

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Специальная часть	4
1.1 Проектирование схем	4
1.2 Разработка структуры базы данных.....	8
1.3 Описание реализации проекта	9
1.4 Описание программы.....	16
1.5 Проведение испытаний.....	16
ЗАКЛЮЧЕНИЕ	22

ВВЕДЕНИЕ

Ателье – слово, ассоциирующиеся у современного человека в первую очередь с деятельностью организации, направленной на ремонт одежды. Подобные предприятия чаще всего занимаются почти любыми видами работ связанными с перекроем, пошивом или работами восстановительного характера.

Деятельность ателье базируется на умении специального обученного персонала определять вид работ, ее примерную стоимость и необходимое время. Мастера работают в определенных условиях, одним из которых, к примеру, является наличия специального оборудования.

Также как и любая другая организация, ателье обязано иметь свою собственную отчетность о ведении деятельности для персонала и клиентуры. Однако как бы далеко не зашел прогресс – на данный момент, большинство подобных организаций пользуются традиционным видом оформления своей информационной системы – пересчитывая огромное кол-во квитанций каждый месяц, что весьма не практично. Отсюда и возникает актуальность темы автоматизации информационной системы. Необходим, переход от бумажной коммуникации к безбумажной, иначе говоря – электронной, чтобы сократить время оформления необходимых документов, а также иметь возможность получать доступ к информации о клиентах, заказах почти мгновенно.

Уже более десятилетия информационные технологии являются одной из важнейших частей жизнедеятельности человека. Экономические информационные системы, связанные с предоставлением и обработкой информации для всех уровней управления экономическими объектами, приобретают особую важность в общественной жизни. На данный момент невозможно представить какую-либо организацию, не использующую компьютерных технологий.

1 Специальная часть

1.1 Проектирование схем

Use Case Diagram / Диаграмма вариантов использования Диаграмма вариантов использования в UML — диаграмма, отражающая отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне. В данном случае рассматриваются отношения между «Администратором», «Мастерами», «Клиентами» и надлежащих им прецедентов. На основе выявленных ролей в структуре, были составлены диаграммы использования для каждой, отображенные на рисунка 1, 2 и 3 соответственно.

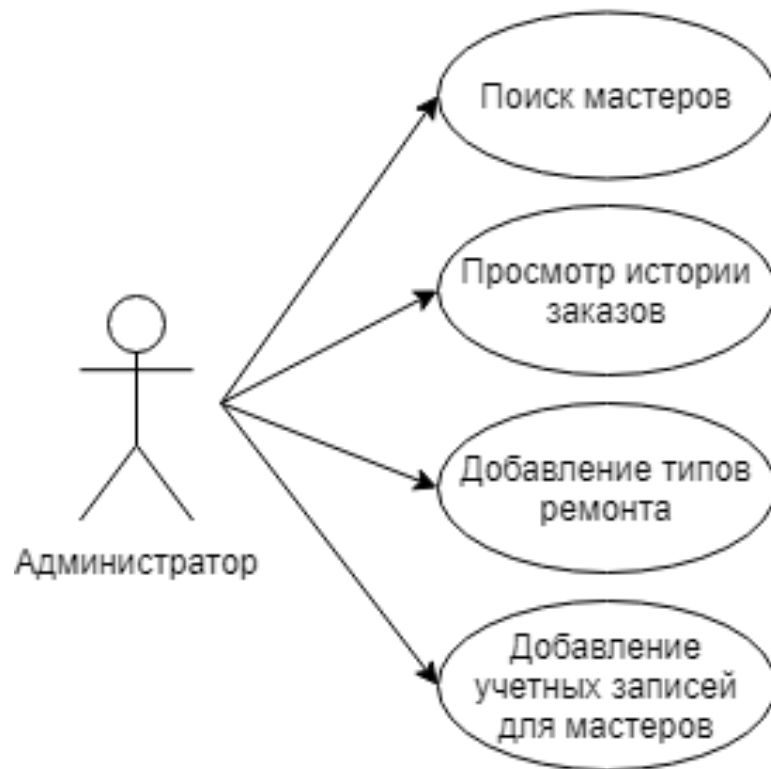


Рисунок 1 – Диаграмма вариантов использования - клиент

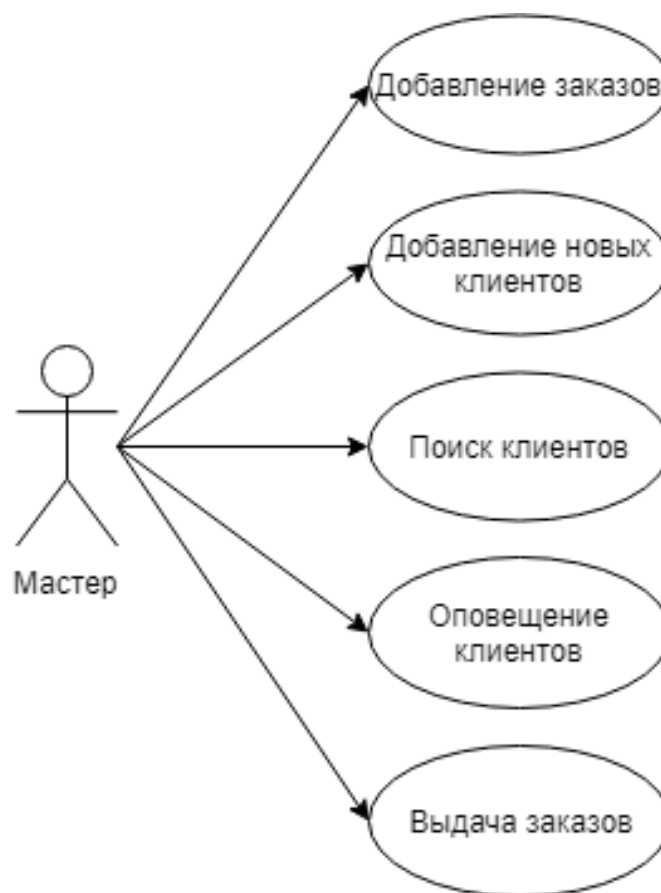


Рисунок 2 – Диаграмма вариантов использования - мастер

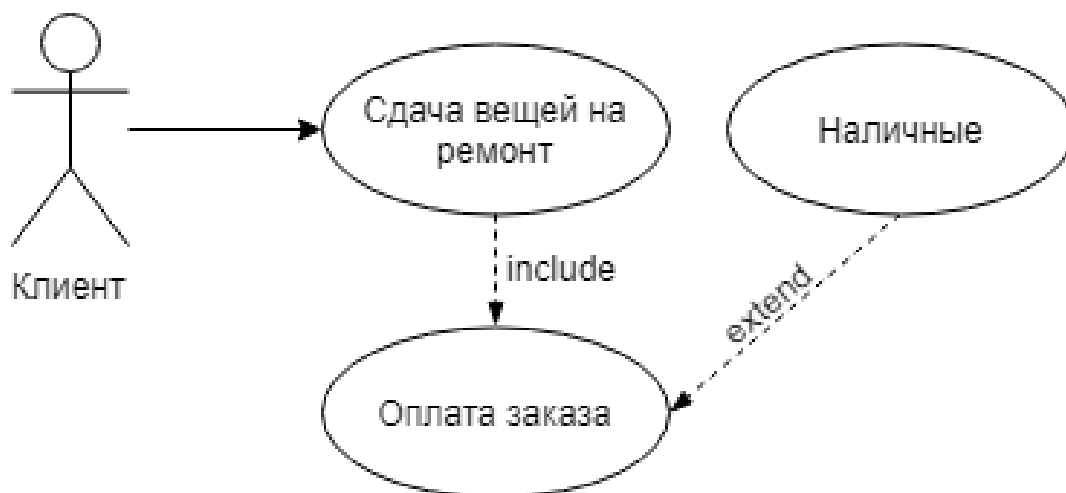


Рисунок 3 – Диаграмма вариантов использования – клиент

На диаграмме последовательности отображен порядок взаимодействия элементов системы: пользователь (мастер), клиент, программа. Сама диаграмма последовательности представлена на рисунке 4.

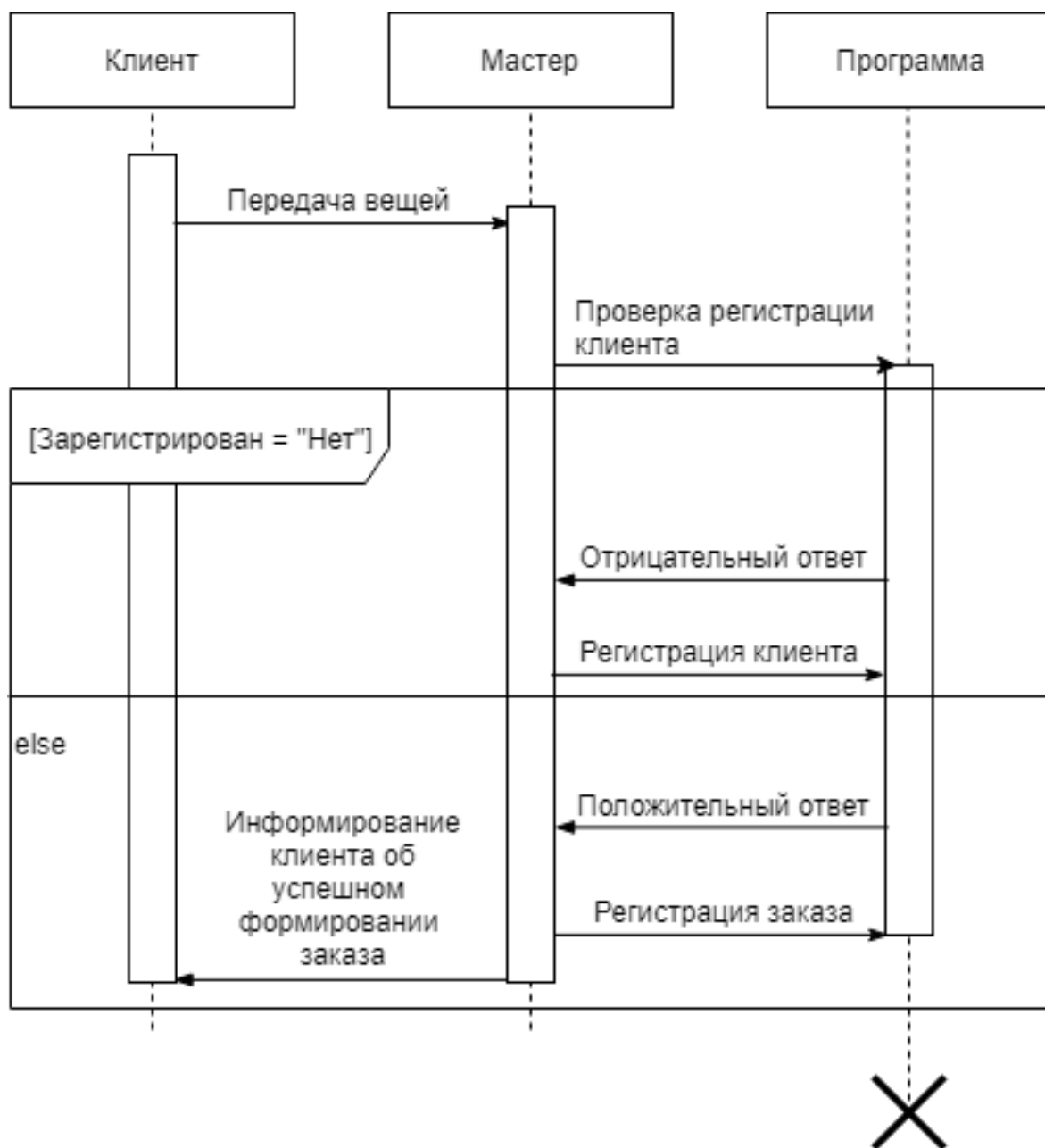


Рисунок 4 – Диаграмма последовательности создания заказа

Activity Diagram / Диаграмма активности является представлением алгоритмов неких действий (активностей), выполняющихся в системе. На основе требований к функциональным характеристикам, была спроектирована диаграмма активности, представленная на рисунке 5, представляющая «процесс приема-выдачи заказа».

Процесс приема-выдачи заказа



Рисунок 5 – Диаграмма активности

1.2 Разработка структуры базы данных

Для работы программы необходимо было создать базу данных, которая включает в себя следующие сущности:

- «Пользователь» необходима для хранения учетных записей пользователей, содержит: ФИО, логин и пароль;
- «Клиент» необходима для хранения данных о клиентах, содержит: ФИО, адрес электронной почты, номер мобильного телефона;
- «Категории» необходима для хранения перечня видов работ, предоставляемых организацией, содержит: название;
- «Заказ» необходима для хранения данных о заказах, содержит: стоимость работы, дату приема заказа, дату выдачи заказа, статус выполнения, кто выдал заказ, а также название категории, имя клиента и имя мастера.

Схема базы данных отображена на рисунке 6.

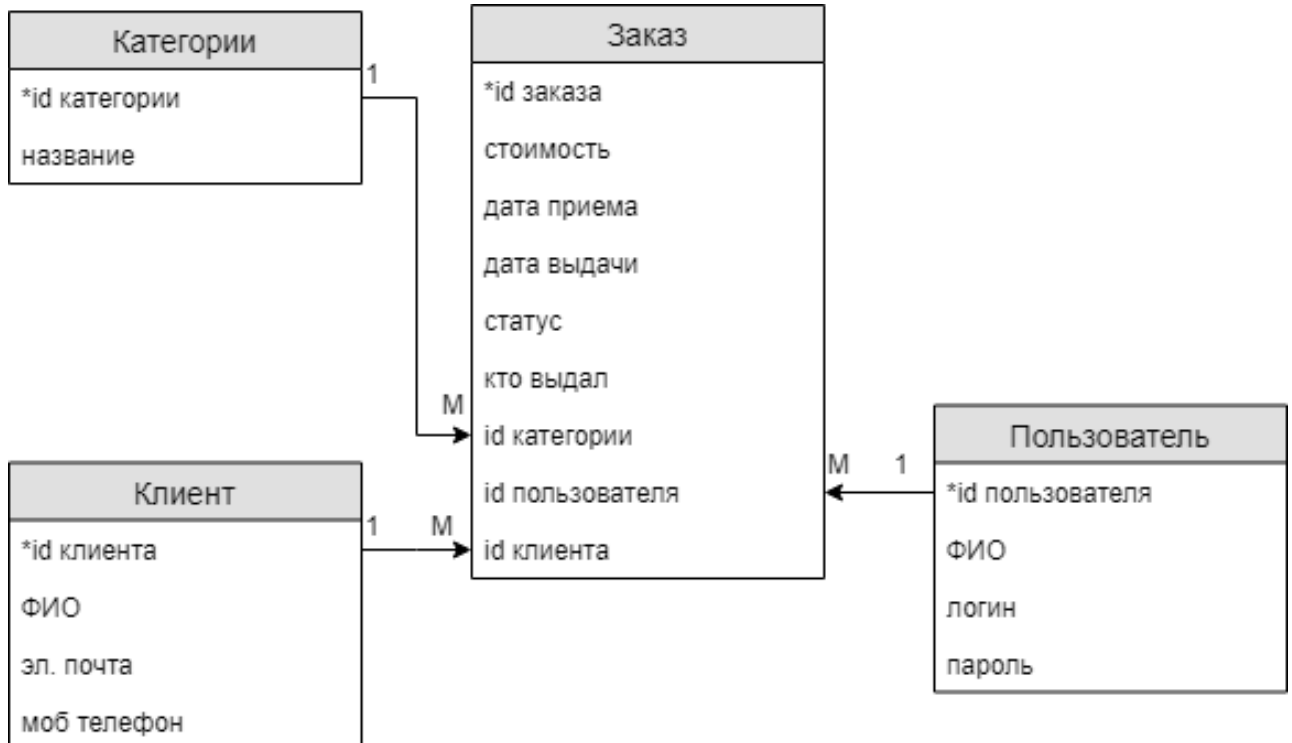


Рисунок 6 – Схема базы данных

Реализовать базу данных было решено в среде Microsoft SQL Server Management Studio 2017 (интегрированная среда для управления любой инфраструктурой SQL). Расположение базы данных в MS SSMS представлено на рисунке 7.

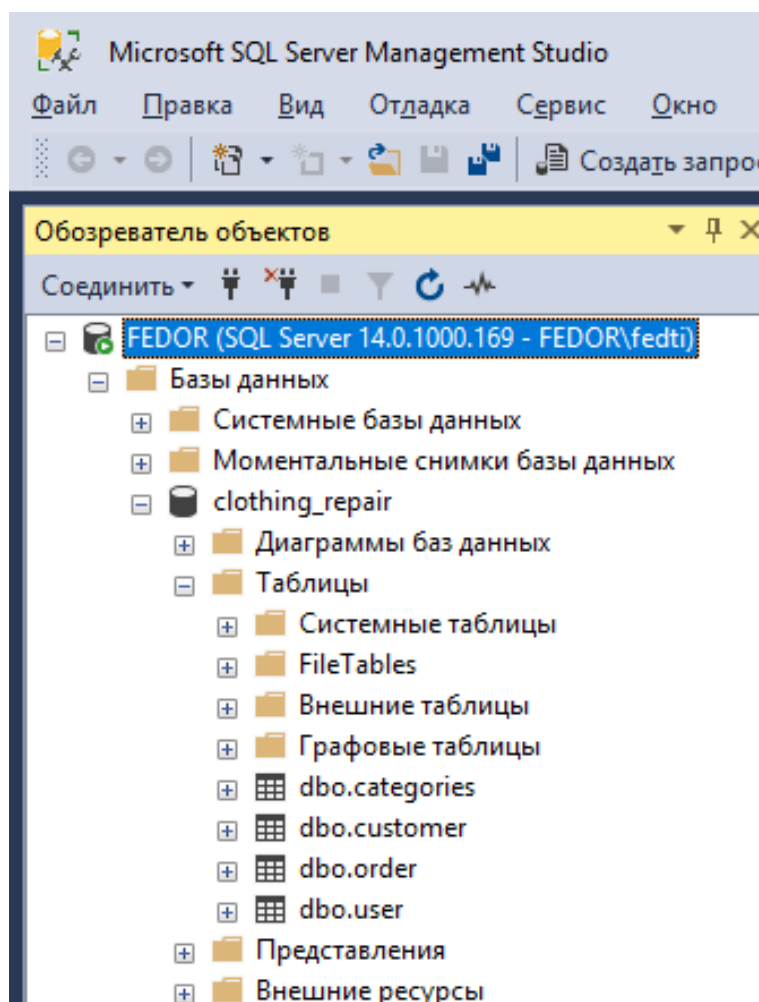


Рисунок 7 – База данных в MS SSMS 2017

1.3 Описание реализации проекта

Следующим этапом разработки стало проектирование и создание модели базы данных, спроектированной в среде MS Visual Studio 2017 (рисунок 8), а также пользовательского интерфейса конечной информационной системы.

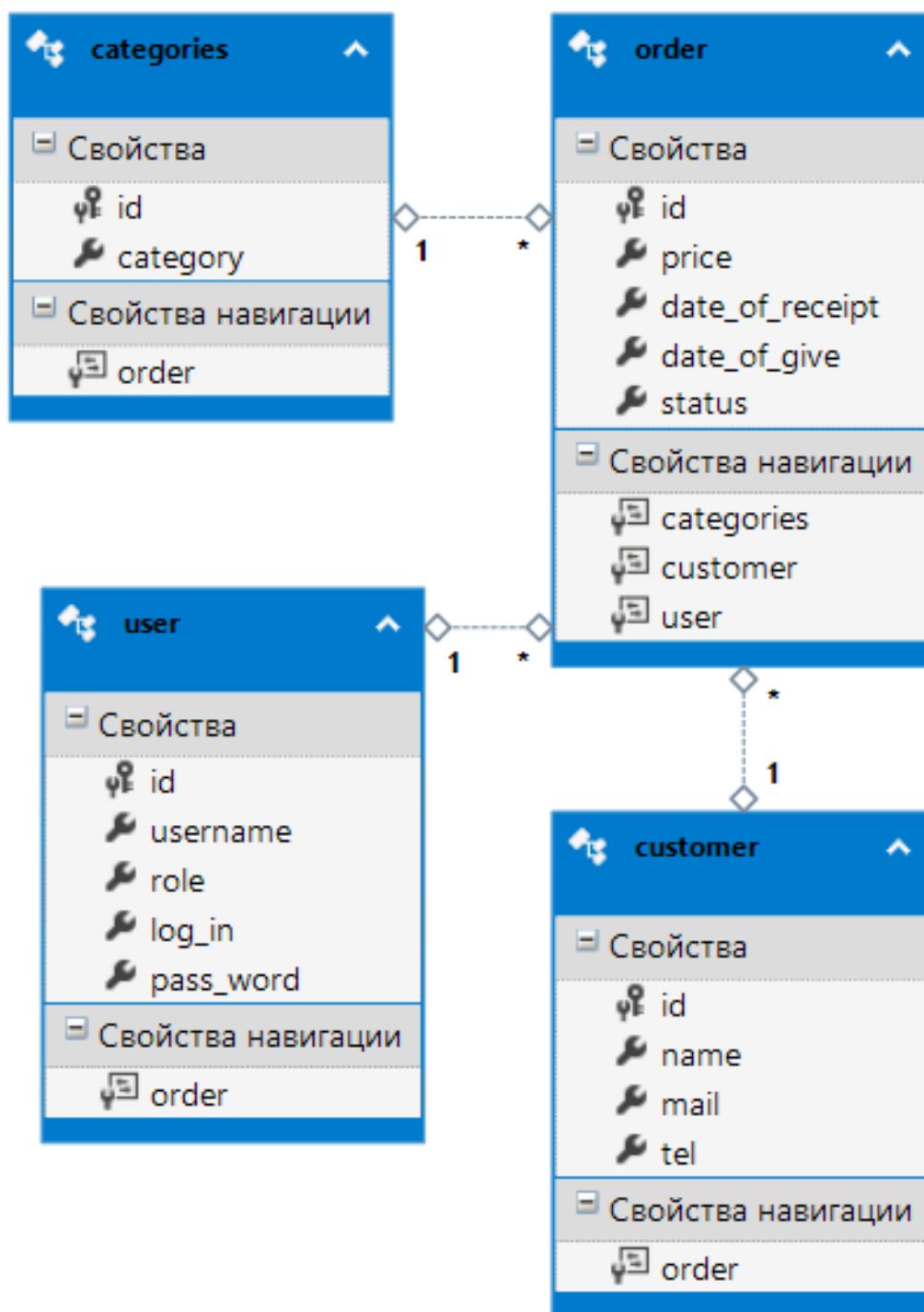


Рисунок 8 – Модель базы данных

При разработке интерфейса важно учитывать потребности пользователя. Цвета должны в меру яркими, преимущественно с цветами логотипа организации. Пример дизайна интерфейса представлен на рисунке 9.



Рисунок 9 – Пример дизайна формы авторизации

Переход между активностями программы реализован через соответствующие кнопки в окнах.

На рисунке 9 отображена кнопка «Войти», задействуется после ввода необходимых данных выше (логина и пароля) для перехода на соответствующую форму (административную или рабочую).

На рисунке 10 представлен внешний вид профиля руководства (административная форма). Он содержит несколько элементов, такие как: строка поиска, таблица зарегистрированных мастеров, форма регистрации мастеров, кнопки «Категории» и «Перечень всех заказов», а также – «Выйти» (осуществляет выход из профиля, перемещая пользователя назад на форму авторизации).

CR 1.0 - Профиль руководства [Admin]

Поиск *фюо*

Имя	Уровень	Логин	Пароль
sas	master	sas	sas
Admin	admin	admin	admin

ФИО работника *фюо*

Специализация *Администратор* ▾

Логин *логин*

Пароль *пароль*

Зарегистрировать

Категории

Перечень всех заказов

Выйти

Рисунок 10 – Форма «Профиль руководства»

По нажатию кнопки «Категории» пользователю откроется окно создания новых видов ремонта (или проще – добавления новой услуги), представленная на рисунке 11. После ввода названия нового вида ремонта, пользователь завершает добавление записи путем нажатия кнопки «Добавить».

CR 1.0 - Новая категория

Название замена молнии

Добавить

Наименование
пошив

Рисунок 11 – Окно добавления новых категорий (видов ремонта)

По нажатию кнопки «Перечень всех заказов» пользователю откроется форма, на которой будут отображены все созданные заказы (рисунок 12).

CR 1.0 - Список заказов

Мастер	Клиент	Вуд	Цена	Прием	Статус	Выдача	Кто выдал
sas	Тимушев Ф.А.	пошив	5000	5/19/2019 11:22	Выдан	5/19/2019 11:27	sas

Рисунок 12 – Окно списка всех заказов

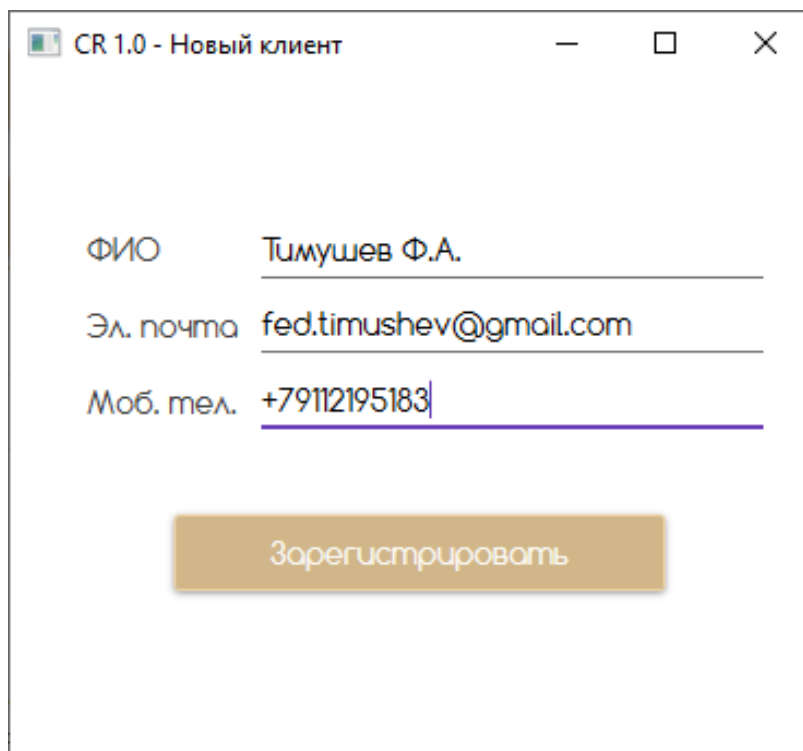
На рисунке 13 представлен внешний вид панели мастера (рабочая форма). Она содержит несколько элементов, такие как: строка поиска, таблица зарегистрированных клиентов, кнопка регистрации клиентов, форма создания нового заказа, кнопки «Создать заказ», «Оповестить заказчика», «Выдать заказ» а также – «Выйти» (осуществляет выход из профиля, перемещая пользователя назад на форму авторизации).

The screenshot shows a web application window titled "CR 1.0 - Панель мастера [sas]". The interface is divided into several sections:

- Registration Section:** A button labeled "Регистрация клиента" is at the top left. Below it is a search bar with a "Поиск" button and a "моб. тел." input field.
- Client Table:** A table with 8 columns: "Клиент", "Bug", "Цена", "Маст", "Прием", "Стат", "Выдача", and "Кто в". The first row contains the data: "Тимуш", "пошив", "5000", "sas", "5/19/20", "Выдан", "5/19/20", and "sas".
- Form Section:** Below the search bar is a form with three fields: "Имя" (filled with "Тимушев Ф."), "Эл. почта" (filled with "fed.timushe"), and "Телефон" (filled with "+79112195183"). Below this is a "Bug ремонта" dropdown menu (set to "пошив") and a "Цена" input field (filled with "цена"). There is also a "Скидка:" label.
- Action Buttons:** At the bottom, there are four buttons: "Создать заказ" (bottom left), "Оповестить заказчика" (bottom center), "Выдать заказ" (bottom right), and "Выйти" (bottom right corner).

Рисунок 13 – Форма «Панель мастера»

По нажатию кнопки «Регистрация клиента» пользователь перейдет к форме регистрации новых клиентов (рисунок 14), заполнив все необходимые поля, пользователь завершает регистрацию клиента нажатием кнопки «Зарегистрировать».



CR 1.0 - Новый клиент

ФИО Тимушев Ф.А.

Эл. почта fed.timushev@gmail.com

Моб. тел. +79112195183

Зарегистрировать

Рисунок 14 – Форма «Новый клиент»

По нажатию кнопки «Создать заказ» производится добавление нового заказа в правую таблицу на основе введенных пользователем данных в поля «Вид ремонта» и «Цена», а также выбранного клиента в таблице слева.

Действие кнопки «Оповестить заказчика» кроется в самом названии. После выбора соответствующего заказа в таблице справа и последующего нажатия на кнопку – на адрес электронной почты заказчика будет выслано письмо с просьбой забрать заказ.

Кнопка «Выдать заказ» используется только в момент непосредственной передачи готового заказа клиенту, после чего в таблице справа в выбранном заранее заказе, в поле «Статус» появится соответствующая запись «Выдан».

1.4 Описание программы

1.4.1 Завершение работы программы

Если необходимо завершить работу программы, требуется нажать на кнопку «Выйти» на любой профильной форме, затем на кнопку закрытия программы. Программа завершит свою работу. Все сессии закроются и при последующем входе, потребуется повторная авторизация.

1.5 Проведение испытаний

1.5.1 Объект испытаний

Наименование программного продукта «АИС управления заказами для ИП Тимушева А.Ю.».

Программа предназначена для использования персоналом ателье указанной выше организацией.

1.5.2 Цель испытаний

Целью испытаний является проверка соответствия программного продукта требованиям Технического задания.

1.5.3 Требования к функциональным характеристикам

На основе потребностей клиента были сформированы следующие требования к функциональным характеристикам:

- авторизация в программе;
- возможность оформления заказа;
- система клиентской базы (регистрация заказчиков);
- система скидок по кварталу;
- регистрация работников (мастера и один администратор);

- полная статусность заказа;
- клиентская рассылка;
- сводка по всем заказам для всех мастеров;

1.5.4 Требования к программной документации

Состав программной документации должен включать в себя:

- спецификация;
- техническое задание;
- текст программы;
- программу и методики испытаний;
- руководство оператора.

1.5.5 Средства и порядок испытаний

Требуются следующие технические средства:

- 32-разрядный (x86) или 64-разрядный (x64) двухъядерный процессор с тактовой частотой 2,0 гигагерц (ГГц) или выше;
- 2 гигабайт (ГБ) (для 32-разрядной системы) или 4 ГБ (для 64-разрядной системы) оперативной памяти (ОЗУ);
- 50 гигабайт (ГБ) свободного пространства на жестком диске.

Требуются следующие программные средства: ОС Windows 7 (32, 64-bit) и выше, MS SQL Server 2017

1.5.6 Методы испытаний

Тестирование программного продукта проводилось по методу черного ящика на основе функциональных требований, заявленных в Техническом задании, и включало в себя ручное тестирование (тест-кейсы).

1.5.7 Тест-кейсы

Текст-кейс 1 - Проверка способности системы предоставлять доступ к профилю руководства.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – admin, пароль – admin)
- нажать на кнопку «войти»

Ожидаемый результат:

- после успешной авторизации откроется профиль руководства.

Фактический результат:

- тест успешно пройден.

Текст-кейс 2 - Проверка способности системы предоставлять доступ к профилю мастера.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – sas, пароль – sas)
- нажать на кнопку «войти»

Ожидаемый результат:

- после успешной авторизации откроется профиль мастера.

Фактический результат:

- тест успешно пройден.

Текст-кейс 3 - Проверка способности добавлять учетные записи мастеров.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – admin, пароль – admin)
- нажать на кнопку «войти»
- в форме регистрации мастера ввести необходимые данные
- нажать кнопку «зарегистрировать»

Ожидаемый результат:

- запись мастера добавиться в БД, а также в таблицу мастеров слева

Фактический результат:

- тест успешно пройден.

Текст-кейс 4 - Проверка способности программы добавлять новые виды услуг (категории ремонта).

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – admin, пароль – admin)
- нажать на кнопку «войти»
- нажать кнопку «категории»
- в открывшейся форме ввести название категории в соответствующее поле
- нажать кнопку «добавить»

Ожидаемый результат:

- созданная категория добавится в БД и отобразится в форме регистрации категорий в таблице.

Фактический результат:

- тест успешно пройден.

Текст-кейс 5 - Проверка способности программы предоставлять перечень всех заказов.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – admin, пароль – admin)
- нажать на кнопку «войти»
- нажать кнопку «Перечень всех заказов»

Ожидаемый результат:

- вывод перечня всех заказов в отдельной форме в соответствующей таблице.

Фактический результат:

- тест успешно пройден.

Текст-кейс 6 - Проверка способности выполнять регистрацию клиента.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – sas, пароль – sas)
- нажать на кнопку «войти»
- нажать кнопку «регистрация клиента»
- в открывшейся форме ввести необходимые данные в соответствующих полях
- нажать кнопку «зарегистрировать»

Ожидаемый результат:

- форма добавления клиентов закроется, новый клиент появится в БД и в таблице слева в «профиле мастера»

Фактический результат:

- тест успешно пройден.

Текст-кейс 7 - Проверка способности программы добавлять новые заказы.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – sas, пароль – sas)
- нажать на кнопку «войти»
- выбрать нужного клиента в таблице клиентов слева
- ввести необходимые данные в соответствующих полях
- нажать кнопку «Создать заказ»

Ожидаемый результат:

- новая запись заказа добавится в БД, а также выведется в таблице заказов справа

Фактический результат:

- тест успешно пройден.

Текст-кейс 8 - Проверка способности программы выполнять выдачу заказа и менять статус заказа.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – sas, пароль – sas)
- нажать на кнопку «войти»
- выбрать необходимый заказ в таблице справа
- нажать кнопку «Выдать заказ»

Ожидаемый результат:

- в таблице заказов выбранная запись изменит поля статус на «Выдан», «Дата выдачи» на текущую дату и «Выдал» на имя мастера, зашедшего в данный момент под своими данными.

Фактический результат:

- тест успешно пройден.

Текст-кейс 9 - Проверка способности программы осуществлять выход/смену учетной записи.

Тестовый сценарий:

- запустить программу
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – sas, пароль – sas)
- нажать на кнопку «войти»
- нажать на кнопку «выйти»
- в стартовом окне ввести логин и пароль в соответствующие поля (логин – admin, пароль – admin)
- нажать на кнопку «войти»

Ожидаемый результат:

- после первого входа пользователь попадает в профиль мастера, после второго – в профиль администратора.

Фактический результат:

- тест успешно пройден.

ЗАКЛЮЧЕНИЕ

В результате прохождения производственной практики были получены следующие результаты:

- спроектирована диаграмма активности, отражающая работу клинико-диагностической лаборатории;
- спроектирована диаграмма последовательности, отражающая работу лаборанта в клинико-диагностической лаборатории;
- разработана структура базы данных;
- разработан дизайн интерфейса;
- создано настольное приложение на языке C#;
- разработаны тест кейсы, для тестирования созданного приложения;
- проведено тестирование настольного приложения.

Модуль лаборанта является частью более масштабного проекта, основы для других модулей которого были разработаны в ходе курсового проектирования. Программный продукт, с расширенным набором функций уже разработан в рамках дипломного проекта.