# Ознакомление со структурой и характером деятельности предприятия

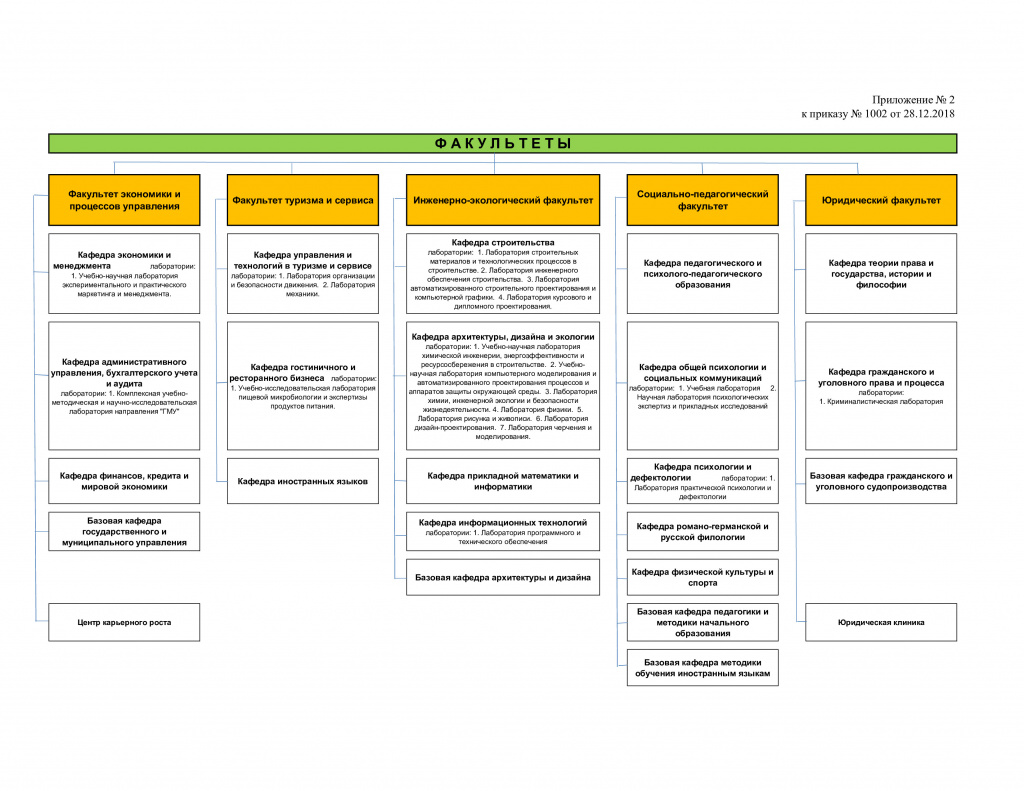
Для изучения общей организационной структуры предприятия была взята информация с сайта Университетского экономико-технологического колледжа Сочинского государственного университета.

Университетский экономико-технологический колледж (далее – УЭТК) является структурным подразделением Сочинского государственного университета. Педагогический штат колледжа насчитывает 110 профессионалов, в том числе кандидатов наук, преподавателей высшей и первой квалификационных категорий.

В настоящее время в колледже ведется подготовка по 15 специальностям среднего профессионального образования. Для качественного обеспечения процесса обучения УЭТК обладает развитой материально-технической базой.

При помощи сайта была изучена структура университета, в состав которого входит колледж УЭТК.

На рисунке 1 представлена структура университета.



1. Структура и органы управления факультетов

Так как разработка программного обеспечения ведётся для колледжа, необходимо было изучить структуру, непосредственно, самого колледжа УЭТК (рисунок 2).



1. Структура и органы управления колледжа

Согласно данным, основным руководителем колледжа является директор. Ему подчиняются четыре заместителя:

* заместитель директора по методической и научной работе;
* заместитель директора по учебно-производственной работе;
* заместитель директора по учебно-воспитательной работе;
* заместитель директора по реализации программ среднего общего образования.

Директор и заместители директора будут основными пользователями будущего приложения.

# Анализ и разработка системы

## Анализ деятельности организации УЭТК СГУ

Учёт информации о студентах и преподавателях СГУ ведётся в отделе кадров с помощью программного обеспечения «1С:Предприятие». Однако в колледже такой учёт осуществляется в Microsoft Word, а для учёта информации о сотрудниках используется Microsoft Excel.

Информацию о студентах колледжа собирают и обрабатывают, в очном, сотрудники отделения, а также сотрудники воспитательного отдела. Информацию о сотрудниках ведет директор и секретарь директора колледжа, передавая сведения в отдел кадров СГУ.

## Анализ программных средств

При разработке модуля для учета информации о студентах и сотрудниках УЭТК СГУ будут использованы следующие технологии и языки программирования:

* SQLite – СУБД;
* WPF – система для построения клиентских приложений;
* XAML – язык разметки в WPF;
* C# язык программирования отвечающий за логику приложения и управление элементами пользовательского интерфейса в WPF;

А также следующие программы и сайты:

* Для проектирования базы данных: StarUML, Excel;
* Для проектирования приложения: StarUML, diagrams.net, figma.com;
* Для разработки базы данных: SQLiteStudio;
* Для разработки приложения: Visual Studio 2022.

## Описание этапов разработки подсистемы

Разработка подсистемы ДоСиП проходит по следующим этапам:

1. Разработка технического задания;
2. Проектирование базы данных:
   1. ER-диаграмма;
   2. Словарь данных;
3. Проектирование приложения:
   1. Диаграмма вариантов использования (use case);
   2. Диаграмма переходов состояний;
   3. Диаграмма последовательностей состояний;
   4. Wireframe будущего приложения;
4. Написание тестовых сценариев для приложения;
5. Разработка базы данных для приложения;
6. Разработка приложения.

**3.3.1 Разработка технического задания**

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

Техническое задание

### Общие сведения

#### Наименование системы

##### Полное наименование системы

Полное наименование – Система автоматизации хранения данных о студентах и преподавателях.

##### Краткое наименование системы

Краткое наименование – ДоСиП, ДСП, или Система.

#### Основания для проведения работ

Работа выполняется в рамках производственной практики по ПМ.02 «Осуществление интеграции программных продуктов» в «УЭТК СГУ»

#### Наименование организаций

##### Заказчик

Заказчик: «УЭТК СГУ».

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

##### Разработчик

Разработчик: Богидаев Андрей Васильевич.

Адрес фактический: г. Сочи.

### Назначение и цели создания системы

#### Назначение системы

ДоСиП предназначена для повышения оперативности и качества работы сотрудников колледжа.

Основным назначением ДоСиП является автоматизация хранения данных о студентах и преподавателях.

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 студентах: фамилия, имя, отчество, номер телефон, адрес электронной почты, группа обучения, серия и номер паспорта, номер СНИЛС, номер страхового полиса, пол, дата рождения, адрес места рождения, кем выдан паспорт, дата выдачи, код подразделения, дата регистрации, адрес регистрации, адрес проживания;
3. сортировка, фильтрация и поиск данных о студентах;
4. сохранение, изменение и удаление данных о преподавателях: фамилия, имя, отчество, номер телефон, адрес электронной почты, должности, категории, степени, серия и номер паспорта, номер СНИЛС, номер страхового полиса, пол, дата рождения, адрес места рождения, кем выдан паспорт, дата выдачи, код подразделения, дата регистрации, адрес регистрации, адрес проживания;
5. сортировка, фильтрация и поиск данных о преподавателях;
6. печать выбранных данных в формате PDF документа.

#### Цели создания системы

Целью разработки и внедрения Системы является автоматизация процесса сохранения данных о студентах и преподавателях, за счёт разработки программного обеспечения.

Система должна решать следующие задачи:

1. обеспечивать авторизацию в системе с использованием ролей;
2. хранить данные о студентах;
3. хранить данные о преподавателях;
4. предоставлять возможность печати выбранных данных в формате PDF.

### Требования к системе

#### Требования к системе в целом

##### Требования к структуре и функционированию системы

Система ДоСиП должна быть централизованной, то есть все данные должны располагаться в центральном хранилище.

В Системе предлагается выделить следующие функциональные подсистемы:

* подсистема сбора, обработки и загрузки данных, которая предназначена для реализации процессов сбора данных из систем источников, приведения указанных данных к виду, необходимому для наполнения подсистемы хранения данных;
* подсистема хранения данных, которая предназначена для хранения данных в структурах, нацеленных на принятие решений;

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

###### Требования к квалификации персонала системы

К квалификации персонала, эксплуатирующего Систему ДоСиП, предъявляются следующие требования:

* конечный пользователь – знание соответствующей предметной области;

###### Требования к режимам работы персонала

Персонал, работающий с Системой и выполняющий функции её сопровождения и обслуживания, должен работать в следующих режимах:

* конечный пользователь - в соответствии с основным рабочим графиком подразделений Заказчика;

##### Требования к эргономике и технической эстетике

Подсистема формирования и визуализации отчетности данных должна обеспечивать удобный для конечного пользователя интерфейс, отвечающий следующим требованиям.

В части внешнего оформления:

* интерфейсы подсистем должен быть типизированы;
* должно быть обеспечено наличие локализованного (русскоязычного) интерфейса пользователя;
* должен использоваться шрифт: Verdana;
* размер шрифта должен быть: 16.

В части диалога с пользователем:

* для наиболее частых операций должны быть предусмотрены «горячие» клавиши;
* при возникновении ошибок в работе подсистемы на экран монитора должно выводиться сообщение с наименованием ошибки и с рекомендациями по её устранению на русском языке.

К другим подсистемам предъявляются следующие требования к эргономике и технической эстетике.

В части внешнего оформления:

* интерфейсы по подсистемам должен быть типизированы.

В части диалога с пользователем:

* для наиболее частых операций должны быть предусмотрены «горячие» клавиши;
* при возникновении ошибок в работе подсистемы на экран монитора должно выводиться сообщение с наименованием ошибки и с рекомендациями по её устранению на русском языке.

##### Требования к защите информации от несанкционированного доступа

###### Требования к информационной безопасности

Обеспечение информационное безопасности Системы должно удовлетворять следующим требованиям:

* защита Системы должна обеспечиваться комплексом программно-технических средств и поддерживающих их организационных мер;
* защита Системы должна обеспечиваться на всех технологических этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ;
* программно-технические средства защиты не должны существенно ухудшать основные функциональные характеристики Системы (надежность, быстродействие, возможность изменения конфигурации);
* разграничение прав доступа пользователей и администраторов Системы должно строиться по принципу «что не разрешено, то запрещено».

###### Требования к антивирусной защите

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

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

#### Требования к видам обеспечения

##### Требования к информационному обеспечению

###### Требования по применению систем управления базами данных

Для реализации подсистемы хранения данных должна использоваться промышленная СУБД Microsoft SQL Server.

###### Требования к защите данных от разрушений при авариях и сбоях в электропитании

Информация в базе данных системы должна сохраняться при возникновении аварийных ситуаций, связанных со сбоями электропитания.

Система должна иметь бесперебойное электропитание, обеспечивающее её нормальное функционирование в течение 15 минут в случае отсутствия внешнего энергоснабжения, и 5 минут дополнительно для корректного завершения всех процессов.

Резервное копирование данных должно осуществляться на регулярной основе, в объёмах, достаточных для восстановления информации в подсистеме хранения данных.

##### Требования к лингвистическому обеспечению

При реализации системы должны применяться язык программирования – C#, язык разметки – XAML и язык запросов – SQL.

Для реализации алгоритмов манипулирования данными в ДоСиП необходимо использовать стандартный язык запроса к данным SQL.

Для организации диалога системы с пользователем должен применяться графический оконный пользовательский интерфейс.

##### Требования к организационному обеспечению

Основными пользователями системы ДоСиП являются сотрудники Заказчика (например, секретари).

Обеспечивает эксплуатацию Системы подразделение информационных технологий Заказчика.

Состав сотрудников каждого из подразделений определяется штатным расписанием Заказчика, которое, в случае необходимости, может изменяться.

К организации функционирования Системы ДоСиП и порядку взаимодействия персонала, обеспечивающего эксплуатацию, и пользователей предъявляются следующие требования:

* в случае возникновения со стороны функционального подразделения необходимости изменения функциональности системы ДоСиП, пользователи должны обратиться к администратору;
* подразделение, обеспечивающее эксплуатацию системы, должно заранее (не менее чем за 3 дня) информировать всех пользователей (с указанием точного времени и продолжительности) о переходе её в профилактический режим.

К защите от ошибочных действий персонала предъявляются следующие требования:

* должна быть предусмотрена система подтверждения легитимности пользователя при просмотре данных;
* для всех пользователей должна быть запрещена возможность удаления пред настроенных объектов и отчетности;
* для снижения ошибочных действий пользователей должно быть разработано полное и доступное руководство пользователя.

##### Требования к патентной чистоте

По всем техническим и программным средствам, применяемым в системе, должны соблюдаться условия лицензионных соглашений и обеспечиваться патентная чистота.

Патентная чистота – это юридическое свойство объекта, заключающиеся в том, что он может быть свободно использован в данной стране без опасности нарушения, действующих на ее территории патентов исключительного права, принадлежащего третьим лицам (права промышленной собственности).

### 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).

### Проектирование базы данных

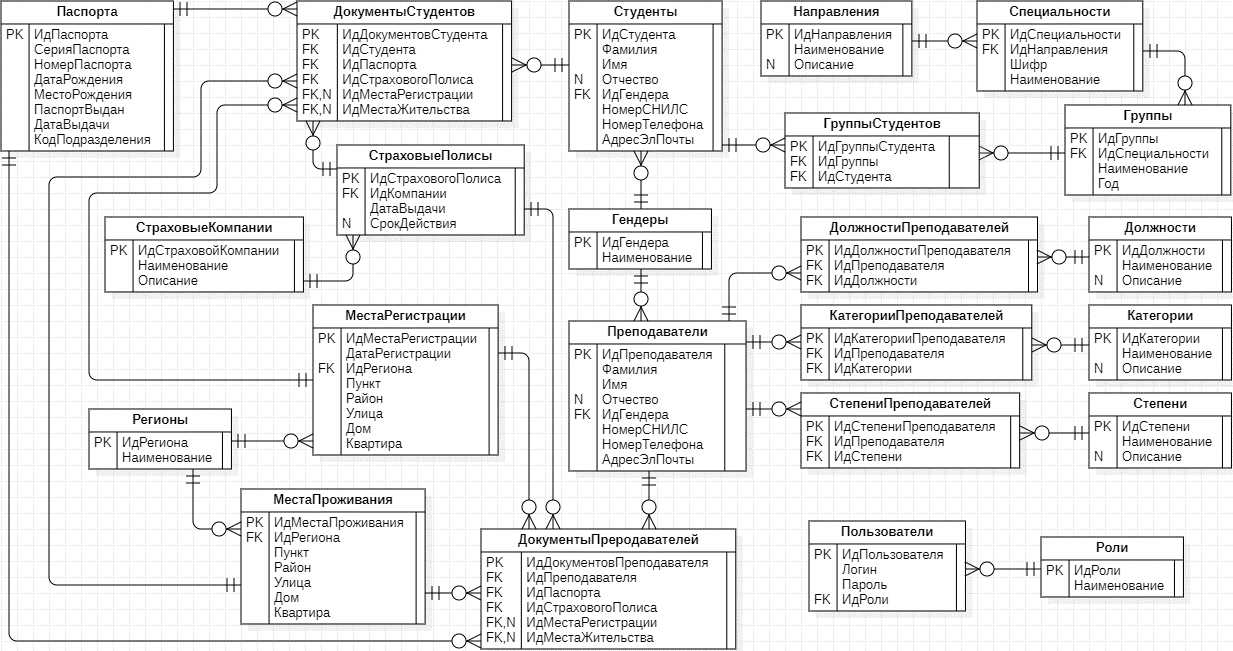
После того, как техническое задание было написано можно приступать к проектированию системы. Первое что было спроектировано – это база данных.

ER-модель — модель данных, позволяющая описывать концептуальные схемы предметной области.

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

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

На рисунке 5 представлена ER-диаграмма для ДоСиП.



1. ER-диаграмма

База данных будет состоять из двадцати трёх таблиц.

С помощьютаблиц «Пользователи» и «Роли» будет реализована возможность авторизации в приложении и разграничение прав на определённые функции.

Таблицы «Студенты» и «Преподаватели» содержат информацию о студентах и преподавателях, которая не меняется или если при изменении нет необходимости сохранять предыдущую версию.

Для преподавателей была разработана таблица «Преподаватели» содержащая те же поля, что и у студентов.

Так как студенты обучаются в группах, необходимо хранить информацию о группах, в которых он состоит, для этого была разработана таблица «ГруппыСтудентов».

Таблицы «Группы», «Специальности» и «Направления» будут хранить информацию о группах колледжа.

Также в базе данных должна хранится информация о должности, категории и степени преподавателя, для этого были разработаны таблицы «Должности», «ДолжностиПреподавателей», «Категории», «КатегорииПреподавателей», «Степени» и «СтепениПреподавателей».

Таблицы «ДокументыСтудентов» и «ДокументыПреподавателей» необходимы для объединения данных из следующих таблиц: «Паспорта», «СтраховыеПолисы», «МестаРегистрации» и «МестаПроживания».

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

1. Словарь данных

### Проектирование приложения

Диаграмма вариантов использования (use case)

После проектирования базы данных можно начать проектирование приложения.

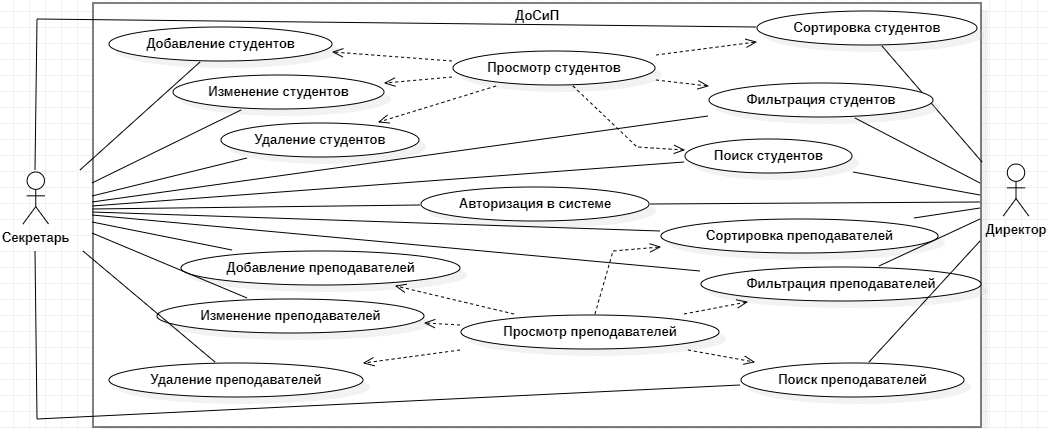
Диаграмма прецедентов или диаграмма вариантов использования в UML — диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Основное назначение диаграммы — описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему.

Данная диаграмма показывает функционал приложения для каждого пользователя.

Диаграмма вариантов использования для разрабатываемого приложения была спроектирована на основании технического задания в программе «StarUML» с помощью модели «Use Case Diagram».

На рисунке 3 представлена диаграмма вариантов использования для ДоСиП.



1. Диаграмма вариантов использования

Данная диаграмма состоит из двух акторов и соответствующих им прецедентов.

В системе выделены два актора:

* секретарь,
* директор.

А также следующих прецедентов:

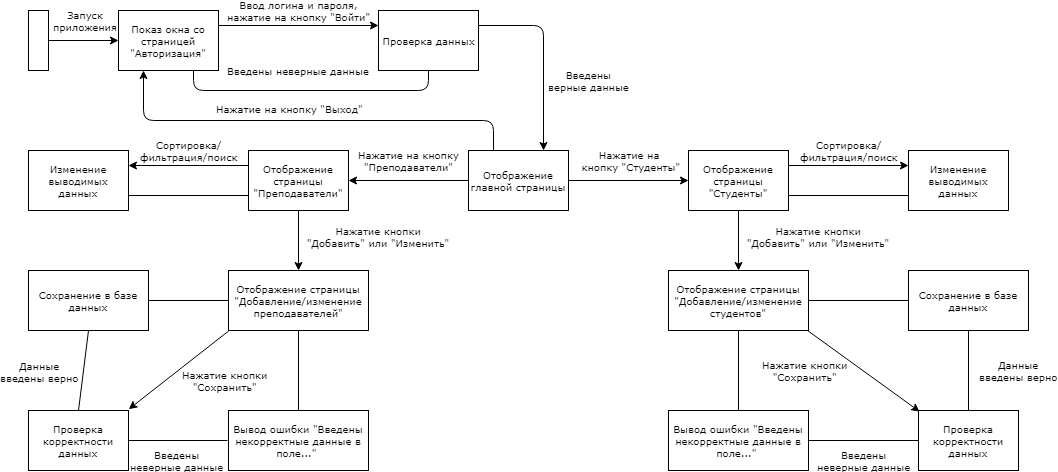
* «Авторизация в системе» - доступ имеют секретарь и директор;
* «Сортировка студентов» - доступ имеют секретарь и директор;
* «Фильтрация студентов» - доступ имеют секретарь и директор;
* «Поиск студентов» - доступ имеют секретарь и директор;
* «Добавление студентов» - доступ имеет только секретарь;
* «Изменение студентов» - доступ имеет только секретарь;
* «Удаление студентов» - доступ имеет только секретарь;
* «Просмотр студентов» - доступ имеют секретарь и директор;
* «Сортировка студентов» - доступ имеют секретарь и директор;
* «Фильтрация студентов» - доступ имеют секретарь и директор;
* «Добавление преподавателей» - доступ имеет только секретарь
* «Изменение преподавателей» - доступ имеет только секретарь;
* «Удаление преподавателей» - доступ имеет только секретарь;
* «Просмотр преподавателей» - доступ имеют секретарь и директор.

Диаграмма переходов состояний

Диаграмма переходов состояний — это схема переходов и состояния, специальная техника для перехода технического задания из одного статуса в другой. С ее помощью пользователь в наглядной форме может просматривать переход продукта из одной стадии в другую.

Диаграмма переходов состояний была разработана на основании технического задания и диаграммы вариантов использования с помощью сайта «diagrameditor.com».

На рисунке 4 представлена диаграмма переходов состояний для ДоСиП.



1. Диаграмма переходов состояний

После запуска приложения, появляется главное окно, содержащее страницу «Авторизация». На этой странице пользователю необходимо ввести логин и пароль, после чего нажать на кнопку «Войти». 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 главной странице пользователь может нажать на кнопку «Выход», после чего будет отображена страница «Авторизация», или нажать кнопку «Студенты» или «Преподаватели». При нажатии на данные кнопки отобразится страница, соответствующая нажатой кнопке. Так как функциональность данных страниц идентична, далее будет описана только страница «Студенты».

На странице «Студенты» пользователь может использовать сортировку, при помощи нажатия на столбцы таблицы, фильтрацию, выбирая пункты в выпадающем списке, найти студента с определёнными данными, введя эти данные в поле «Поиск», добавить в таблицу нового студента, а также изменить или удалить данные уже имеющегося в таблице. При добавлении и изменении данных, будет отображена страница «Добавление/изменение студентов».

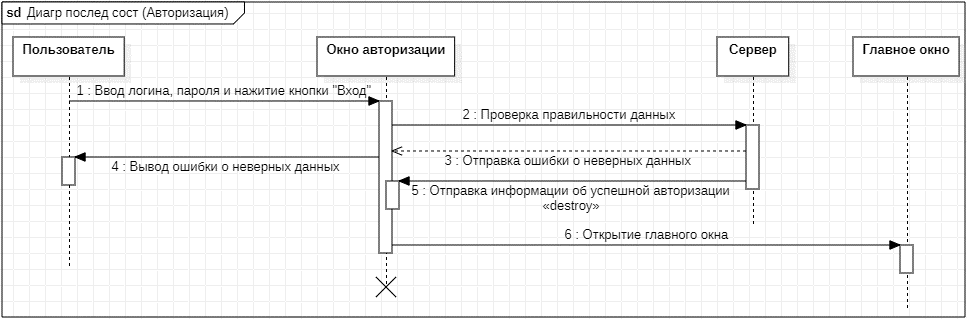
На странице «Добавление/изменение студентов» пользователь может вернуться назад без сохранения изменений, нажав на кнопку «Отмена», или ввести данные и нажать на кнопку «Сохранить», чтобы введённые данные внеслись в базу данных.

Диаграмма последовательностей состояний

Диаграмма последовательности — UML-диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта (создание-деятельность-уничтожение некой сущности) и взаимодействие актеров (действующих лиц) информационной системы в рамках прецедента.

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

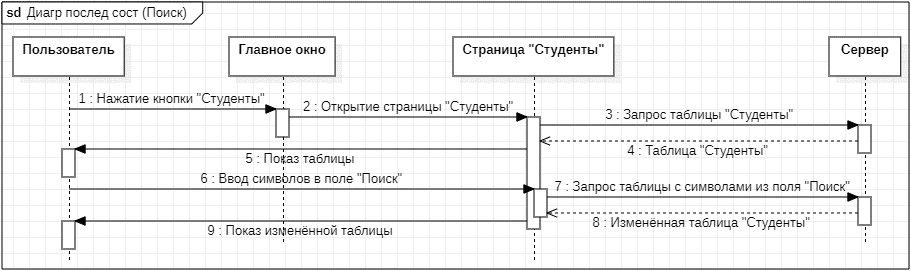
На рисунке 6 представлена диаграмма последовательностей состояний для подсистемы «Авторизация».



1. 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.

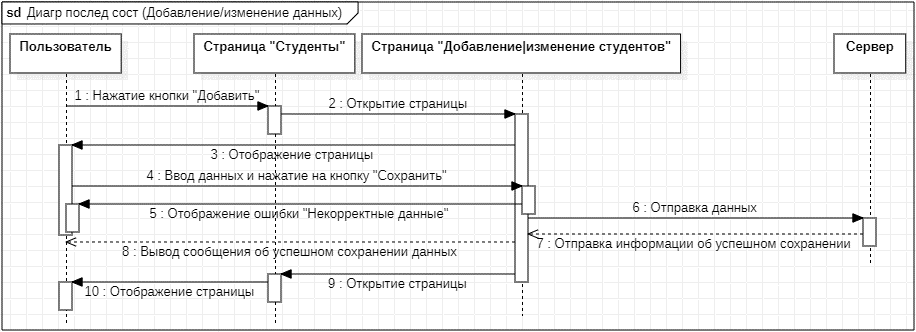
На рисунке 7 представлена диаграмма последовательностей состояний для подсистемы «Поиск».



1. Диаграмма последовательностей состояний для подсистемы «Поиск»

С главной страницы пользователь может перейти на страницу «Студенты» или «Преподаватели», нажав на соответствующие кнопки. Функционал данных страниц схож, поэтому будет описана только одна из них – «Студенты». На данной странице доступен активный поиск по хранящейся в базе данных информации о студенте. При вводе символов в поле «Поиск», система отправляет запрос на сервер, с введёнными символами, и возвращает изменённую таблицу, которая отображается на странице.

На рисунке 8 представлена диаграмма последовательностей состояний для подсистемы «Добавление/изменение данных студента».



1. Диаграмма последовательностей состояний для подсистемы «Добавление/изменение данных студента»

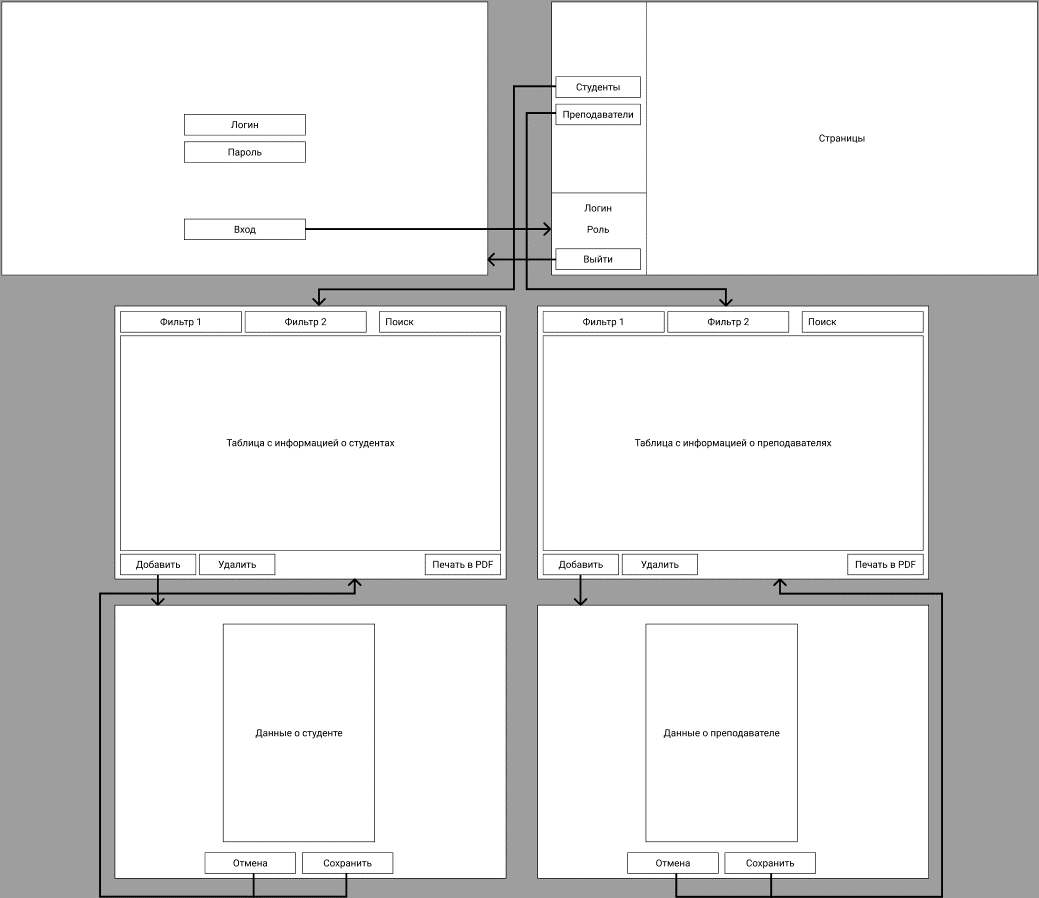
На странице «Добавление/изменение данных студента» пользователь может добавить новые данные о студенте, если до этого была нажата кнопка «Добавить», или изменить уже существующие данные, выбрав строку в таблице.

Wireframe будущего приложения

Wireframes — это организация полного функционала конечного продукта, в виде структуры с представлением элементов интерфейса и навигации, их взаимодействия друг с другом. Как правило, в wireframes не используется типографика, цвет и любые другие графические элементы оформления, так как основное внимание уделяется функциональности, поведению и содержанию. Wireframes фокусируется на том, как экраны взаимодействуют между собой, а не на то, как они выглядят.

Wireframe был разработан на основании технического задания с помощью сайта figma.com.

На рисунке 9 представлен wireframe будущего приложения.



1. Wireframe будущего приложения

Приложение содержит одно главное окно и шесть связанных с ним страниц.

Для приложения были спроектированы следующие страницы:

* Главная страница,
* «Авторизация»,
* «Студенты»,
* «Добавление/изменение студентов»,
* «Преподаватели»,
* «Добавление/изменение преподавателей».

Главное окно содержит frame, размером во всё окно, в котором будут отображаться разработанные страницы.

На странице «Авторизация» пользователь может авторизоваться в системе, после чего отобразится главная страница приложения.

На главной странице пользователь может:

* нажать на кнопку «Выход», чтобы выйти из своего аккаунта, после нажатия отобразится страница «Авторизация»;
* нажать на кнопку «Студенты», чтобы посмотреть данные о студентах, после нажатия отобразится страница «Студенты»;
* нажать на кнопку «Преподаватели», чтобы посмотреть данные о преподавателях, после нажатия отобразится страница «Преподаватели».

На странице «Студенты», как и на странице «Преподаватели», пользователь может:

* сортировать данные, с помощью нажатия на столбцы в таблице;
* фильтровать данные, с помощью выпадающих списков;
* осуществлять поиск студентов или преподавателей с определёнными данными, введёнными в поле «Поиск»;
* нажать на кнопку «Добавить» или выбрать строку со студентом или преподавателем, чтобы перейти на страницу «Добавление/изменение студентов» или «Добавление/изменение преподавателей» соответственно;
* нажать на кнопку «Удалить» для удаления выбранного студента или преподавателя из базы данных;
* нажать на кнопку «Печать в PDF», чтобы сформировался PDF документ, который можно распечатать.

На странице «Добавление/изменение студентов» и на странице «Добавление/изменение преподавателей», пользователь может:

* ввести данные о студенте или преподавателе и нажать на кнопку «Сохранить», после чего данные будут внесены в базу данных и отобразится предыдущая страница с таблицей.
* вернуться на предыдущую страницу, нажав на кнопку «Отмена».

### Текстовые сценарии

Для того, чтобы в будущем проверить приложение на корректное функционирование, были написаны пять тестов для подсистемы «Изменение информации о студенте», приведённые в таблицах 1-5.

1. Тестовый пример №1

|  |  |
| --- | --- |
| Тестовый пример # | 1 |
| Приоритет тестирования | Высокий |
| Заголовок/название теста | Проверка изменения серии паспорта |
| Краткое изложение теста | При проверке серии паспорта, необходимым условием для выполнения теста является ввод только цифр с ограничением по количеству 6 цифр. |
| Этапы теста | Ввод серии паспорта в соответствующее поле для ввода.  Нажатие на кнопку «Сохранить».  Получение ответа о сохранении. |
| Тестовые данные | Серия паспорта = «123456» |
| Ожидаемый результат | Ожидаемым результатом при вводе тестовых данных является сообщение об успешном сохранении информации. |
| Фактический результат |  |
| Статус |  |
| Предварительное условие | Предварительным условием для выполнения тестового примера №1 является стопроцентная вероятность выполнения ожидаемого результата. |
| Постусловие | База данных связанная непосредственно с приложением должна содержать информации о хотя бы 1 студенте. |

1. Тестовый пример №2

|  |  |
| --- | --- |
| Тестовый пример # | 2 |
| Приоритет тестирования | Высокий |
| Заголовок/название теста | Проверка изменения серии паспорта студента |
| Краткое изложение теста | При проверке серии паспорта, необходимым условием для выполнения теста является ввод только цифр с ограничением по количеству 6 штук. |
| Этапы теста | Ввод серии паспорта в соответствующее поле для ввода.  Нажатие на кнопку «Сохранить».  Получение ответа о сохранении. |
| Тестовые данные | Серия паспорта = «1234567» |
| Ожидаемый результат | Ожидаемым результатом при вводе тестовых данных является сообщение с ошибкой количества символов. |
| Фактический результат |  |
| Статус |  |
| Предварительное условие | Предварительным условием для выполнения тестового примера №2 является стопроцентная вероятность выполнения ожидаемого результата. |
| Постусловие | База данных связанная непосредственно с приложением должна содержать информации о хотя бы 1 студенте. |

1. Тестовый пример №3

|  |  |
| --- | --- |
| Тестовый пример # | 3 |
| Приоритет тестирования | Высокий |
| Заголовок/название теста | Проверка изменения имени студента |
| Краткое изложение теста | При проверке имени, необходимым условием для выполнения теста является ввод хотя бы двух символов. |
| Этапы теста | Ввод имени в соответствующее поле для ввода.  Нажатие на кнопку «Сохранить».  Получение ответа о сохранении. |
| Тестовые данные | Имя = «» |
| Ожидаемый результат | Ожидаемым результатом при вводе тестовых данных является сообщение с ошибкой пустой строки имени. |
| Фактический результат |  |
| Статус |  |
| Предварительное условие | Предварительным условием для выполнения тестового примера №3 является стопроцентная вероятность выполнения ожидаемого результата. |
| Постусловие | База данных связанная непосредственно с приложением должна содержать информации о хотя бы 1 студенте. |

1. Тестовый пример №4

|  |  |
| --- | --- |
| Тестовый пример # | 4 |
| Приоритет тестирования | Высокий |
| Заголовок/название теста | Проверка изменения имени студента |
| Краткое изложение теста | При проверке имени, необходимым условием для выполнения теста является ввод хотя бы двух символов. |
| Этапы теста | Ввод имени в соответствующее поле для ввода.  Нажатие на кнопку «Сохранить».  Получение ответа о сохранении. |
| Тестовые данные | Имя = «Андрей» |
| Ожидаемый результат | Ожидаемым результатом при вводе тестовых данных является сообщение об успешном сохранении информации. |
| Фактический результат |  |
| Статус |  |
| Предварительное условие | Предварительным условием для выполнения тестового примера №4 является стопроцентная вероятность выполнения ожидаемого результата. |
| Постусловие | База данных связанная непосредственно с приложением должна содержать информации о хотя бы 1 студенте. |

1. Тестовый пример №5

|  |  |
| --- | --- |
| Тестовый пример # | 5 |
| Приоритет тестирования | Высокий |
| Заголовок/название теста | Проверка изменения отчества студента |
| Краткое изложение теста | При проверке отчества, необходимым условием для выполнения теста является ввод хотя бы двух символов или пустая строка. |
| Этапы теста | Ввод отчества в соответствующее поле для ввода.  Нажатие на кнопку «Сохранить».  Получение ответа о сохранении. |
| Тестовые данные | Отчество = «» |
| Ожидаемый результат | Ожидаемым результатом при вводе тестовых данных является сообщение об успешном сохранении информации. |
| Фактический результат |  |
| Статус |  |
| Предварительное условие | Предварительным условием для выполнения тестового примера №5 является стопроцентная вероятность выполнения ожидаемого результата. |
| Постусловие | База данных связанная непосредственно с приложением должна содержать информации о хотя бы 1 студенте. |

# Эксплуатационная документация

## Описание основных элементов подсистемы

Подсистема «ДоСиП» имеет 3 основных элемента:

* Авторизация;
* Хранение данных о студентах и преподавателях;
* Изменение данных о студентах и преподавателях.

## Руководство пользователя

Для того, чтобы пользователи быстрее разобрались в интерфейсе приложения, а также знать, что делать при возникновении ошибок, было разработано «Руководство пользователя»

Руководство пользователя

### Введение

#### Область применения

Требования настоящего документа применяются при:

* предварительных комплексных испытаниях;
* опытной эксплуатации;
* приемочных испытаниях;
* промышленной эксплуатации.

#### Краткое описание возможностей

Система автоматизации хранения данных о студентах и преподавателях (ДоСиП) предназначена для автоматизации хранения данных о студентах и преподавателях.

#### Уровень подготовки пользователя

Пользователь ДоСиП должен иметь опыт работы с ОС MS Windows 10 и знать соответствующую предметную область;

### Описание операций

#### Выполняемые функции и задачи

ДоСиП выполняет функции и задачи, приведенные в таблице 1:

1. Функции и задачи ДоСиП

|  |  |
| --- | --- |
| Функции | Задачи |
| Обеспечивает авторизацию пользователей | Авторизация пользователей в приложении |
| Обеспечивает хранение и изменение данных о студентах | Добавление информации о студентах в базу данных |
| Изменение информации о студентах в базе данных |
| Удаление информации о студентах из базы данных |
| Обеспечивает хранение и изменение данных о преподавателях | Добавление информации о преподавателях в базу данных |
| Изменение информации о преподавателях в базе данных |
| Удаление информации о преподавателях из базы данных |

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

В таблицах 2-5 приведено описание пользовательских операций для выполнения каждой из задач.

1. 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 «ДоСиП» 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 | 15-30 секунд |

1. Задача: «Добавление информации о студентах в базу данных»

|  |  |
| --- | --- |
| Условия, при соблюдении которых возможно выполнение операции | * компьютер пользователя подключен к серверу колледжа; * сервер колледжа функционирует в штатном режиме. |
| Подготовительные действия | Авторизация в ДоСиП. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Студенты» в меню слева; 2. нажать на кнопку «Добавить» в нижней части окна; 3. заполнить необходимые данные; 4. нажать на кнопку «Сохранить». |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | 5-10 минут |

1. Задача: «Изменение информации о студентах в базе данных»

|  |  |
| --- | --- |
| 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 в ДоСиП. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Студенты» в меню слева; 2. выбрать студента в таблице; 3. нажать на кнопку «Изменить» в нижней части окна; 4. изменить необходимые данные; 5. нажать на кнопку «Сохранить». |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | 1-10 минут |

1. Задача: «Удаление информации о студентах из базы данных»

|  |  |
| --- | --- |
| 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 в ДоСиП. |
| Основные действия в требуемой последовательности | 1. нажать на кнопку «Студенты» в меню слева; 2. выбрать студента в таблице; 3. нажать на кнопку «Удалить» в нижней части окна; 4. нажать на кнопку «Ок» во всплывающем окне. |
| Заключительные действия | Не требуются |
| Ресурсы, расходуемые на операцию | 5-10 секунд |

Задачи «Добавление информации о преподавателях в базу данных», «Изменение информации о преподавателях в базе данных» и «Удаление информации о преподавателях из базы данных» выполняются по аналогии с предыдущими задачами.

### Аварийные ситуации

В случае возникновения ошибок при работе ДоСиП, не описанных в таблице 6, необходимо обращаться к администратору ДоСиП.

1. Аварийные ситуации

|  |  |
| --- | --- |
| Ошибка | Требуемые действия пользователя при возникновении ошибки |
| Нет подключения к серверу | Обратиться к администратору ДоСиП. |
| Неверный логин или пароль | Произвести 3 попытки входа. Если ошибка осталась, обратиться к администратору ДоСиП. |
| Введены некорректные данные в поле «П» | Отредактировать данные в поле «П», следуя инструкции в ошибке. |
| Выберите строку для «Д» | Выбрать строку в таблице для совершения действия «Д». |

ЗАКЛЮЧЕНИЕ

В процессе выполнения данной производственной преддипломной практики были закреплены знания и навыки проектирования и разработки технического задания, диаграмм, wireframe, тестовых сценариев, разработки базы данных в SQLite, десктопного приложения на WPF и языка программирования C#.

В самом начале проекта были поставлены следующие задачи:

1. Изучить инструктаж по охране труда и технике безопасности, распределение по рабочим местам.
2. Ознакомиться со структурой и характером деятельности предприятия.
3. Собрать материалы для составления технического задания по теме дипломной работы:
   1. Анализ деятельности организации УЭТК СГУ;
   2. Анализ программных средств, необходимых для разработки модуля для учета информации о студентах и сотрудниках УЭТК СГУ;
   3. Описание этапов разработки подсистемы;
4. Разработать эксплуатационную документацию по модулю для учета информации о студентах и сотрудниках УЭТК СГУ:
   1. Описание основных элементов подсистемы;
   2. Руководство пользователя.

Все данные задачи были успешно реализованы.

Практическая работа состоит из четырёх глав и заключения.

В первой главе практической работы «Ознакомление со структурой и характером деятельности предприятия» была изучена структура колледжа УЭТК и университета СГУ.

Во второй главе «Анализ и разработка системы» было проанализирована деятельность колледжа УЭТК и изучено программное обеспечение, используемое для учёта студентов и преподавателей. Проанализированы программные средства, которые будут использоваться при проектировании и разработке. Подробно описаны основные этапы проектирования и разработки базы данных для работы приложения и самого приложения.

В третей главе «Эксплуатационная документация» описаны основные этапы проектирования и разработки программного обеспечения для автоматизации выполнения заказов изделий.

В результате работы над программным обеспечением были созданы десктопный программный модуль и его упрощённая мобильная версия.

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: 27.11.2022)
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: 27.11.2022). – 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: 25.11.2022). – 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: 27.11.2022).
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: 05.12.2022).
7. Сведения об образовательной организации. – URL: <https://sutr.ru/sveden/> (дата обращения: 26.11.2022).
8. Стандарты и шаблоны для ТЗ на разработку ПО / Хабр. – URL: https://habr.com/ru/post/328822/ (дата обращения: 07.12.2022).
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: 07.12.2022).
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: 26.11.2022).
11. Использование диаграммы вариантов использования UML при проектировании программного обеспечения. – URL: https://habr.com/ru/post/566218/ (дата обращения: 26.11.2022).
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: 26.11.2022).
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: 26.11.2022).
14. State & Transition Diagram — что это и как применять / Хабр. – URL: https://habr.com/ru/post/548192/ (дата обращения: 26.11.2022).
15. Краткий путеводитель по методологиям и нотациям описания и моделирования бизнес-процессов. Часть 4. – URL: https://infostart.ru/1c/articles/1451560/ (дата обращения: 26.11.2022).
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: 26.11.2022).
17. Проектирование программного обеспечения / Хабр. – URL: https://habr.com/ru/post/74330/ (дата обращения: 26.11.2022).
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: 27.11.2022).
19. Wireframes в разработке: особенности и преимущества / Хабр. – URL: https://habr.com/ru/post/690598/ (дата обращения: 27.11.2022).
20. Free Online Diagram Editor. – URL: https://www.diagrameditor.com/ (дата обращения: 26.11.2022).
21. Краткое руководство по работе с SQLite | Timeweb Cloud – URL: <https://timeweb.cloud/tutorials/sqlite/rukovodstvo-po-nastrojke-sqlite> (дата обращения: 26.04.2023)
22. C# и WPF | Компоновка – URL: <https://metanit.com/sharp/wpf/4.php> (дата обращения: 24.04.2023).
23. C# и WPF | Grid – URL: <https://metanit.com/sharp/wpf/4.2.php> (дата обращения: 24.04.2023).
24. C# и WPF | StackPanel – URL: <https://metanit.com/sharp/wpf/4.4.php> (дата обращения: 24.04.2023).
25. C# и WPF | WrapPanel – URL: <https://metanit.com/sharp/wpf/4.6.php> (дата обращения: 24.04.2023).
26. C# и WPF | Canvas – URL: <https://metanit.com/sharp/wpf/4.7.php> (дата обращения: 24.04.2023).
27. C# и WPF | Свойства компоновки элементов – URL: <https://metanit.com/sharp/wpf/4.8.php> (дата обращения: 24.04.2023).
28. Связи между таблицами базы данных / Хабр. – URL: https://habr.com/ru/post/488054/ (дата обращения: 26.11.2022).
29. Figma. – URL: https://www.figma.com/ (дата обращения: 27.11.2022).
30. Документ "Руководство пользователя" - RuGost. – URL: http://www.rugost.com/index.php?option=com\_content&view=article&id=181&catid=27&Itemid=64 (дата обращения: 07.12.2022).
31. 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: 07.12.2022).