

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра Математического и компьютерного моделирования

Проектирование и реализация информационной системы

для автоматизации деятельности гостиницы

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студентки 4 курса 441 группы

направление 09.03.03 — Прикладная информатика

механико-математического факультета

Вертеевой Ксении Николаевны

Научный руководитель
старший преподаватель

В. С. Кожанов

Зав. кафедрой
зав. каф., д.ф.-м.н., доцент

Ю.А. Блинков

Саратов 2019

Введение. В современном обществе своевременная обработка информации способствует совершенствованию организации производства, оперативному и долгосрочному планированию, прогнозированию и анализу хозяйственной деятельности. Каждая организация стремится минимизировать затраты времени, материальных, трудовых ресурсов в ходе своей деятельности и упростить процесс обработки информации. Эти задачи можно решить с использованием автоматизированных информационных систем.

Использование баз данных и информационных систем становится неотъемлемой составляющей функционирования преуспевающих организаций. В связи с этим большую актуальность приобретает освоение принципов построения и эффективного применения соответствующих технологий и программных продуктов.

Цель работы состоит в проектировании и дальнейшей реализации информационной системы для автоматизации деятельности гостиницы. Для осуществления этой цели сформулирован ряд задач:

- Моделирование UML-диаграмм;
- Создание ER-диаграммы предметной области;
- Создание базы данных;
- Написание программного кода на языке Python и подключение к нему базы данных;
- Создание графического интерфейса системы.

Основная часть. В первом разделе рассматриваются основные понятия информационной системы и описание предметной области.

Информационная система – это программный комплекс, функции которого состоят в поддержке надежного хранения информации в памяти компьютера, выполнении специфических для данного приложения преобразований информации и вычислений, предоставления для пользователя удобного и легко осваиваемого интерфейса. Обычно объемы информации, с которыми приходится иметь дело таким системам, достаточно велики, а сама информация имеет достаточно сложную структуру. Классическими примерами информационных систем являются банковские и бухгалтерские системы, системы авиационных или железнодорожных билетов, системы налоговой службы, статистические системы, системы резервирования мест в отеле и так далее.

Информационные системы организационного управления предназначены для автоматизации функций управленческого персонала как промышленных предприятий, так и непромышленных объектов.

При разработке информационной системы важнейшим процессом является ее проектирование. Проектированием информационных систем называется многоступенчатый процесс их создания и модернизации путём применения упорядоченной совокупности методологий и инструментария. Проектирование предполагает работу с пока несуществующим объектом и направлено на создание информационной системы. Этот процесс охватывает две основные области:

- проектирование объектов данных, которые будут реализованы в базе данных;
- проектирование программ, экранных форм, которые будут обеспечивать выполнение запросов к данным.

Предметом изучения является гостиница, которая может быть определена как коммерческое предприятие. Основной задачей является предоставление клиентам услуг по размещению в номерах. Каждый номер характеризуется вместимостью, комфортностью (люкс, стандарт, эконом) и ценой. Клиентами гостиницы являются различные лица, о которых администратор при помощи информационной системы собирает определенную информацию и предоставляет доступ к услугам. Сдача номера клиенту производится при наличии

свободных мест в номерах, подходящих клиенту по указанным выше параметрам. При поселении фиксируется дата поселения. При выезде из гостиницы для каждого места запоминается дата освобождения. Любой номер гостиницы имеет уникальный номер, по которому ведется учет проживающих в гостинице.

Во втором разделе рассматривается создание UML диаграмм. Существует множество технологий и инструментальных средств, с помощью которых можно реализовать оптимальный проект информационной системы, начиная с этапа анализа и заканчивая созданием программного кода системы. Одним из них является унифицированный язык объектно-ориентированного моделирования Unified Modeling Language (UML).

UML обеспечивает поддержку всех этапов жизненного цикла информационной системы и предоставляет для этих целей ряд графических средств — диаграмм. На этапе создания логической модели информационной системы предварительное проектирование осуществляется с использованием диаграмм претендентов и диаграмм активности.

Конструкции языка UML позволяют смоделировать статику (структуру) и динамику (поведение) системы. Система представляется в виде взаимодействующих программных модулей, которые реагируют на внешние события.

Преимущества использования UML - диаграмм при моделировании системы:

- Позволяет переписать исходное техническое задание в очень простой и наглядной графической форме, как совокупность простых утверждений относительно того, что делает система для пользователей.
- Предполагает формулирование требований к системе абсолютно независимо от ее реализации. Другими словами, представление использования описывает только, что делает система
- Каждый вариант использования описывает некоторое множество последовательностей действий, доставляющих значимый для пользователя результат. Однако нет описания алгоритма, а значит, нет алгоритмических ошибок.

Диаграмма прецедентов (вариантов использования) в UML — диаграмма, отражающая отношения между актерами и прецедентами, которая позволяет

описать систему на концептуальном уровне. С помощью диаграммы прецедентов выявляются основные пользователи системы и задачи, которые данная система должна решать.

На диаграмме в соответствии с рисунком 1 отражено взаимодействие между основным действующим актёром «Администратор» и его вариантами использования. Актёр имеет такие варианты использования, как: «Просмотр данных о номерах», «Оформление услуги». Администратор имеет возможность забронировать номер для клиента. Вариант использования «Бронирование номера» имеет связь включения: «Ввод данных клиента». Таким образом определяется взаимосвязь базового варианта использования с другим вариантом использования, функциональное поведение которого всегда задействуется базовым вариантом использования. Связь «include» иллюстрирует, что именно использует базовый вариант для выполнения операции. Ещё одним вариантом использования является «Изменение информации о сотрудниках» со связями включения «Удаление сотрудников из базы данных» и «Добавление нового сотрудника».

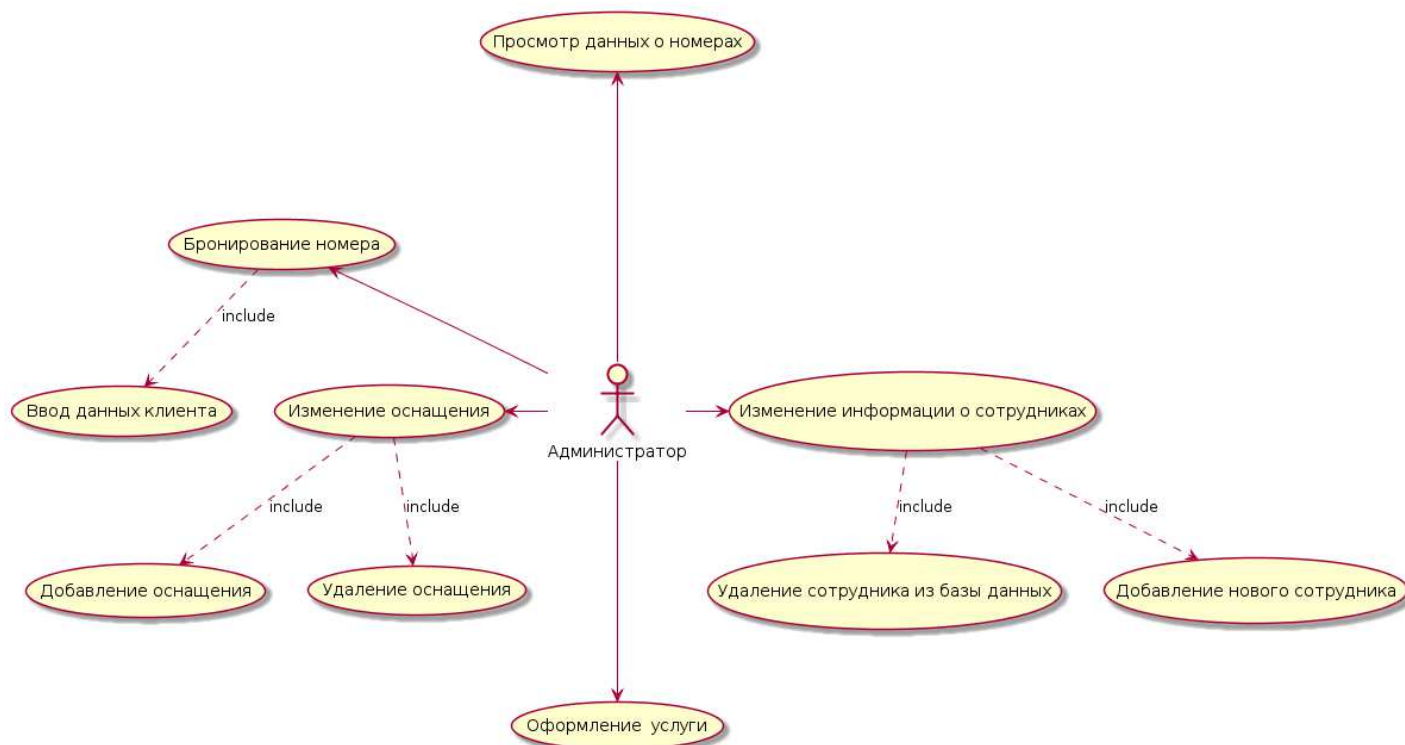


Рисунок 1 — Диаграмма претендентов

Диаграмма активности позволяет раскрыть детали алгоритмической реализации операций, выполняемых системой. Она используется, чтобы показать поток сообщений от одного вида деятельности к другому [8].

Одно из основных направлений использования диаграмм активности - отображение внутрисистемной точки зрения на прецедент. Диаграммы активности применяют для описания шагов, которые должна предпринять система после того, как инициирован прецедент.

Разработка диаграммы деятельности преследует цели:

- детализировать особенности алгоритмической и логической реализации прецедентов;
- выделить последовательные и параллельные потоки управления;
- подготовить детальную документацию для взаимодействия разработчиков системы с ее заказчиками и проектировщиками.

Графически диаграмма деятельности представляется в форме графа деятельности, вершинами которого являются состояния действия или состояния деятельности, а дугами — переходы от одного состояния действия/деятельности к другому. Каждая диаграмма деятельности должна иметь единственное начальное и единственное конечное состояния. Саму диаграмму деятельности принято располагать таким образом, чтобы действия следовали сверху вниз. В этом случае начальное состояние будет изображаться в верхней части диаграммы, а конечное — в ее нижней части.

В соответствии с рисунком 2 была реализованна диаграмма активности. В момент старта программы администратор вводит логин и пароль. Система проверят правильность введенных данных. Если данные введены неверно система возвращается в состояние «Ввод логина». Ромбом на диаграмме показано ветвление. Если авторизация пройдена успешно, аминистратор выбирает дальнейшее действие работы: клиенты, номера, оснощение, бронирование, сотрудники или услуги. При выборе режима «Клиенты» снова происходит ветвление. Администротор может добавить или удалить клиента. При выборе режима «Номера» администротор имеет возможность просмотреть информацию о конкретном номере, а именно: категорию комфортности, количество мест. Выбрав «Оснащение», администратор может менять оснащение конкретной категории комфортности, добавив или удалив необходимую

позицию. В режиме «Бронирование» дальнейший ход активностей идет в следующем порядке: выбрать клиента, выбрать номер комнаты, выбрать дату и время бронирования, забронировать номер. Если администратор нажимает «Сотрудники», он может удалить сотрудника из базы данных или добавить нового. После чего действие завершается.

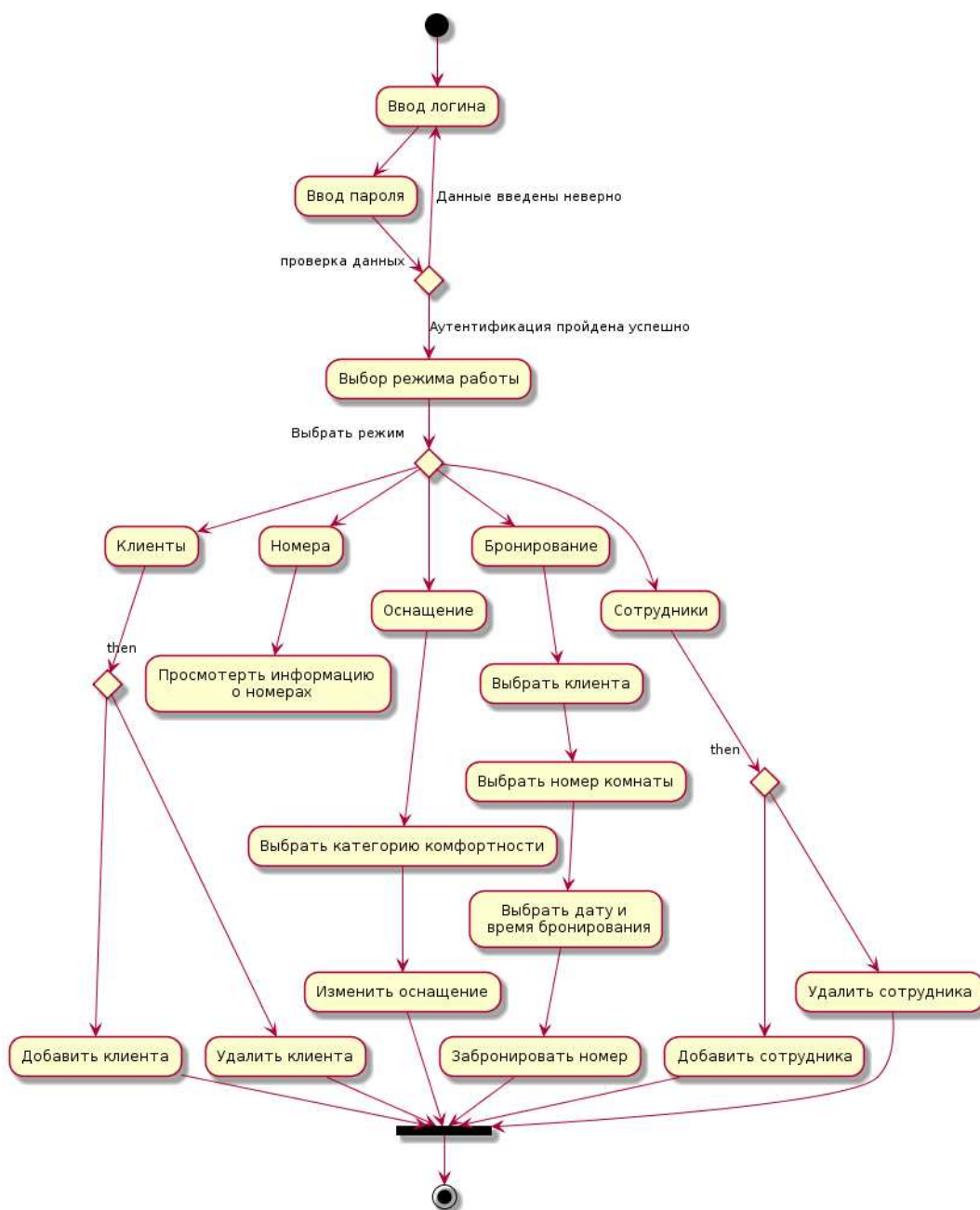


Рисунок 2 – Диаграмма активности

В третьем разделе описывается процесс проектирования базы данных. База данных – структурированный, организованный набор данных, объединен-

ный в соответствии с некоторой выбранной моделью и описывающий характеристики какой-либо физической или виртуальной системы.

Именно база данных позволяет эксплуатировать информационную систему, выполнять ее текущее обслуживание, модифицировать и развивать её при модернизации предприятия или изменении информационных потоков, законодательства и форм отчетности предприятия организации.

Использование базы данных позволяет решить такие задачи, как:

- хранение и поиск данных о сотрудниках гостиницы;
- хранение и поиск информации о клиентах гостиницы;
- хранение и поиск информации о номерах;
- хранение данных о бронировании, заселении, выезде клиентов и вычисление оплаты за проживание.

В базе данных хранятся персональные данные о клиентах и сотрудниках. С точки зрения функционирования гостиница располагает номерами с разным уровнем сервиса. Каждый номер характеризуется вместимостью, комфортностью и ценой. Номера могут быть разных типов: люкс, стандарт и эконом. Любая комната гостиницы имеет номер, по которому ведется учет клиентов, проживающих в гостинице. При бронировании номера фиксируется дата заселения и дата освобождения.

Модель данных — это графическое или текстовое представление анализа, которое выявляет данные, необходимые организации с целью достижения ее функций, целей и стратегий. При создании моделей данных используется метод семантического моделирования. Семантическое моделирование основывается на значении структурных компонентов или характеристик данных.

В качестве инструмента используются диаграммы сущность-связь (ER — Entity-Relationship). ER-диаграмма содержит информацию о сущностях системы и способах их взаимодействия, включает идентификацию объектов, важных для предметной области, свойства этих объектов (атрибутов) и связи между данными.

Для отображения полной информации по состоянию гостиницы и направлениям её деятельности создана диаграмма предметной области в соответствии с рисунком 3.

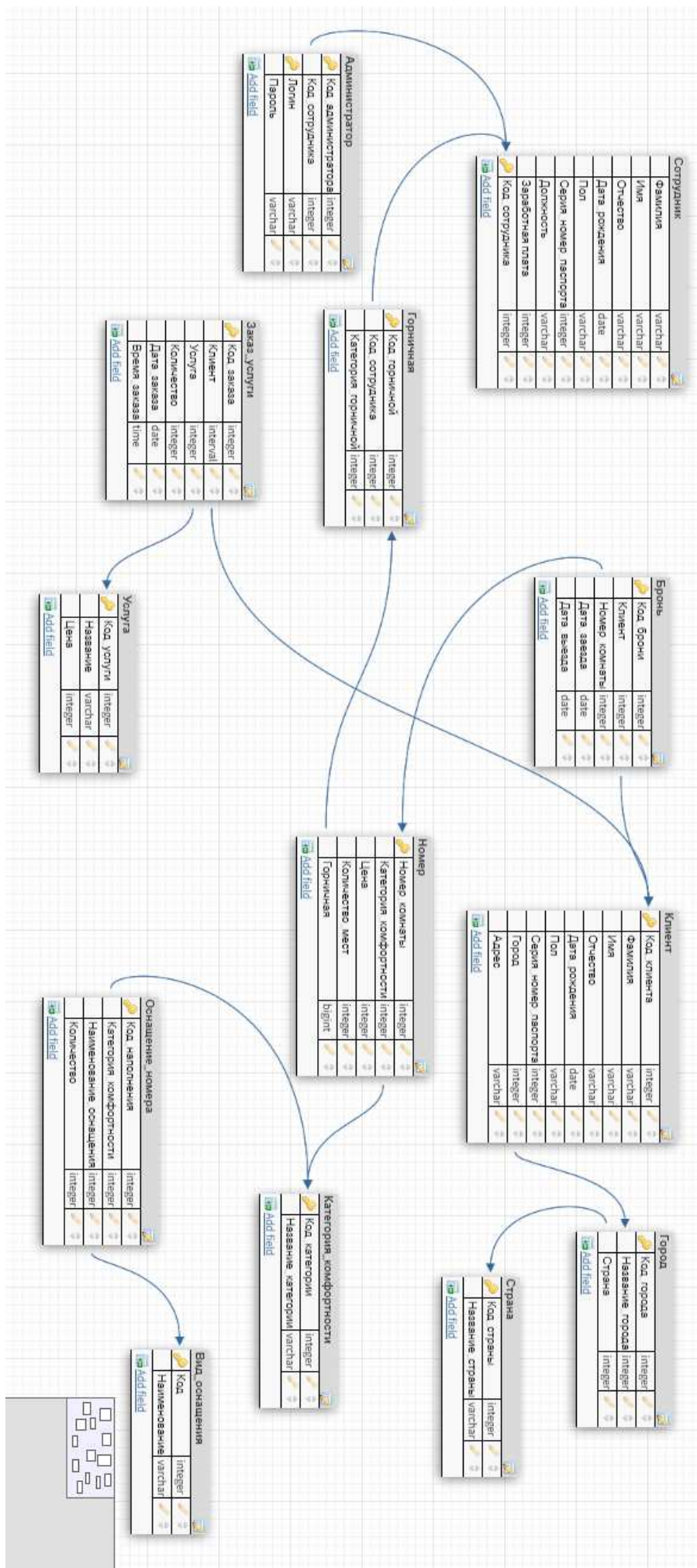


Рисунок 3 — ER-диаграмма.

В четвертом разделе рассматривается реализация информационной системы. В качестве инструмента разработки был выбран язык Python. Этот язык является мощным, но в то же время простым для изучения языком программирования, где предоставлены проработанные высокоуровневые структуры данных. Язык обладает такими возможностями, как: динамичность, поддержка ООП и кросс-платформенность, что позволяет легко и быстро внедрять проекты в разные операционные системы. Синтаксис Python минималистичен, библиотека включает множество полезных функций.

Графический интерфейс пользователя (англ. Graphical User Interface, GUI) — система взаимодействия пользователя с программным приложением, основанная на представлении всех доступных пользователю системных объектов и функций в виде графических компонентов экрана (окон, меню, панелей инструментов, кнопок, элементов диалога).

Одним из наиболее часто используемых модулей для создания GUI приложений в Python является PyQt5. Этот модуль также включает в себя Qt Designer — дизайнер графического интерфейса пользователя. Программа ruic генерирует Python код из файлов, созданных в Qt Designer.

На рисунке 4 представлен графический интерфейс процесса авторизации пользователя в системе. Администратор вводит логин и пароль, если данные введены верно, происходит вход в систему. Далее администратору предоставляется возможность работы с клиентами и номерами гостиницы.

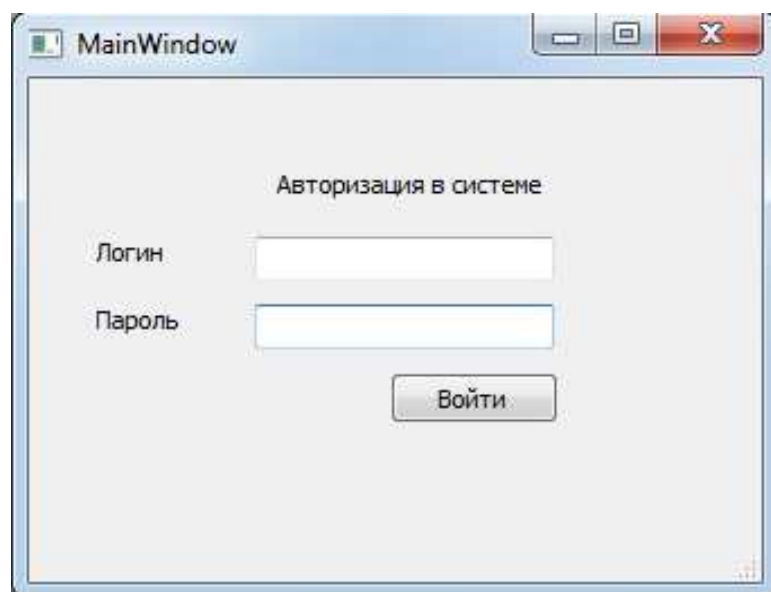


Рисунок 4 — Авторизация пользователя.

На рисунке 5 представлен графический интерфейс выбора категории комфорта. Администратор имеет возможность удалить одну из категорий или добавить новую. Выбрав одну из имеющихся категорий, администратор может посмотреть ее оснащение.

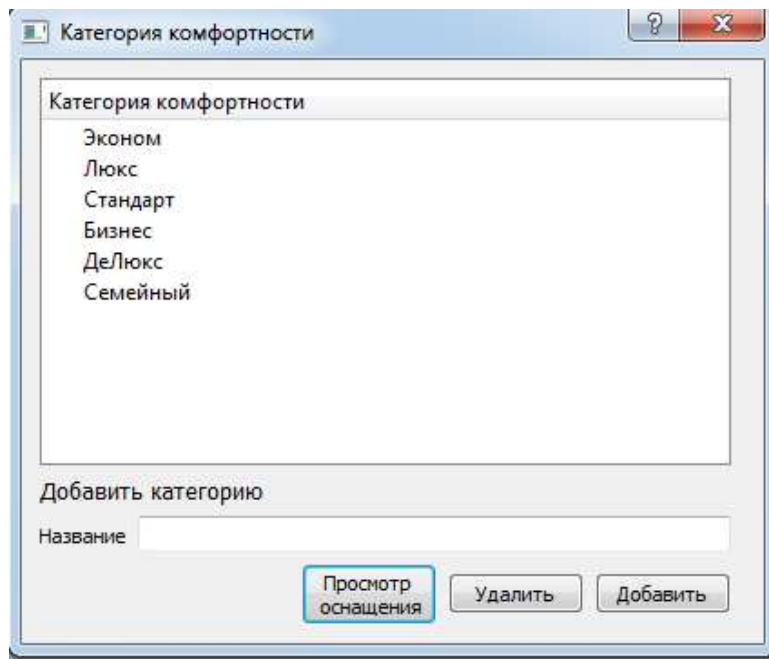


Рисунок 5 — Выбор категории комфорта.

На рисунке 6 показан вариант оснащения номера категории комфорта «люкс».

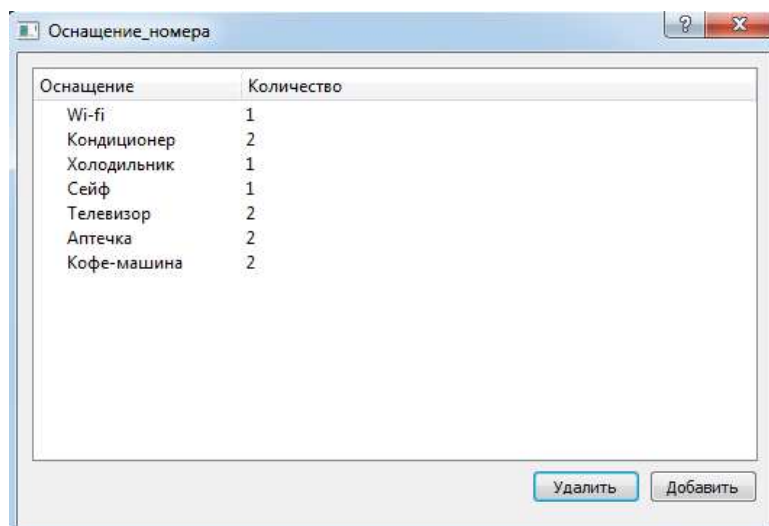


Рисунок 6 — Оснащение номера.

Заключение. В процессе выполнения дипломного проекта разработаны UML-диаграммы прецедентов и активности, отражающие основные функции клиента гостиницы, которые он может выполнить посредством информационной системы. Таким образом, удалось визуализировать систему с различных точек зрения. Одна из диаграмм описывает взаимодействие пользователя с системой, другая — изменение состояний системы в процессе ее работы, третья — взаимодействие между собой элементов системы.

Разработана база данных с использованием СУБД PostgreSQL, содержащая подробные сведения о гостинице, номерах, о стоимости номеров и об услугах, предоставляемых гостиницей. Обеспечивает доступ к информации о клиентах, проживающих в номерах гостиницы, их регистрации во время приезда, и о возможности бронирования.

Была реализована информационная система с помощью языка программирования Python и модуля PyQt5, который позволил создать графический интерфейс системы.