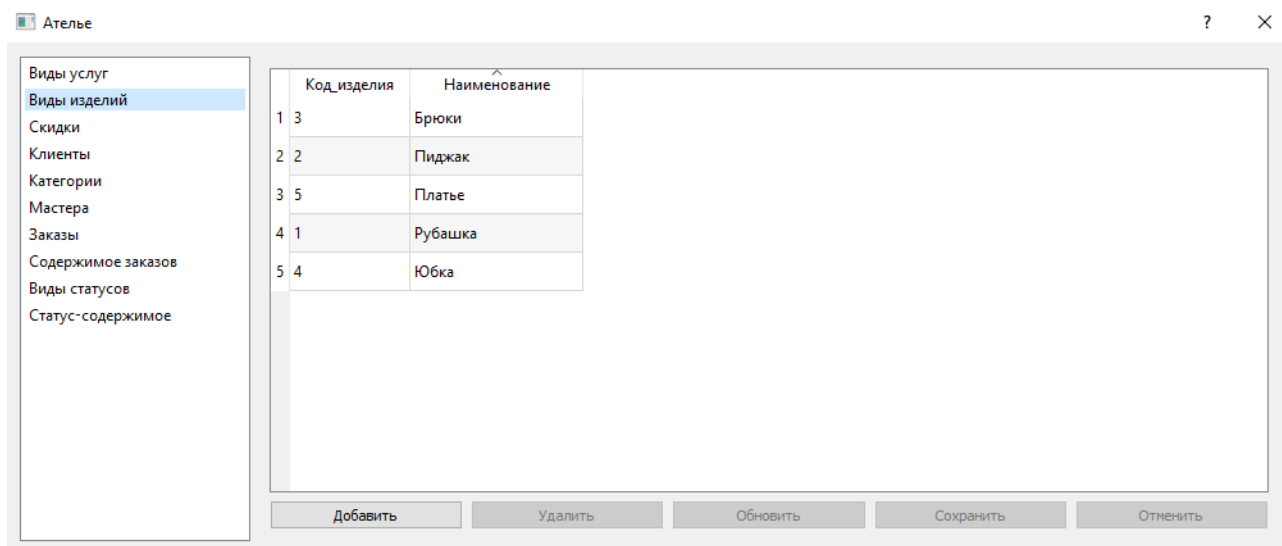


Рисунок 3 – Редактирование таблицы «Виды изделий»

Данные таблицы «Скидки» можно редактировать в соответствии с рисунком 4.

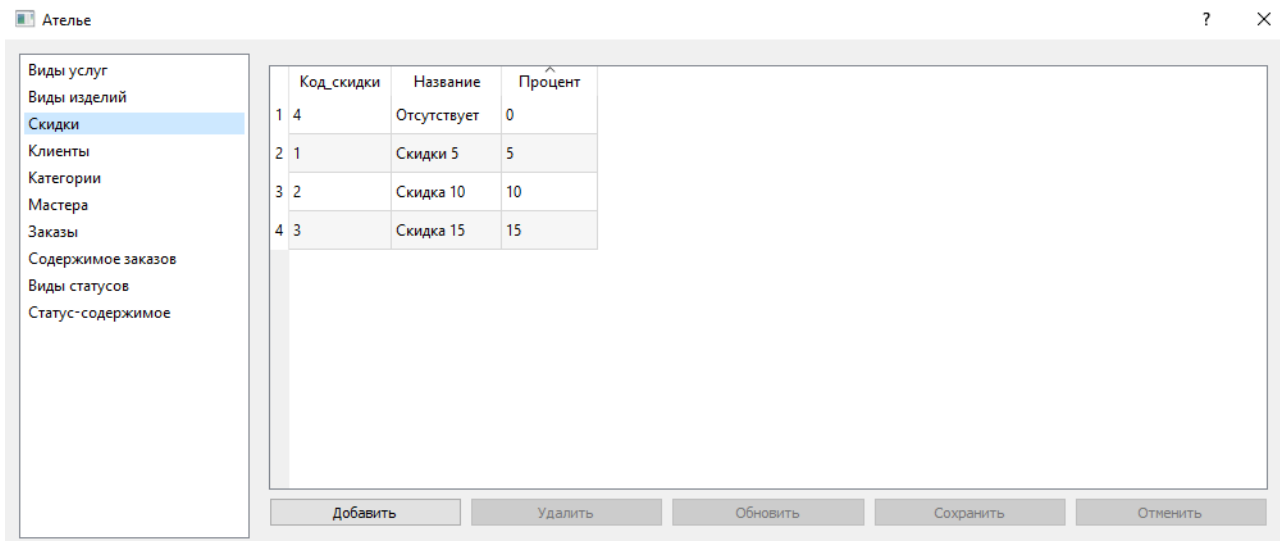


Рисунок 4 – Редактирование таблицы «Скидки»

Данные таблицы «Клиенты» можно редактировать в соответствии с рисунком 5.

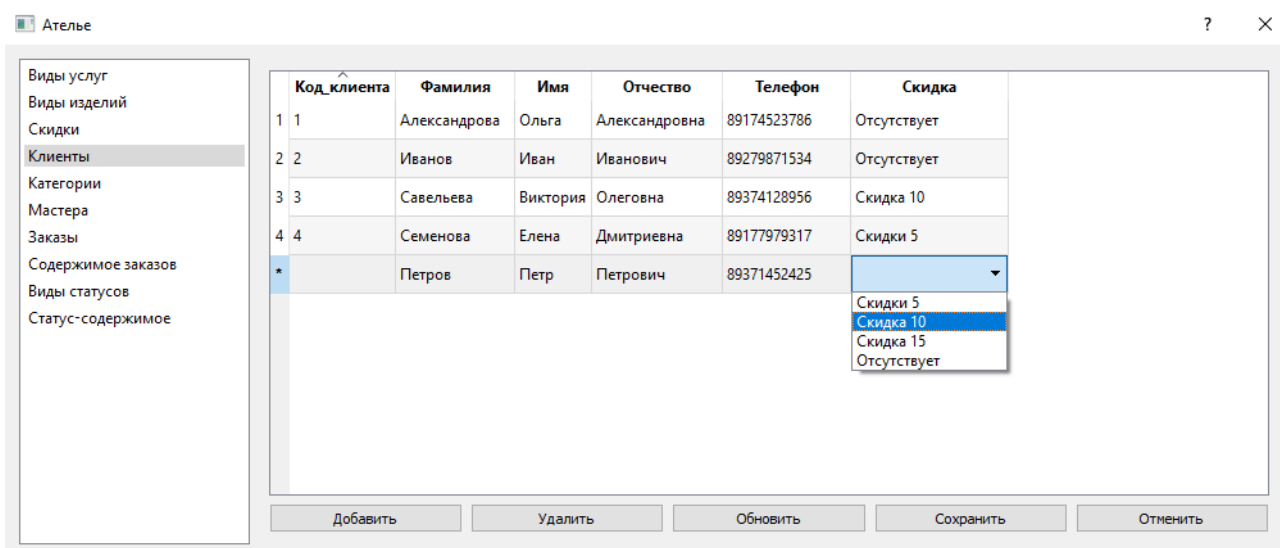


Рисунок 5 – Редактирование таблицы «Клиенты»

Данные таблицы «Категории» можно редактировать в соответствии с рисунком 6.

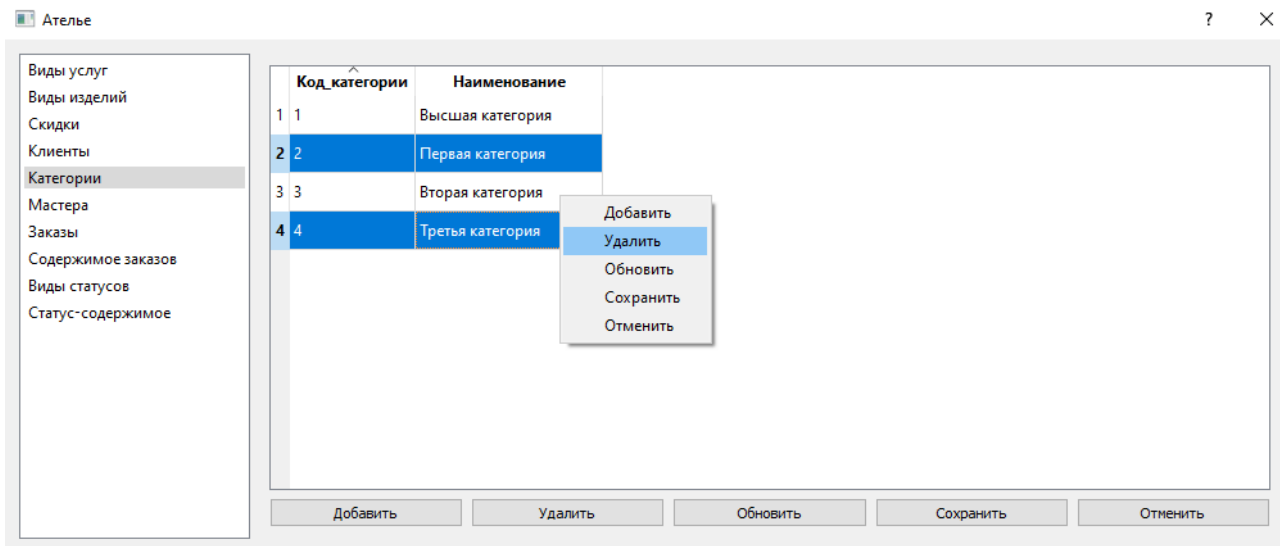


Рисунок 6 – Редактирование таблицы «Категории»

Данные таблицы «Мастера» можно редактировать в соответствии с рисунком 7.

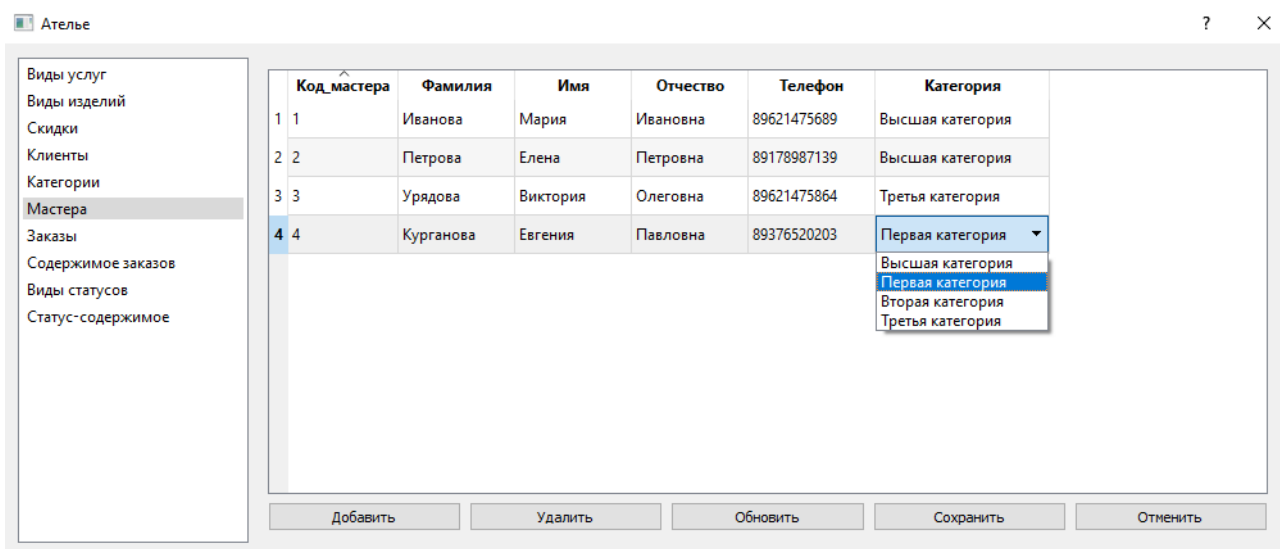


Рисунок 7 – Редактирование таблицы «Мастера»

Данные таблицы «Заказы» можно редактировать в соответствии с рисунком 8.

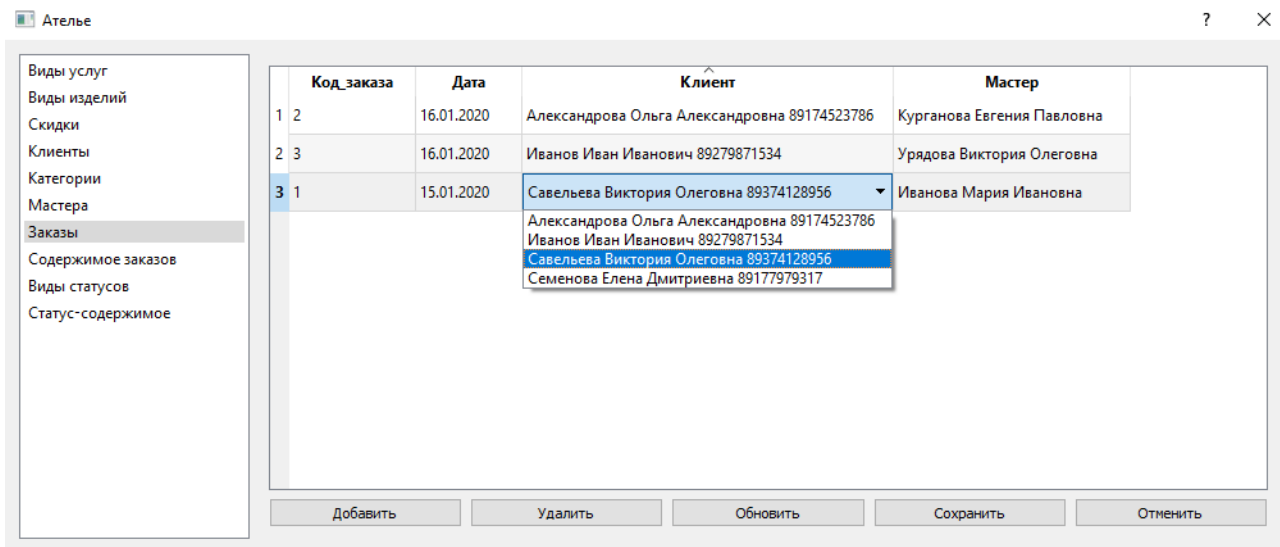


Рисунок 8 – Редактирование таблицы «Заказы»

Данные таблицы «Содержание заказов» можно редактировать в соответствии с рисунком 9.

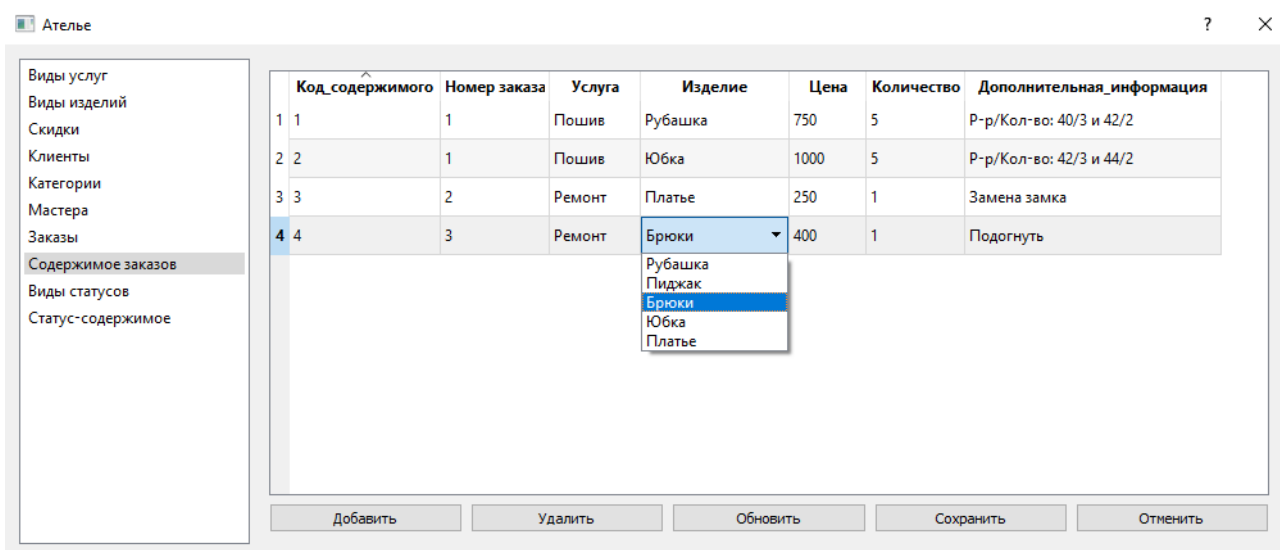


Рисунок 9 – Редактирование таблицы «Содержание заказов»

ЗАКЛЮЧЕНИЕ

Целью данной бакалаврской работы являлось проектирование информационной системы «Ателье», состоящее из нескольких взаимосвязанных частей. В ходе выполнения работы было изучено средство UML, как способ создания модели бизнес-процессов. С помощью Интернет-ресурса PlantUML были построены такие диаграммы, как:

- диаграмма вариантов использования «Ателье»;
- диаграмма последовательности для варианта использования «Добавление информации о клиентах»;
- диаграмма деятельности для варианта использования «Добавление информации о клиентах».

Также в процессе работы на основе полученных знаний при помощи системы управления базами данных SQLite была произведена разработка базы данных для информационной системы «Ателье». Для построения базы данных была создана ER-диаграмма «Ателье», которая представляет собой стандартный способ определения данных и отношений между ними. Также был реализован графический интерфейс пользователя при помощи высокоуровневого языка программирования Python и библиотеки PySide.

В ходе выполнения работы поставленная цель была достигнута, все задачи были реализованы.