СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc136857753)

[1 Теоретические основы разработки приложения 5](#_Toc136857754)

[1.1 Обзор и анализ существующих приложений по автоматизации приема заявок на предприятии 5](#_Toc136857755)

[1.2 Достоинства и недостатки существующих приложений по автоматизации приема заявок 5](#_Toc136857756)

[1.3 Постановка задачи на разработку 5](#_Toc136857757)

[2 Проектирование приложения по автоматизации работы Вычислительного центра 13](#_Toc136857758)

[2.1 Проектирование базы данных 13](#_Toc136857759)

[2.2 Проектирование приложения 16](#_Toc136857760)

[2.3 Описание основных элементов подсистемы 21](#_Toc136857761)

[2.4 Руководство пользователя 21](#_Toc136857762)

[2.5 Руководство администратора 23](#_Toc136857763)

[ЗАКЛЮЧЕНИЕ 27](#_Toc136857764)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 28](#_Toc136857765)

# ВВЕДЕНИЕ

Функциональность предприятия напрямую зависит от работоспособности и состояния электронного оборудования. Своевременный ремонт, обслуживание, реагирование на заявки пользователей и обновление техники необходимо для поддержания эффективного функционирования колледжа и комфортной работы сотрудников.

Оптимизация приема обращений сотрудников необходима. Проблемой является необходимость перевода в электронный вид всех обращений, созданных сотрудниками, для возможности отслеживания и своевременного разрешения проблемных ситуаций на предприятии.

Целью выпускной квалификационной работы является разработка приложения по автоматизации работы Вычислительного центра Университетского экономико-технологического колледжа, как средства автоматизированной и информационной поддержки работы логистики Вычислительного центра в плане процессов приема заявок, хранения информации и создания отчетов о выполненной работе.

В соответствии с поставленной целью можно сформулировать следующие задачи разработки системы:

- определение состава автоматизируемых задач;

- постановка задач системы;

- разработка технологии функционирования системы в условиях автоматизированного управления;

- определение структуры базы данных системы, построение информационной модели системы;

- построение интерфейса взаимодействия пользователя с системой;

Объектом исследования является колледж, который занимается подготовкой специалистов среднего профессионального уровня.

Предметом выпускной квалификационной работы является приложение, разрабатываемое на платформе Visual Studio с использование SQL Server Management Studio для организации базы данных.

# 1 Теоретические основы разработки приложения

## 1.1 Обзор и анализ существующих приложений по автоматизации приема заявок на предприятии

Одними из самых популярных сервисов по автоматизации приема заявок и процессов техподдержки являются «1С Коннект» и «HelpDeskEddy».

## 1.2 Достоинства и недостатки существующих приложений по автоматизации приема заявок

## 1.3 Постановка задачи на разработку

Перед проектированием приложения необходимо знать, какие требования предъявляются к программному обеспечению. Поэтому необходимо разработать документ «Техническое задание».

Техническое задание

1 Общие сведения

1.1 Наименование системы

1.1.1 Полное наименование системы

Полное наименование – приложения для обеспечения контроля и автоматизации работы ВЦ УЭТК.

1.2 Наименование организаций

1.2.1 Заказчик

Заказчик: «УЭТК СГУ».

Адрес фактический: г. Сочи, ул. Чайковского, д. 45.

1.2.2 Разработчик

Разработчик: Силенко Лев Андреевич.

Адрес фактический: г. Сочи.

2 Назначение и цели создания системы

2.1 Назначение системы

Приложение предназначено для повышения оперативности и качества работы сотрудников колледжа.

Основным назначением является автоматизация приема заявок на ремонты и обслуживание сотрудниками Вычислительного центра УЭТК СГУ.

В рамках проекта автоматизируется информационно-аналитическая деятельность в следующих бизнес-процессах:

1. авторизация в системе с использованием ролей;
2. сохранение, изменение и удаление данных о заявках;
3. сортировка, фильтрация и поиск данных о заявках;
4. сортировка, фильтрация и поиск данных о сотрудниках;

2.2 Цели создания системы

Целью разработки и внедрения Системы является автоматизация процесса приема заявок, за счёт разработки программного обеспечения.

Система должна решать следующие задачи:

1. обеспечивать авторизацию в системе с использованием ролей;
2. хранить данные о заявках;
3. хранить данные о сотрудниках;
4. предоставлять возможность работать с заявками.

3 Характеристика объектов автоматизации

Объектом автоматизации является процесс работы ВЦ, который включает в себя:

1. отслеживание выполненных и взятых в работу каждым работником заявок;
2. работа со статусами заявок;
3. комментирование и добавление информации о заявках.

Процессы осуществляются следующими специалистами:

работники ВЦ.

4 Технические требования к программе или программному изделию

4.1 Требования к системе в целом

4.1.1 Требования к структуре и функционированию системы

Программа состоит из двух основных частей: клиентской и серверной, между которыми должно быть налажено взаимодействие.

4.1.1.1 Требования к серверной части

На серверной части должна быть реализована база данных для хранения необходимых данных. Также должно быть реализовано взаимодействие с базой данных для отслеживания истории заявок пользователей, а также их добавлению.

4.1.1.2 Требование к взаимодействию клиентской и серверной частей

Взаимодействие между клиентской и серверной частями должно осуществляться посредством SQL-запросов. При получении запроса от сотрудника УЭТК, сервер должен ответить сообщением в виде списка заявок, а также сервер должен фиксировать изменения статусов и дополнительной информации заявок, которые сотрудники ВЦ и УЭТК отправляют при изменении настроек.

4.1.1.3 Требования к клиентской части

Клиентская часть должна быть реализована в виде десктопного приложения, запускаемого на персональном компьютере, и представлена в виде взаимодействующих между собой окон, для перемещения между которыми необходимо разработать элементы пользовательского интерфейса. Также пользователь должен иметь возможность изменять следующие параметры в настройках приложения:

1. сортировка и фильтрация заявок по различным критериям,
2. просмотр информации профиля.

4.1.2 Требования к численности и квалификации персонала системы и режиму его работы

4.1.2.1 Требования к численности персонала

Численность и квалификация персонала системы должны определяться с учетом следующих требований:

1. конфигурация системы должна быть спроектирована и реализована с целью минимизации количественного состава обслуживающего персонала;
2. структура системы должна предоставлять возможность управления всем доступным функционалом системы как одному сотруднику ВЦ, так и предоставлять возможность разделения ответственности по администрированию между несколькими сотрудниками.

4.1.2.2 Требования к квалификации персонала

Для работы с приложением определены следующие роли:

1. Администратор;
2. работник ВЦ УЭТК;
3. сотрудник УЭТК.

Обязанностями сотрудника ВЦ УЭТК являются:

1. работа с заявками;
2. добавление сведений о завершенных и находящихся в работе заявках.

Администратор должен обладать хорошим уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию программных средств.

Пользователи системы должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows.

4.1.2.3 Требования к режимам работы персонала

Все специалисты должны работать с нормальным графиком работы не более 8 часов в сутки.

4.1.3 Показатели назначения

4.1.3.1 Параметры, характеризующие степень соответствия системы назначению

Приложение должно обеспечивать следующие количественные показатели:

1. количество аналитических отчетов – по всем системам;
2. система должна работать без снижения скорости обработки данных;
3. работоспособность при загрузке данных более 100 пользователей.

4.1.3.2 Требования к приспособляемости системы к изменениям

Обеспечение приспособляемости системы должно выполняться за счет:

1. своевременности администрирования;
2. модернизации процессов сбора, обработки и загрузки данных в соответствии с новыми требованиями;
3. модификации процедур доступа и представления данных конечным пользователям;
4. наличия настроечных и конфигурационных файлов у программного обеспечения подсистем.

4.1.4 Требования к надежности

Уровень надежности должен достигаться согласованным применением организационных мероприятий и программно-аппаратных средств.

Надежность должна обеспечиваться за счет:

1. применения технических средств, системного и базового программного обеспечения;
2. соблюдения правил эксплуатации и технического обслуживания;
3. обучение пользователей и обслуживающего персонала.

При работе системы возможны следующие аварийные ситуации, которые влияют на работу системы:

1. сбой в электроснабжении сервера;
2. сбой в работе интернета;
3. ошибки Приложения, не выявленные при отладке и испытании;
4. сбои программного обеспечения сервера.

К надежности электроснабжения предъявляются следующие требования:

должно быть обеспечено бесперебойное питание сетевого оборудования.

Надежность аппаратных и программных средств должна обеспечиваться за счет:

1. предварительного обучения пользователей и обслуживающего персонала;
2. своевременного выполнения процессов администрирования;
3. соблюдения правил эксплуатации и технического обслуживания программных средств.

4.1.5 Требования к эргономике и технической эстетике

Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Элементы навигации должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны быть понятны и удобны. Интерфейс должен соответствовать современным эргономическим требованиям и обеспечивать удобный доступ к основным функциям системы. Клавиатурный режим ввода должен использоваться главным образом при заполнении и редактировании текстовых и числовых полей экранных форм.

Приложение должно обеспечивать корректную обработку аварийных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях приложение должно выдавать пользователю соответствующие сообщения, после чего возвращаться в рабочее состояние.

4.1.6 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Приложение должно быть рассчитано на эксплуатацию в составе программного комплекса заказчика. Техническая и физическая защита аппаратных компонентов системы, бесперебойное энергоснабжение, резервирование ресурсов, реализуется техническими и организационными средствами.

На основании результатов тестирования технических средств должны проводиться анализ причин возникновения обнаруженных неполадок и приниматься меры по их решению.

4.1.7 Требования к защите информации от несанкционированного доступа

4.1.7.1 Требования к информационной безопасности

Приложение должно обеспечивать защиту от несанкционированного доступа на уровне, не ниже установленного требованиями рабочего Приложения.

Компоненты системы защиты должны обеспечивать:

1. идентификацию пользователя;
2. подтверждение личности пользователя.

Защищённая часть приложения должна использовать «слепые» пароли (при наборе пароля его символы не показываются на экране либо заменяются одним типом символов). Защищённая часть системы должна предотвратить работу с неточной информацией под сеансом пользователя. Защищённая часть системы должна использовать многоуровневую систему защиты.

4.1.7.2 Требования к антивирусной защите

Средства антивирусной защиты должны быть установлены на всех рабочих местах пользователей и преподавателей. Средства антивирусной защиты рабочих местах пользователей и администраторов должны обеспечивать:

1. централизованное управление сканированием, удалением вирусов и протоколированием вирусной активности на рабочих местах пользователей;
2. централизованную автоматическую инсталляцию клиентского программного обеспечения на рабочих местах пользователей и администраторов;
3. централизованное автоматическое обновление вирусных сигнатур на рабочих местах пользователей и администраторов;
4. ведение журналов вирусной активности;
5. администрирование всех антивирусных продуктов.

4.1.8 Требования по сохранности информации при авариях

Программное обеспечение приложения должно восстанавливать свое функционирование при корректном перезапуске аппаратных средств. Должна быть предусмотрена возможность организации автоматического и ручного резервного копирования данных системы средствами системного и базового программного обеспечения.

4.1.9 Требования к защите от влияния от внешних воздействий

Защита от влияния внешних воздействий должна обеспечиваться средствами программно-технического комплекса Заказчика:

1. все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне;
2. для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы.

5. Состав и содержание работ по созданию системы

Работы по созданию системы выполняются в три этапа:

Проектирование. Разработка эскизного проекта. Разработка технического проекта (продолжительность — 1 месяц).

Разработка рабочей документации. Адаптация программ (продолжительность — 2 месяца).

Ввод в действие (продолжительность — 1 месяц).

# 2 Проектирование приложения по автоматизации работы Вычислительного центра

## 2.1 Проектирование базы данных

Так как главная задача приложения – это хранение данных, то необходимо разработать базу данных.

Для воспроизведения необходимых подсистем приложения, сперва необходимо разработать модель базы данных.

Исходя из нужных функций, в базе данных будет храниться информация для работы с авторизацией пользователей и данные заявках. Ради увеличения стабильности, сами материалы будут храниться на сервере.

ER-диаграмма была разработана на основании технического задания в программе «StarUML» с помощью модели «ER Diagram».

Сначала была проработана концептуальная модель данных для будущей базы данных, в которой будет хранится вся необходимая информация.

На рисунке 1 представлена разработанная концептуальная модель базы данных.

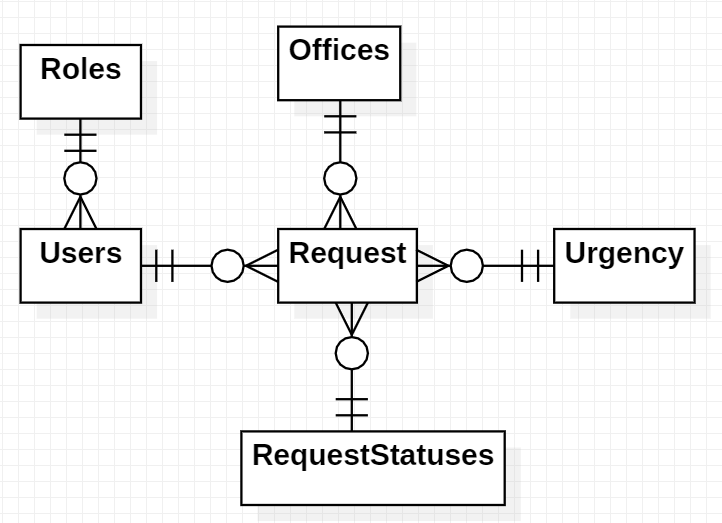


Рисунок 1 – Концептуальная модель базы данных

После разработки концептуальной схемы в среде SQL Management Studio была спроектирована и разработана физическая база данных, диаграмма которой представлена на рисунке 2.

База данных состоит из шести таблиц, которые полностью описывают предметную область разработки и позволят хранить всю необходимую информацию

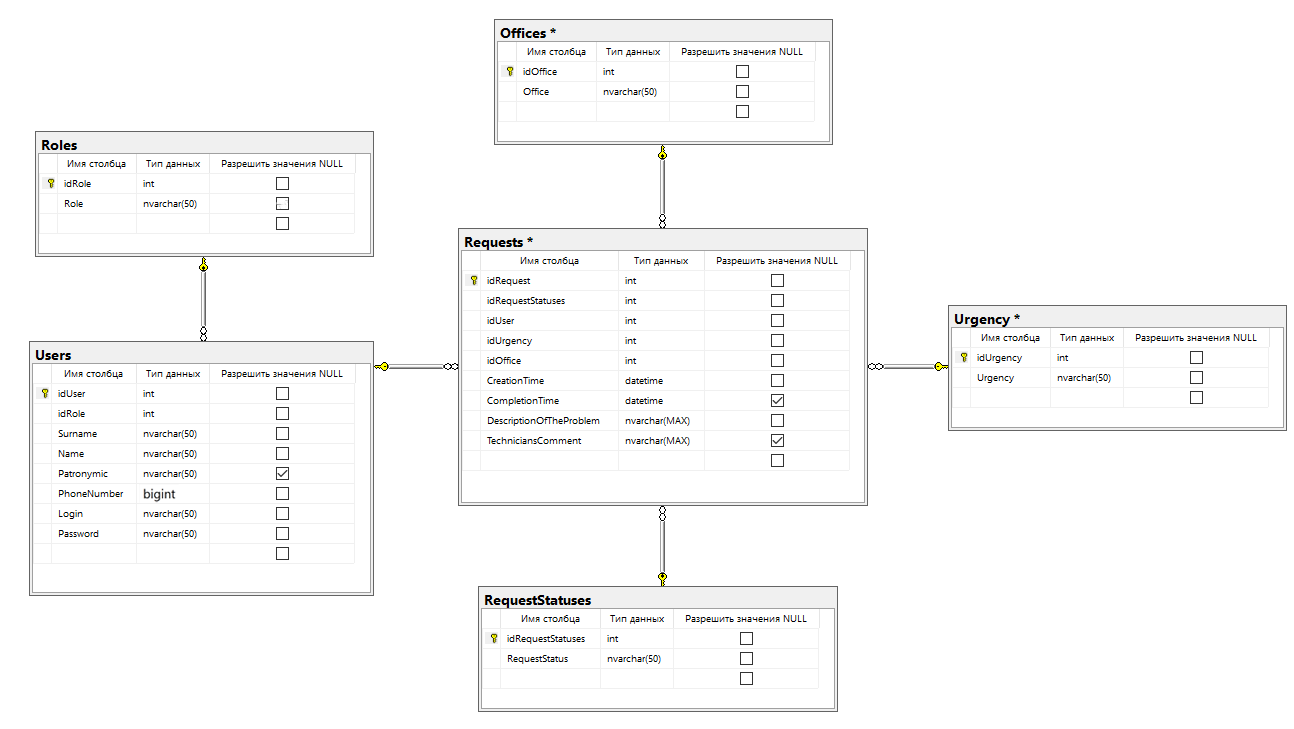


Рисунок 2 - Вид ER-диаграммы базы данных

Для будущей базы данных был создан словарь данных, приведенный в таблице 1.

Таблица 1 – Словарь данных

|  |  |  |  |
| --- | --- | --- | --- |
| Словарь данных | | | |
| Ключ | Поле | Null | Тип данных |
| 1 | 2 | 3 | 4 |
| Roles (Роли) | | | |
| PK | idRole (идРоли) | Нет | Int |
|  | Role (Роль) | Нет | Nvarchar(50) |
| Offices (Кабинеты) | | | |
| PK | idOffice (ИдКабинета) | Нет | Int |
|  | Office (Кабинет) | Нет | Nvarchar(50) |
| Urgency (Срочность) | | | |
| PK | idUrgency (ИдСрочности) | Нет | Id |
|  | Urgency (Срочность) | Нет | Nvarchar(50) |
| RequestStatuses (СтатусыОбращения) | | | |
| PK | idRequestStatus(идСтатусаОбращения) | Нет | Int |
|  | RequestStatus(СтатусОбращения) | Нет | Nvarchar(50) |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| Users (Пользователи) | | | |
| PK | idUser (ИдПользователя) | Нет | Int |
| FK | idRole (ИдРоли) | Нет | Int |
|  | Surname (Фамилия) | Нет | Nvarchar(50) |
|  | Name (Имя) | Нет | Nvarchar(50) |
|  | Patronymic (Отвество) | Да | Nvarchar(50) |
|  | PhoneNumber (НомерТелефона) | Нет | Bigint |
|  | Login (Логин) | Нет | Nvarchar(50) |
|  | Password (Пароль) | Нет | Nvarchar(50) |
| Requests (Заявки) | | | |
| PK | idRequest (идЗаявки) | Нет | Int |
| FK | idRequestStatus (идСтатусаЗаявки) | Нет | Int |
| FK | idUser (идПользователя) | Нет | Int |
| FK | idUrgency (идСрочности) | Нет | Int |
| FK | idOffice (идКабинета) | Нет | Int |
|  | CreationTime (ВремяСоздания) | Нет | datetime |
|  | CompletionTime (ВремяЗавершения) | Да | datetime |
|  | DescriptionOfTheProblem (ОписаниеПроблемы) | Нет | Nvarchar(50) |
|  | TechnicinasComment (КомментарийТехника) | Да | Nvarchar(50) |

## 2.2 Проектирование приложения

Диаграмма вариантов использования (use case)

Далее необходимо разработать диаграмму прецедентов, которая применяется для моделирования вида системы с точки зрения внешнего наблюдателя (актора). На диаграмме прецедентов графически показана совокупность прецедентов и акторов, а также отношения между ними.

В процессе проектирования были выделены следующие субъекты (акторы) системы:

- Администратор;

- Пользователь (сотрудники колледжа);

- Технический специалист.

На рисунке 3 представлена диаграмма вариантов использования

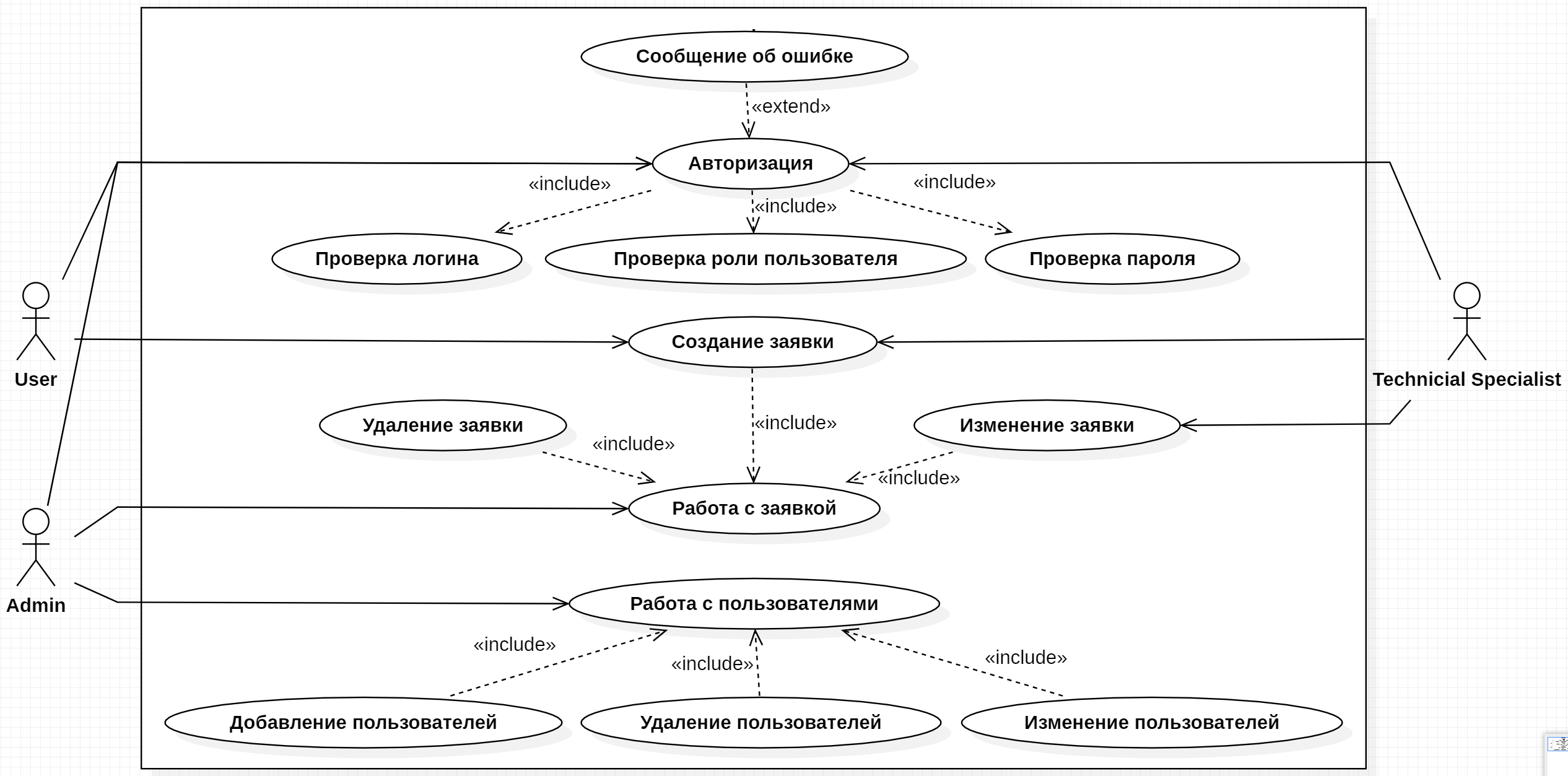


Рисунок 3 – Диаграмма прецедентов

Прецеденты (use case) для каждого субъекта представлены в виде структурных элементов (предоставленных графически в виде овалов) с названием прецедента для каждого субъекта.

В разрабатываемой системе каждый из субъектов имеет следующие прецеденты:

- субъект «Технический специалист» - прецедент «Авторизация», «Создание заявки», «Изменение заявки»;

- субъект «Администратор» - прецеденты: прецедент «Авторизация», «Работа с заявкой», «Работа с пользователями»;

- субъект «Пользователь» - прецеденты: прецедент «Авторизация», «Создание заявки».

Диаграмма последовательностей состояний

Диаграммы последовательностей состояний были разработаны на основании технического задания в программе «StarUML» с помощью модели «Sequence Diagram».

На рисунке 4 представлена диаграмма последовательностей состояний для подсистемы «Авторизация».

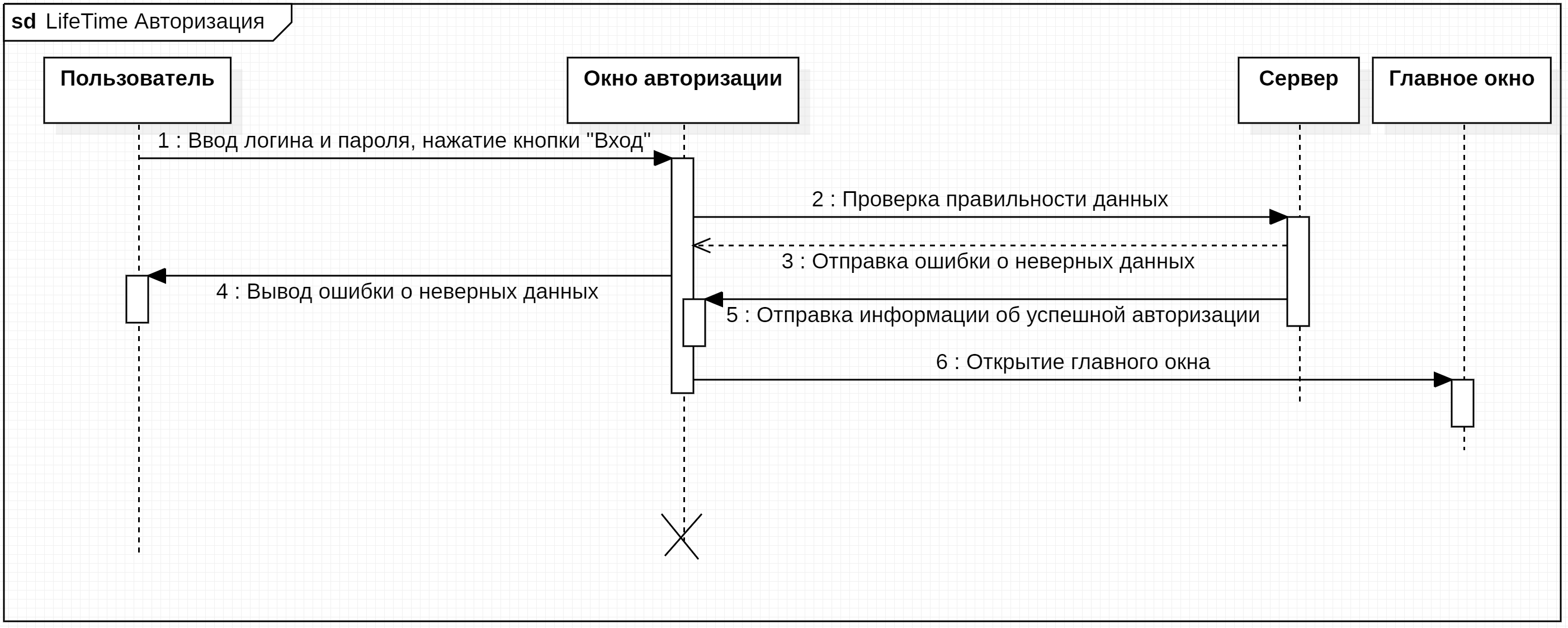


Рисунок 4 – Диаграмма последовательностей состояний для подсистемы «Авторизация»

При запуске приложения открывается страница «Авторизация», в которой можно войти в систему под своим логином и паролем. После ввода логина, пароля и нажатия на кнопку «Вход», данные отправляются на сервер, для проверки наличия в базе данных. Если таких логина и пароля в базе данных нет, то выводится ошибка. Если же такие данные существуют, то открывается главная страница.

На рисунке 5 представлена диаграмма последовательностей состояний для подсистемы «Создание заявок».

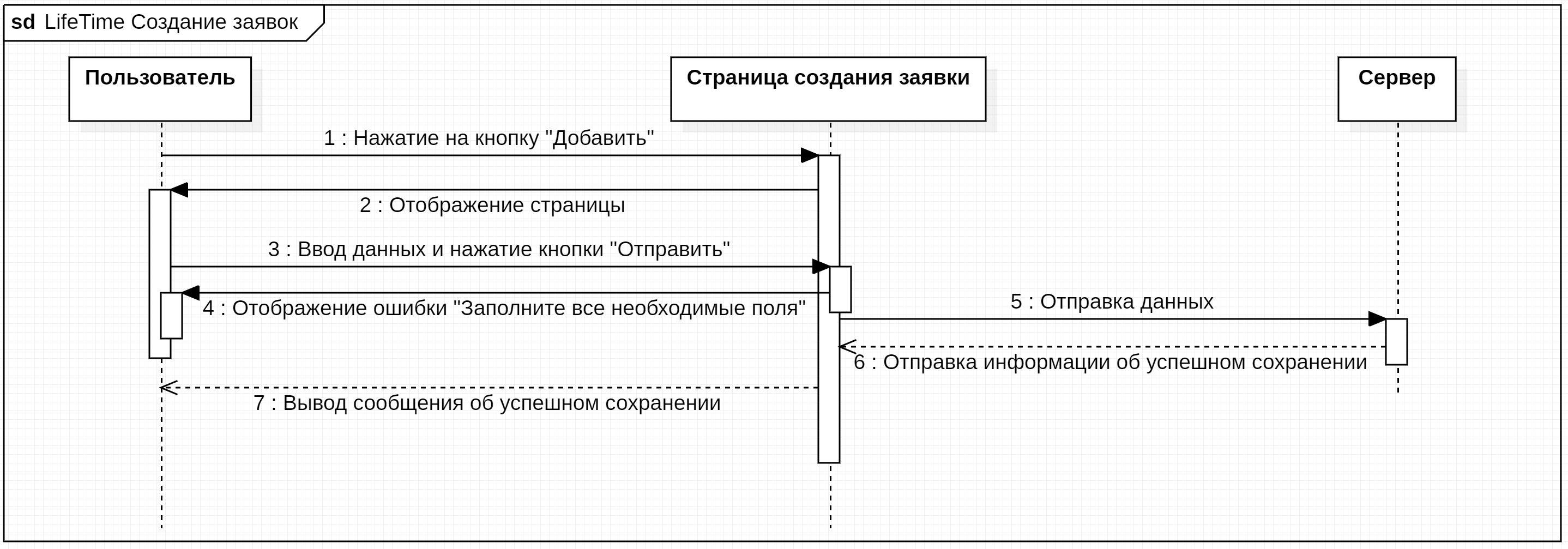


Рисунок 5 – Диаграмма последовательностей состояний для подсистемы «Создание заявок»

На странице создание заявки пользователю необходимо заполнить поля и написать комментарий о возникшей проблеме. После ввода данных и нажатия на кнопку «Отправить» данные отправляются на сервер, для проверки наличия в базе данных. Если заполнены не все необходимые поля, то выводится ошибка. Если же все поля были заполнены, то происходит сохранение данных на сервере и вывод сообщения об успешном сохранении.

Wireframe будущего приложения.

Проектирование приложения было начато с разработки форм приложения и организации логики взаимодействия между ними. Разработка логики взаимодействия между структурными элементами должна быть продумана таким образом, чтобы пользователю был интуитивно понятен каждый шаг работы с программным обеспечением.

Для отображения логики взаимодействия между формами приложения была разработана схема логического взаимодействия, представленная на рисунке 6.

Были разработаны страницы (frame), которые будут размещены (отрываться) при выполнении соответствующих операций с данными внутри окон (window):

- Страница «Авторизации»;

- Страница «Главного меню»;

- Страница «Пользователь»;

- Страница «Добавления заявки»;

- Страница «Меню пользователя».

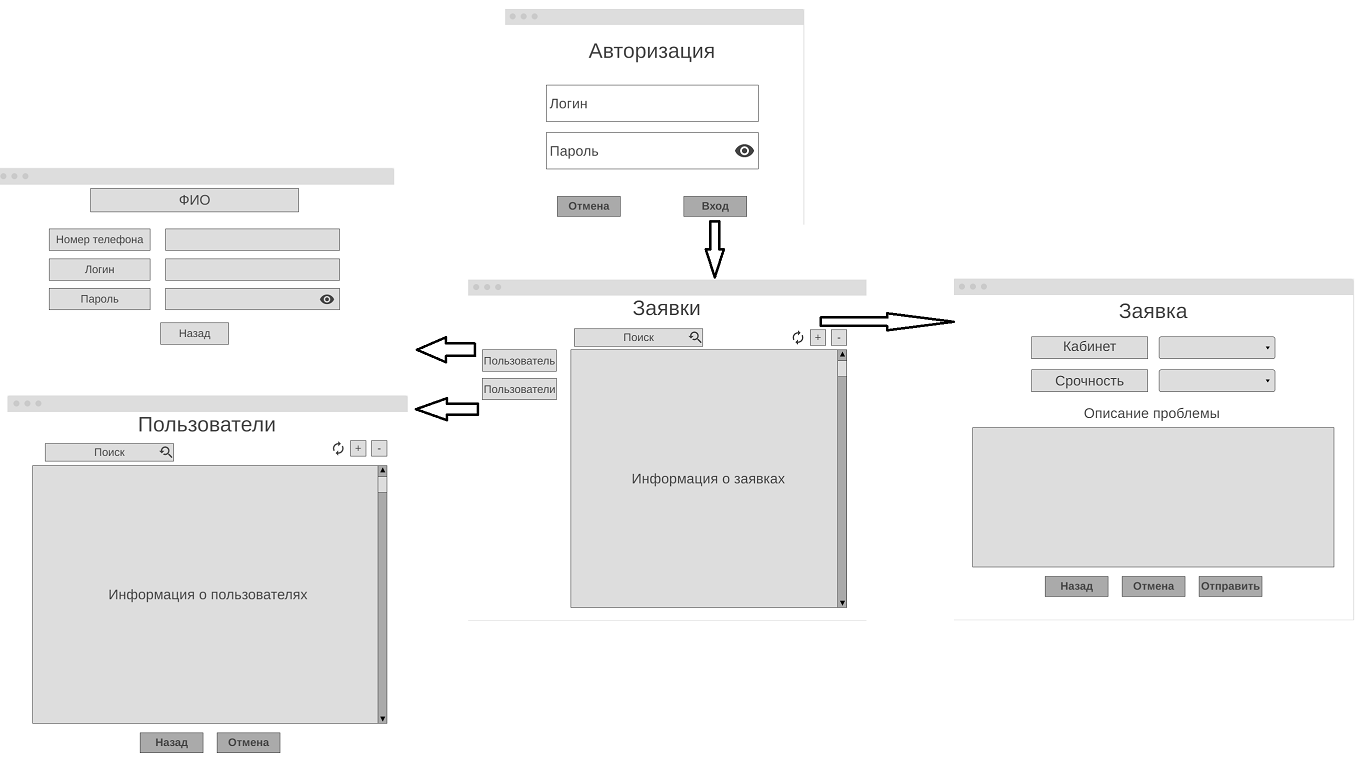


Рисунок 6 – Схема логического взаимодействия между структурными элементами приложения

После запуска приложения появляется главное окно, содержащее страницу «Авторизация». На этой странице пользователю необходимо ввести логин и пароль, после чего нажать на кнопку «Вход». Далее введённые логин и пароль отправляются на сервер для проверки верности данных. Если были введены неверные данные, то выводится ошибка и пользователь остаётся на странице «Авторизация». Если же введённые данные верно, то отобразится главная страница приложения.

На странице «Заявки» пользователь может использовать сортировку, при помощи нажатия на столбцы таблицы, фильтрацию, выбирая пункты в выпадающем списке. При добавлении и изменении данных можно обновить данные таблицы, нажав на соответствующую кнопку или подождать определенное время.

На странице «Создание заявки» пользователь может вернуться назад, нажав на кнопку «Назад», отменить ввод данных, нажав на кнопку «Отмена», или ввести данные и нажать на кнопку «Отправить», чтобы введённые данные внеслись в базу данных.

## 2.3 Описание основных элементов подсистемы

Подсистема состоит из четырёх основных элементов, которые обеспечивают ее функциональность:

Авторизация: Этот элемент подсистемы отвечает за идентификацию и аутентификацию пользователей, позволяя им получить доступ к функциональности. Авторизация обычно выполняется посредством ввода уникальных учетных данных, таких как логин и пароль, которые связаны с конкретным пользователем. После успешной авторизации пользователь получает доступ к хранимым данным заявках и пользователе, а также к возможностям изменения этой информации.

Хранение данных о заявках: Этот элемент подсистемы отвечает за хранение и организацию информации о заявках. Данные могут включать личные сведения, контактную информацию, академическую информацию, информацию о занятиях, а также другую относящуюся к студентам и преподавателям информацию.

Изменение данных о заявках: Этот элемент подсистемы позволяет пользователям вносить изменения в данные о заявках. Пользователи с соответствующими правами могут обновлять информацию о заявках.

Работа с пользователями. Этот элемент позволяет администратору обновлять, добавлять и изменять информацию о сотрудниках колледжа для работы пользователей в приложении.

## 2.4 Руководство пользователя

2 Описание операций

2.1 Выполняемые функции и задачи

выполняет функции и задачи, приведенные в таблице 7:

Таблица 7 - Функции и задачи

|  |  |
| --- | --- |
| Функции | Задачи |
| Обеспечивает авторизацию пользователей | Авторизация пользователей в приложении |
| Обеспечивает хранение и изменение данных о заявках | Добавление заявок |
| Изменение заявок |
| Обеспечивает возможность просмотра данных пользователя | Просмотр личной информации пользователя |
| Просмотр логина и пароля пользователя |

2.2 Описание операций технологического процесса обработки данных, необходимых для выполнения задач

В таблицах 8-11 приведено описание пользовательских операций для выполнения каждой из задач.

Таблица 8 - Задача: «Авторизация пользователей в приложении»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме. |
| Подготовительные действия | Не требуются |
| Основные действия в требуемой последовательности | 1. на иконке рабочего стола произвести двойной щелчок левой кнопкой мышки. 2. в открывшемся окне в поле «Логин» ввести имя пользователя, в поле «Пароль» ввести пароль пользователя. 3. нажать кнопку «Войти». |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. секунд |

Таблица 9 - Задача: «Добавление Заявки»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме. |
| Подготовительные действия | Авторизация. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Плюс» на главной странице справа; 2. заполнить необходимые данные; 3. нажать на кнопку «Добавить» в нижней части окна; 4. нажать на кнопку «Отправить». |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. минут |

Таблица 10 - Задача: «Изменение информации о заявках»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режим; * наличие информации о студенте в базе данных. |
| Подготовительные действия | Авторизация. |
| Основные действия в требуемой последовательности | 1. нажать на поле «комментарий» на главной странице; 2. ввести необходимый комментарий; 3. нажать на кнопку «Enter» на клавиатуре; 4. обновить таблицу; |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. минут |

Таблица 11 - Задача: «Просмотр данных пользователя»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме; * наличие информации о студенте в базе данных. |
| Подготовительные действия | Авторизация. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Пользователь» в меню слева; 2. Нажать на иконку глаза, открывающую видимость пароля; |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. секунд |

3. Аварийные ситуации

В случае возникновения ошибок при работе, не описанных в таблице 13, необходимо обращаться к администратору.

Таблица 12 - Аварийные ситуации

|  |  |
| --- | --- |
| Ошибка | Требуемые действия пользователя при возникновении ошибки |
| Нет подключения к серверу | Обратиться к администратору. |
| Неверный логин или пароль | Произвести 3 попытки входа. Если ошибка осталась, обратиться к администратору. |
| Введены некорректные данные в поле | Отредактировать данные в поле, следуя инструкции в ошибке. |

## 2.5 Руководство администратора

1.1 Краткое описание возможностей

Приложения для обеспечения контроля и автоматизации работы ВЦ УЭТК предназначено для оптимизации приема заявок на выполнение работ сотрудниками ВЦ УЭТК.

1.2 Уровень подготовки администратора

Пользователь должен иметь опыт работы с системами управления базами данных, ОС MS Windows 10 и знать соответствующую предметную область.

1.3 Способ использования системы

Приложение используется как десктопное приложение, требующее подключение к локальной сети и базе данных. Приложение используется для создания и отслеживания статусов заявок сотрудниками колледжа, а также для улучшения коммуникации между отделами и ускорения скорости работы всего колледжа.

2 Логика работы системы

Приложение создано для удобной работы с базой данной и представляет пользователю всю информацию, хранящуюся на сервере. Каждая операция пользования по изменению и добавлению данных отображается в базе данных и позволяет отслеживать работу сотрудников.

При авторизации пользователя проходит сопоставление данных на сервере, после чего приходит ответ приложения.

При создании заявки проверяется заполненное всех необходимых полей приложением, после чего данные отправляются на сервер.

При работе администратора с пользователями системы ему представляется таблица со всеми данными, имеющимися в базе данных пользователей. У администратора есть возможность изменять данные из приложения, вводя и изменяя информацию, записанную в ячейках таблицы. При изменении и добавлении информации в ячейки администратором происходит сопоставления типов данных полей и введенной информации, после чего данные записываются на сервере или выдается ошибка.

3 Функции администратора

3.1 Выполняемые функции и задачи

выполняет функции и задачи, приведенные в таблице 13:

Таблица 13 - Функции и задачи

|  |  |
| --- | --- |
| Функции | Задачи |
| Обеспечение безопасности данных | Контроль и работа с пользователями приложения |
| Резервное копирование данных |
| Разграничение доступа |
| Мониторинг системы | Удаление и изменение заявок |
| Сохранение структуры базы данных |

2.2 Описание операций технологического процесса обработки данных, необходимых для выполнения задач

В таблицах приведено описание пользовательских операций для выполнения каждой из задач.

Таблица 14 - Задача: «Удаление ошибочной заявки»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме. |
| Подготовительные действия | Авторизация |
| Основные действия в требуемой последовательности | 1. на главное странице выделить ошибочную заявку 2. нажать на кнопку «Удалить» в правом углу |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. секунд |

Таблица 15 - Задача: «Добавление Пользователя»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме. |
| Подготовительные действия | Авторизация, получения данных о новом пользователе. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Пользователи» на главной странице слева; 2. заполнить необходимые данные; 3. нажать на кнопку «Добавить». |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. минут |

Таблица 16 - Задача: «Изменение информации о заявке»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режим; * наличие информации о студенте в базе данных. |
| Подготовительные действия | Авторизация, получение информации о ошибке и корректной информации. |
| Основные действия в требуемой последовательности | 1. нажать на поле с ошибкой в заявке на главной странице; 2. ввести необходимый данные; 3. нажать на кнопку «Enter» на клавиатуре; 4. обновить таблицу; |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. минут |

Таблица 17 - Задача: «Изменение данных пользователя»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме; * наличие информации о студенте в базе данных. |
| Подготовительные действия | Авторизация, получение обновленной информации о пользователе. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Пользователи» в меню слева; 2. Выбрать необходимое поле; 3. Внести обновленную информацию; 4. Нажать кнопку «Enter» на клавиатуре; 5. Обновить таблицу |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | * 1. секунд |

3. Аварийные ситуации

Таблица 18 - Аварийные ситуации

|  |  |
| --- | --- |
| Ошибка | Требуемые действия пользователя при возникновении ошибки |
| Нет подключения к серверу | Проверка состояния сети колледжа и доступа базы данных |
| Несанкционированный доступ к данным | Отключение возможности подключения другим пользователям к базе данных и исправления информации пользователей |
| Нарушение структурной целостности базы данных | Восстановление данных из резервной копии |

# ЗАКЛЮЧЕНИЕ

Оптимизация приема обращений сотрудников необходима. Проблемой является необходимость перевода в электронный вид всех обращений, созданных сотрудниками, для возможности отслеживания и своевременного разрешения проблемных ситуаций на предприятии.

Целью выпускной квалификационной работы являлось создание приложение для оптимизации приема заявок на ремонт и обслуживание на предприятии. Для достижения данной цели били изучены вопросы, связанные с организацией рабочего процесса в общем, а также изучено программное обеспечение присутствующее на российском рынке.

Во второй главе выпускной квалификационной работы, было описано программное обеспечение, которое было выбрано в качестве платформы для разработки модуля ведения складского учета.

Далее проведена работа по проектированию базы данных для хранения и манипулирования данными в будущем приложении. После проектирования была разработана физическая модель базы данных с использованием MS SQL Server MS.

Также в данной главе было проведено проектирование самого модуля складского учета: построена Use Case диаграмма, прорисованы и описаны пользовательские сценарии, прорисован пользовательский интерфейс. Подробно описана работа по созданию всех элементов программного модуля.

В пункте «Руководство администратора» подробно описана работа будущего администратора системы, последовательность его действий при подключении базы данных к разработанному модулю.

В результате работы над выпускной квалификационной работой был разработано приложение по автоматизации работы Вычислительно центра Университетского экономико-технологического колледжа, которое готово к установке на предприятии и проведению нагрузочного тестирования с ним в реальных условиях работы данного предприятия.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Приказ Минобрнауки России от 09.12.2016 №1547 «Об утверждении федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование» (Зарегистрировано в Минюсте России 27.12.2016 №44936)
2. ГОСТ 7.32 ― 2017 Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. – М., Стандартинформ, - 2017, 32 с. (дата обращения: 17.05.2023)
3. ГОСТ Р 7.0.100 ― 2018 Библиографическая запись. Библиографическое описание: общие требования и правила составления. – URL: <http://it-mda.ru/standards/docs/GOST_R/GOST_R_7.0.100-2018.pdf?ysclid> =l0u192vnjk (дата обращения: 17.05.2023). – Текст: электронный.
4. ГОСТ 34.602-89 Техническое задание на создание автоматизированной системы. – URL: http://www.rugost.com/index.php?option=com\_content&view=article&id=96:gost-34602-89&catid=22&Itemid=53 (дата обращения: 17.05.2023). – Текст: электронный.
5. Зуб, А. Т. Управление проектами: учебник и практикум для среднего профессионального образования / А. Т. Зуб. — Москва: Издательство Юрайт, 2022. — 422 с. — (Профессиональное образование). — ISBN 978-5- 534-01505-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/491468 (дата обращения: 17.05.2023).
6. Инструкция №1 по охране труда работников при работе на персональном компьютере. – URL: <https://www.syktsu.ru/about/ot/instukcii_ot/%D0%B8%D0%BD%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%86%D0%B8%D1%8F%20%E2%84%961.pdf> (дата обращения: 17.05.2023).
7. Сведения об образовательной организации. – URL: <https://sutr.ru/sveden/> (дата обращения: 17.05.2023).
8. Стандарты и шаблоны для ТЗ на разработку ПО / Хабр. – URL: https://habr.com/ru/post/328822/ (дата обращения: 17.05.2023).
9. Техническое задание на создание автоматизированной системы ГОСТ 34.602-89. Пример технического задания. Пример техзадание. Проектирование хранилища данных. Проектная документация. – URL: https://www.prj-exp.ru/patterns/pattern\_tech\_task.php#tech\_task9 (дата обращения: 17.05.2023).
10. ER-диаграмма (ERD): определение и обзор | Lucidchart. – URL: https://www.lucidchart.com/pages/ru/erd-%D0%B4%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0 (дата обращения: 17.05.2023).
11. Использование диаграммы вариантов использования UML при проектировании программного обеспечения. – URL: https://habr.com/ru/post/566218/ (дата обращения: 17.05.2023).
12. Проектирование Use Case диаграммы. Определение функциональных возможностей системы — Национальная сборная Worldskills Россия. – URL: https://nationalteam.worldskills.ru/skills/proektirovanie-use-case-diagrammy-opredelenie-funktsionalnykh-vozmozhnostey-sistemy/ (дата обращения: 17.05.2023).
13. Схема состояний и переходов: ее понятие и особенности — TestMatick. – URL: https://testmatick.com/ru/chema-sostoyanij-i-perehodov-ee-ponyatie-i-osobennosti/ (дата обращения: 17.05.2023).
14. State & Transition Diagram — что это и как применять / Хабр. – URL: https://habr.com/ru/post/548192/ (дата обращения: 17.05.2023).
15. Краткий путеводитель по методологиям и нотациям описания и моделирования бизнес-процессов. Часть 4. – URL: https://infostart.ru/1c/articles/1451560/ (дата обращения: 17.05.2023).
16. Учебное пособие по диаграммам последовательностей: полное руководство с примерами - Крейтли Блог. – URL: https://creately.com/blog/ru/%D0%B4%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0/%D1%83%D1%87%D0%B5%D0%B1%D0%BD%D0%BE%D0%B5-%D0%BF%D0%BE%D1%81%D0%BE%D0%B1%D0%B8%D0%B5-%D0%BF%D0%BE-%D0%BF%D0%BE%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D0%B9/ (дата обращения: 17.05.2023).
17. Проектирование программного обеспечения / Хабр. – URL: https://habr.com/ru/post/74330/ (дата обращения: 17.05.2023).
18. Wireframes, Mockups, Prototype — что, куда, зачем | by Elena Saharova | Medium. – URL: https://medium.com/@elenasaharova/wireframe-mockups-prototype-%D1%87%D1%82%D0%BE-%D0%BA%D1%83%D0%B4%D0%B0-%D0%B7%D0%B0%D1%87%D0%B5%D0%BC-1769b53faa1a (дата обращения: 17.05.2023).
19. Wireframes в разработке: особенности и преимущества / Хабр. – URL: https://habr.com/ru/post/690598/ (дата обращения: 17.05.2023).
20. Free Online Diagram Editor. – URL: https://www.diagrameditor.com/ (дата обращения: 17.05.2023).
21. C# и WPF | Компоновка – URL: <https://metanit.com/sharp/wpf/4.php> (дата обращения: 17.05.2023).
22. C# и WPF | Grid – URL: <https://metanit.com/sharp/wpf/4.2.php> (дата обращения: 17.05.2023).
23. C# и WPF | StackPanel – URL: <https://metanit.com/sharp/wpf/4.4.php> (дата обращения: 17.05.2023).
24. C# и WPF | WrapPanel – URL: <https://metanit.com/sharp/wpf/4.6.php> (дата обращения: 17.05.2023).
25. C# и WPF | Canvas – URL: <https://metanit.com/sharp/wpf/4.7.php> (дата обращения: 17.05.2023).
26. C# и WPF | Свойства компоновки элементов – URL: <https://metanit.com/sharp/wpf/4.8.php> (дата обращения: 17.05.2023).
27. Связи между таблицами базы данных / Хабр. – URL: https://habr.com/ru/post/488054/ (дата обращения: 17.05.2023).
28. Документ "Руководство пользователя" - RuGost. – URL: http://www.rugost.com/index.php?option=com\_content&view=article&id=181&catid=27&Itemid=64 (дата обращения: 17.05.2023).
29. Пример оформления ГОСТ РД 50-34.698-90 Руководство пользователя. Oracle Discoverer, описание действий пользователя, рабочая документация. – URL: <https://www.prj-exp.ru/patterns/pattern_user_guide.php> (дата обращения: 17.05.2023).