**МИНИСТЕРСТВО ЭНЕРГЕТИКИ, ПРОМЫШЛЕННОСТИ И СВЯЗИ СТАВРОПОЛЬСКОГО КРАЯ**

Государственное бюджетное профессиональное образовательное учреждение «Ставропольский колледж связи имени Героя Советского Союза В.А. Петрова» (ГБПОУ СКС)

Цикловая комиссия вычислительной техники,

экономики и менеджмента

**КУРСОВАЯ РАБОТА**

**по ПМ.02 «Осуществление интеграции программных модулей»**

**МДК 02.01 «Технология разработки программного обеспечения»**

Тема: Разработка информационной системы «Спортивная школа»

Выполнил Ковалёв Александр Кириллович

4 курс, ИП-202\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | Руководитель |
|  | Лукьянова Светлана Павловна |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Ставрополь, 2024

**«Ставропольский колледж связи**

**имени Героя Советского Союза В.А. Петрова»**

УТВЕРЖДАЮ

Председатель цикловой комиссии

\_\_\_\_\_\_\_\_\_\_\_\_\_\_/И.В. Ерёмина

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

**ЗАДАНИЕ**

**для курсовой работы (проекта)**

Студенту *Фамилия Имя Отчество, курс, группа*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель Лукьянова Светлана Павловна\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тема \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Разделы, подразделы и их содержание** | **Сроки выполнения** |
| 1 | Введение | 00.00.2024 |
| 2 | Теоретические основы разработки программного обеспечения | 00.00.2024 |
| 3 | Техническое задание на разработку | 00.00.2024 |
| 4 | Функциональная модель информационной системы | 00.00.2024 |
| 5 | Объектно-ориентированное проектирование системы | 00.00.2024 |
| 6 | Создание информационной системы | 00.00.2024 |
| 7 | Тестирование программного продукта | 00.00.2024 |
| 8 | Техническая документация | 00.00.2024 |
| 9 | Заключение | 00.00.2024 |
| 10 | Список использованных источников | 00.00.2024 |
| 11 | Приложения | 00.00.2024 |

Преподаватель-руководитель курсовой работы (проекта)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**/С.П. Лукьянова/

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc156296148)

[1 Теоретические основы разработки программного обеспечения 4](#_Toc156296149)

[2 Техническое задание на разработку 5](#_Toc156296150)

[3 Функциональная модель информационной системы 6](#_Toc156296151)

[4 Объектно-ориентированное проектирование системы 8](#_Toc156296152)

[4.1 Модели вариантов использования системы 8](#_Toc156296153)

[4.2 Диаграмма классов 8](#_Toc156296154)

[4.3 Диаграммы деятельности 9](#_Toc156296155)

[4.4 Диаграмма последовательности 10](#_Toc156296156)

[4.5 Диаграмма кооперации 11](#_Toc156296157)

[4.6 Диаграмма состояний 12](#_Toc156296158)

[5 Создание информационной системы 14](#_Toc156296159)

[5.1 Разработка интерфейса программного продукта 14](#_Toc156296160)

[5.2 Разработка программного кода системы 15](#_Toc156296161)

[5.3 Разработка базы данных 15](#_Toc156296162)

[5.4 Справочная система 16](#_Toc156296163)

[6 Тестирование программного продукта 17](#_Toc156296164)

[7 Техническая документация 19](#_Toc156296165)

[ЗАКЛЮЧЕНИЕ 20](#_Toc156296166)

[Приложение А 22](#_Toc156296167)

[Приложение Б 28](#_Toc156296168)

[ПРИЛОЖЕНИЕ В 31](#_Toc156296169)

# **Введение**

Информационные системы позволяют быстро обрабатывать большие объёмы информации.

Внедряя информационную систему на предприятие, мы повышаем точность конечных данных, сокращаем время выполнения работы, а также облегчаем труд специалистов.

Точность конечных данных будет увеличена из-за того, что влияние человеческого фактора будет уменьшено, так как большинство расчётов будет проводиться не сотрудником вручную, а автоматически информационной системой.

Время выполнения работы сократится благодаря тому, что поступающие данные будут сразу обрабатываться информационной системой и не будет необходимости проводить расчёты вручную, или, если участие человека необходимо, то работа будет выполняться с помощью удобного интерфейса, а её результат быстро доставлен туда, куда необходимо, уменьшая давление на бюрократический аппарат рассматриваемого предприятия.

Облегчение труда специалистов является пунктом, который непосредственно вытекает из двух предыдущих. Как уже было сказано, благодаря внедрению информационной системы сотрудникам будет не нужно проводить большинство расчётов самостоятельно, а также, они смогут избежать излишнего заполнения различных бумаг.

Цель курсового проекта: используя современные средства проектирования и разработки реализовать все этапы жизненного цикла программных продуктов.

# **Теоретические основы разработки программного обеспечения**

Предметной областью, данной курсовой работы, является разработка информационной системы для спортивной школы.

Данная информационная система разрабатывается для того, чтобы достигнуть следующих целей:

* Увеличение эффективности работы сотрудников, за счёт введения автоматизации и уменьшения количества задач, которые раньше нужно было выполнять вручную.
* Улучшение экономических показателей предприятия, за счёт автоматизации обработки информации и ускорения времени получения новой информации.
* Улучшение качества обслуживания клиентов, за счёт того, что большинство запросов смогут быть обработаны системой.
* Увеличение точности предоставляемой и хранимой информации.

Разрабатываемая информационная система сможет обеспечить обработку и хранение следующей информации:

* Информация о курсах.
* Информация о сотрудниках.
* Заключённые договора (С клиентами, о предоставляемых услугах; о закупках и т. п.).

К данной информационной системе будут иметь доступ все сотрудники предприятия, однако, имея различные дополнительные возможности и ограничения в связи с занимаемыми ими должностями.

# **Техническое задание на разработку**

В техническом задании описаны необходимые для реализации функции и основные требования к программе.

Также в техническом задании указана структура базы данных и необходимая документация.

Техническое задание на разработку программного продукта «Информационная система спортивной школы» приведено в приложении А.

# **Функциональная модель информационной системы**

Приведённые ниже диаграммы (рисунки 1, 2 и 3) были созданы с помощью приложения Ramus.

На рисунке 1 изображена контекстная диаграмма ИС «Спортивная школа». Система принимает на вход информацию о договорах, клиентах и запросах, а на выходе выдаёт отчёт. Работу системы корректируют законы, указы и инструкции. А взаимодействуют с системой сотрудники.

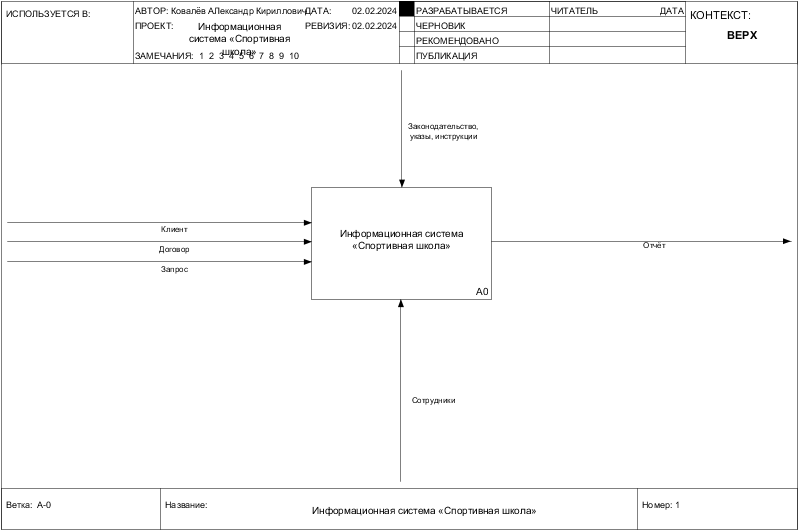


Рисунок 1 – Контекстная диаграмма информационной системы

На рисунке 2 изображена функциональная диаграмма первого уровня. Она отображает последовательность действий, выполняемых системой. Сотрудник вносит данные о клиенте, после чего система проверяет корректность введённых данных. Сотрудник при необходимости может задать запрос к системе на выборку данных клиентах и договорах по определённым критериям.

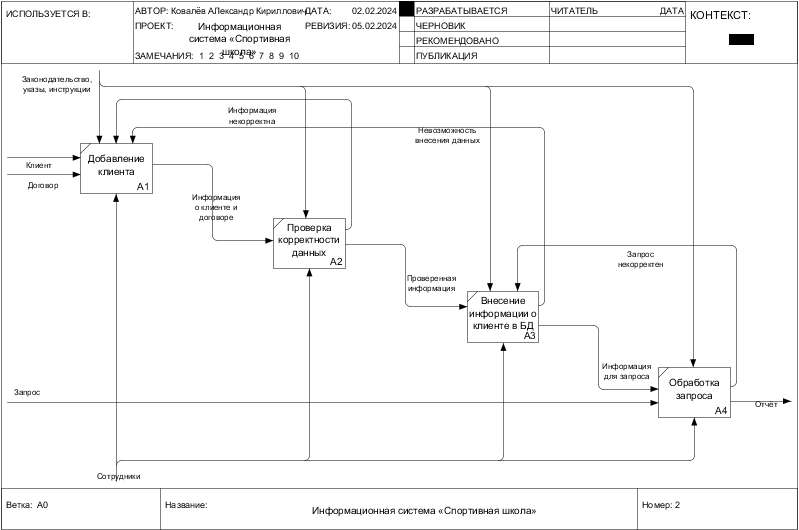


Рисунок 2 – Функциональная диаграмма первого уровня

На рисунке 3 изображена DFD диаграмма для информационной системы. На ней видно, что данные о документах и клиентах загружаются в отдельные таблицы базу данных.

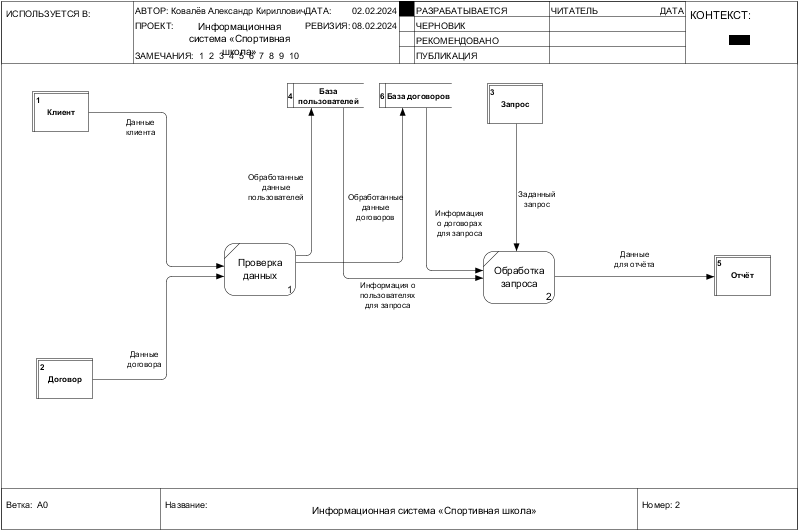


Рисунок 3 – DFD диаграмма для информационной системы

# **Объектно-ориентированное проектирование системы**

## 4.1 Модели вариантов использования системы.

## В ходе анализа для проектируемой информационной системы было выделено 2 действующих лица. Для каждого из них были выделены прецеденты.

Полученная диаграмма вариантов использования ИС «Спортивная школа» показана на рисунке 4.

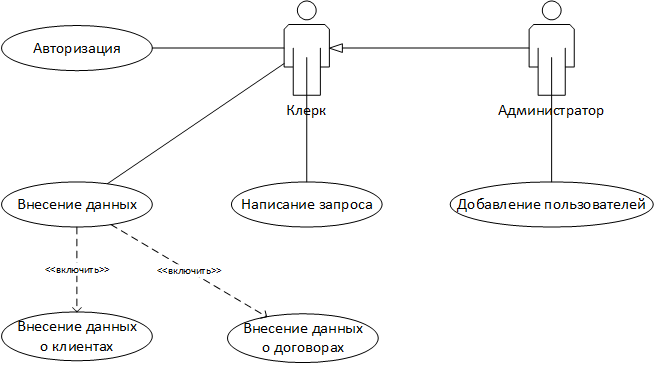


Рисунок 4 – Диаграмма вариантов использования информационной системы

Клерк может авторизироваться в системе, написать запрос, а также вносить данные о клиентах и договорах. Администратор имеет расширенные возможности по сравнению с клерком, а именно может добавлять пользователей и должен отвечать за поддержание работоспособности базы данных.

## 4.2 Диаграмма классов.

Основной диаграммой проекта является диаграмма классов. Статистическая модель организации данных. Основная диаграмма классов представлена на рисунке 5.

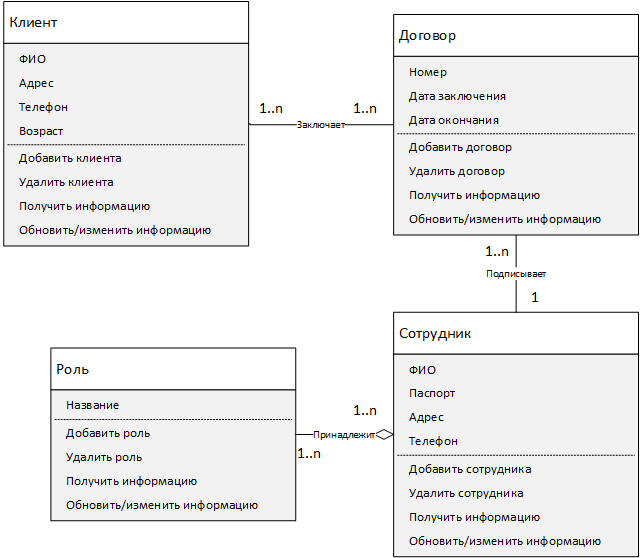


Рисунок 5 – Диаграмма классов

В системе можно выделить четыре класса: клиент, договор, сотрудник, роль. Со всеми классами доступно одинаковое взаимодействие, а именно: добавление, удаление, получение, обновление информации.

Если рассматривать взаимоотношение классов, то можно выделить следующие: клиент может заключить несколько договоров, а в одном договоре может быть указанно несколько клиентов; одна роль может принадлежать множеству сотрудников, в то же время несколько ролей может быть у одного сотрудника; и последним можно выделить то, что один сотрудник может подписать несколько договоров, однако только один сотрудник подписывает договор.

## 4.3 Диаграммы деятельности.

Диаграмма деятельности, составленная для информационной системы, представлена на рисунке 6.

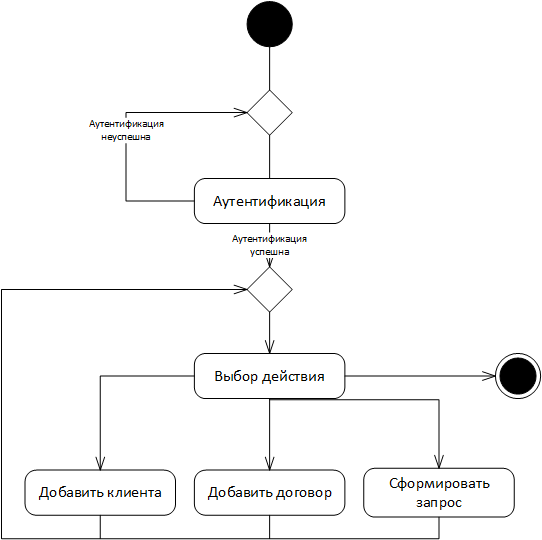


Рисунок 6 – Диаграмма деятельности

На данной диаграмме можно наблюдать основные возможности клерка при взаимодействии с системой. Сперва, для входа в систему необходимо пройти процесс аутентификации. В случае провала пользователю необходимо будет вновь ввести свои данные. При удачном исходе пользователь может выполнить одно из следующих действий: добавить клиента, добавить договор или же составить запрос к системе. По завершении необходимости взаимодействия с системой пользователь выходит из приложения.

## 4.4 Диаграмма последовательности.

Диаграмма последовательности, составленная для варианта использования «Ввод данных», представлена на рисунке 7.

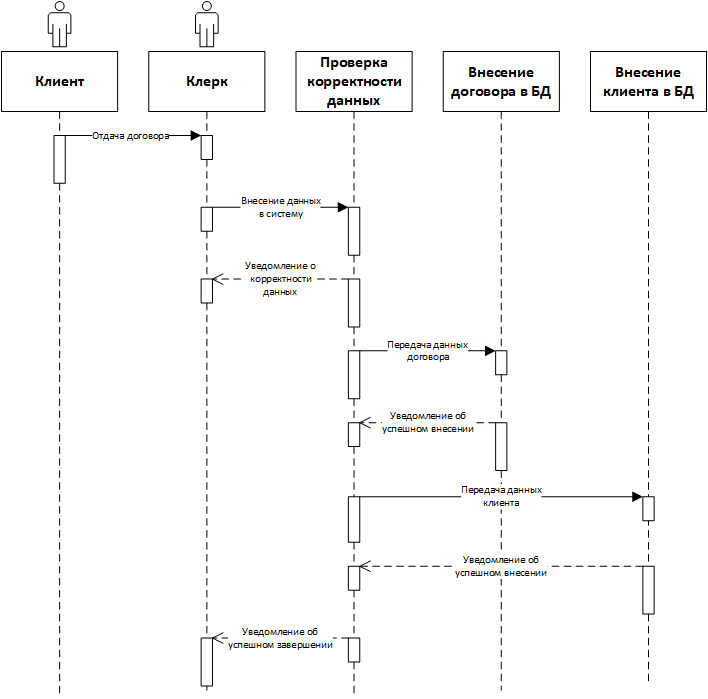


Рисунок 7 – Диаграмма последовательности

Данная диаграмма отображает процесс упомянутый в предыдущем пункте в процессе. Клиент отдаёт данные сотруднику, после чего сотрудник вносит данные в систему. Система проверяет введённые данные, и поочерёдно вносит их в таблицы «Договор» и «Клиент». В конце сотрудник уведомляется об успешном добавлении данных.

## 4.5 Диаграмма кооперации.

Диаграмма кооперации, составленная для варианта использования «Составление запросов», представлена на рисунке 8.

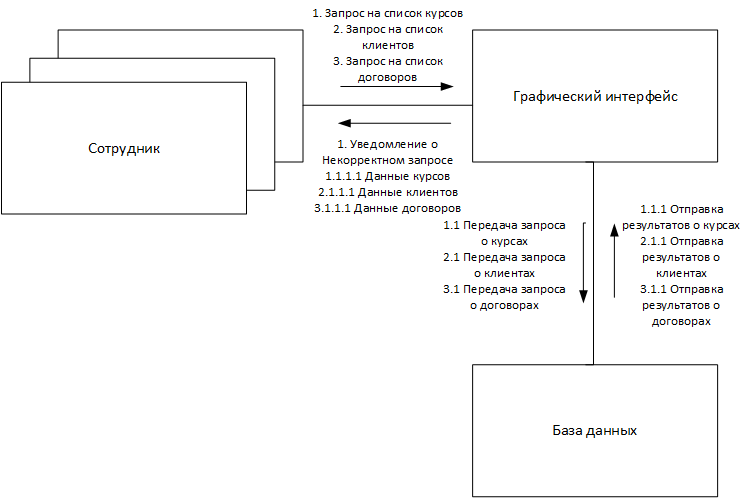


Рисунок 8 – Диаграмма кооперации

На данной диаграмме видно какие запросы можно составить к системе. Пользователь может получить данные о договорах, клиентах и курсах. Так же система может сообщить пользователю, что введённый им запрос ошибочен.

## 4.6 Диаграмма состояний.

Диаграмма состояний, составленная для варианта использования «Внесение данных», представлена на рисунке 9.

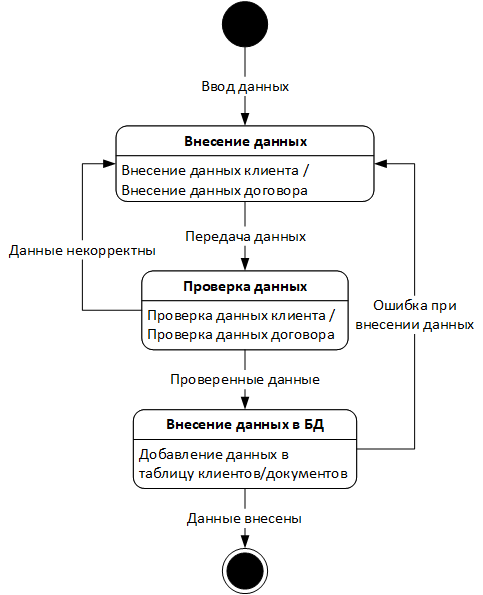


Рисунок 9 – Диаграмма состояний

На данной диаграмме можно наблюдать процесс внесения данных с точки зрения состояний. Здесь можно увидеть, что процесс внесения данных состоит из трёх состояний:

1. Ввод данных пользователем.
2. Проверка данных. Данное состояние может отправить систему, как и в предыдущее состояние, при ошибке в данных, так и перевести её в следующее.
3. Внесение данных в БД. Данное состояние может завершить весь процесс или, при неудачном внесении данных, система возвращается в первое состояние.

На приведённых ранее диаграммах UML можно увидеть большую часть работы системы. Как становиться понятно из них основная цель системы – это хранение информации и выдача её при необходимости.

# **Создание информационной системы**

## 5.1 Разработка интерфейса программного продукта.

В ходе разработке было принято решение использовать в системе графический интерфейс.

Графический интерфейс – это, по моему мнению, самый удобный способ взаимодействия пользователя с системой.

У графического интерфейса можно выделить следующие преимущества по сравнению с другими видами интерфейса:

* Понятность. Графический интерфейс лёгок для понимания.
* Широкая аудитория. Из-за того, что графический интерфейс прост для понимания количество людей, которое может пользоваться приложением, не проходя специальную подготовку возрастает.
* Скорость работы. Поскольку пользователю не нужно прописывать команды, а достаточно нажимать на кнопки.

На рисунке 10 представлен интерфейс разработанного программного продукта.

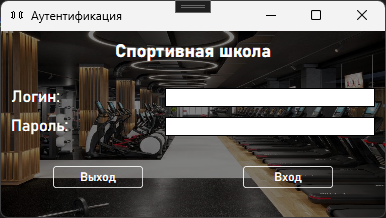


Рисунок 10 – Интерфейс приложения «Спортивная школа»

## 5.2 Разработка программного кода системы.

Разработка приложения производилась на платформе .Net Microsoft на языке программирования С#

Мой выбор языка программирования пал на C# нескольким причинам, вот некоторые из них:

* C# имеет сборщик мусора, что означает, что беспокоится об освобождении выделенной динамической памяти не нужно.
* Простота. Язык C# не сложен для понимания и достаточно прост в освоении.
* Статическая типизация. Для кого-то это может быть недостатком, но по мне это гораздо удобнее, нежели динамическая. При статической типизации переменная не может сменить свой тип, что, по моему мнению, это упрощает разработку и уменьшает количество потенциальных ошибок.
* Объектно-ориентирован.

Часть исходного кода приложения представлена в приложении Б.

## 5.3 Разработка базы данных.

База данных была реализована в Microsoft SQL Server Management Studio.

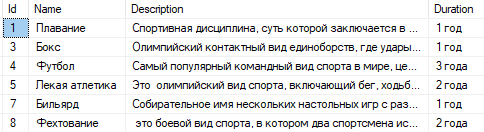


Рисунок 11 - Таблица курсов

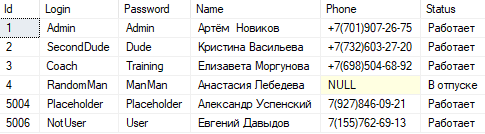


Рисунок 12 - Таблица сотрудников

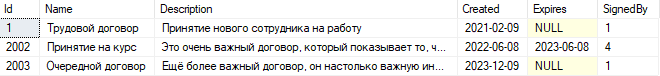


Рисунок 13 - Таблица договоров

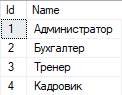


Рисунок 14 - Таблица ролей

## 5.4 Справочная система.

Справочная система для приложения была разработана при помощи специализированной программы Help&Doc. Файл справки, представленный на рисунке 11, был прикреплен к разработанному приложению.

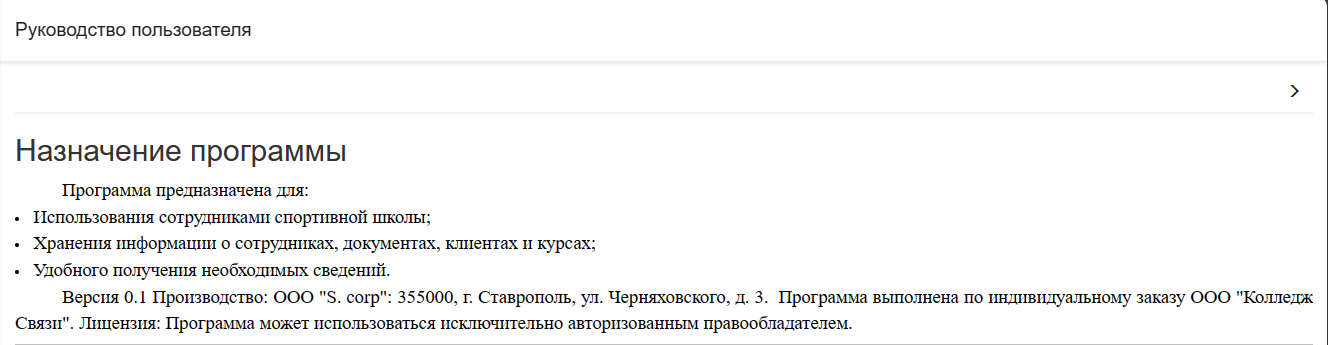


Рисунок 11 – Стартовое окно файла справки

# **Тестирование программного продукта**

Для тестирования программного продукта использовался фреймворк NUnit, который позволяет создавать тест-кейсы, проверяющие нужные функции на корректность их работы.

Основными функциями для составления тест-кейсов стали: DateTimeToDateConverter и FieldsValidator так как они часто используются в программе.

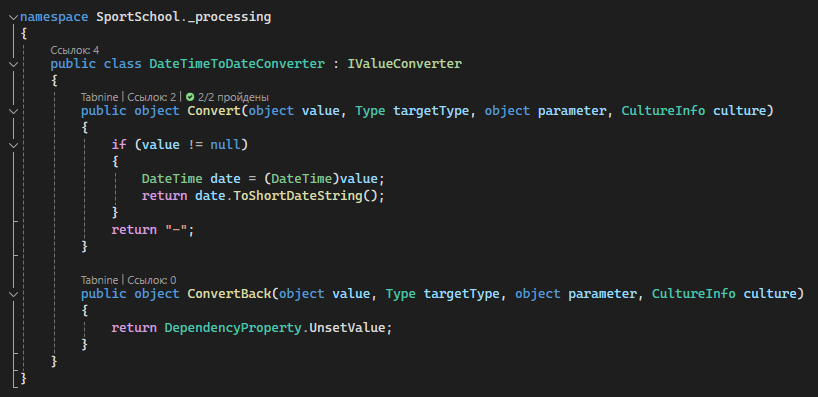


Рисунок 15 - Функция конвертации даты

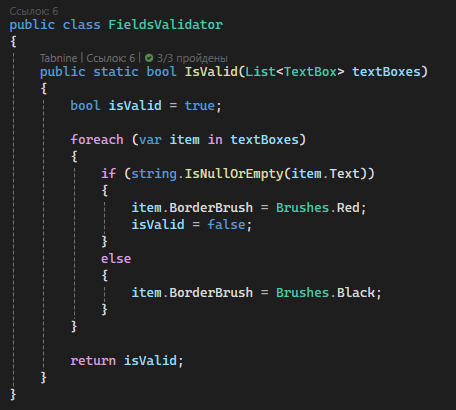


Рисунок 16 - Функция проверки наполненности полей ввода

Ниже, на рисунках 17 и 18 буду приведены тест-кейсы, которые использовались для проверки данных функций, на которых также можно заметить, что все тесты прошли проверку.



Рисунок 17 – Тест-кейс для валидатора

Рисунок 18 – Тест-кейс конвертера даты

# **7** **Техническая документация**

В процессе разработки была разработана программная документация на систему, включающая в себя техническое задание на разработку (Приложение А), руководство пользователя (Приложение В).

**ЗАКЛЮЧЕНИЕ**

В результате выполнения курсовой работы была разработана информационная система для Спортивной школы, предназначенная для использования работниками спортивной школы, позволяющая хранить информацию о сотрудниках, документах и курсах, а также дополнять информацию о них. В результате применения данной информационной системы будет упрощён документооборот, облегчено взаимодействие сотрудников с различными данными, что в следствии приводит к сокращению трудоёмкости и облегчения работы сотрудников.

К недостаткам можно отнести неоптимальную архитектуру приложения, усложняющую восприятие кода и препятствующую его модификации и расширения. Данная проблема будет решаться в приоритете по мере развития программы.

**Список информационных источников**

1. Вендров А.М. Один из подходов к выбору средств проектирования баз данных и приложений. "СУБД", 2003, №3.
2. Зиндер Е.З. Бизнес-реинжиниринг и технологии системного проектирования. Учебное пособие. М., Центр Информационных Технологий, 2000
3. Калянов Г.Н. CASE. Структурный системный анализ (автоматизация и применение). М., "Лори", 1999.
4. Международные стандарты, поддерживающие жизненный цикл программных средств. М., МП "Экономика", 1998
5. Новоженов Ю.В. Объектно-ориентированные технологии разработки сложных программных систем. М., 1999.
6. https://stackoverflow.com/
7. https://learn.microsoft.com/
8. https://www.codeproject.com/
9. https://docs.nunit.org/
10. https://metanit.com/
11. https://www.youtube/
12. https://habr.com/

# **Приложение А**

# **Техническое задание на разработку программы "Спортивная школа"**

**Содержание**

1. Введение

1.1. Наименование программы

1.2. Назначение и область применения

2. Требования к программе

2.1. Требования к функциональным характеристикам

2.2. Требования к надежности

2.2.1. Требования к обеспечению надежного функционирования программы

2.2.2. Время восстановления после отказа

2.2.3. Отказы из-за некорректных действий пользователей системы

3. Условия эксплуатации

3.1. Климатические условия эксплуатации

3.2. Требования к квалификации и численности персонала

3.3. Требования к составу и параметрам технических средств

3.4. Требования к информационной и программной совместимости

3.4.1. Требования к информационным структурам и методам решения

3.4.1.1. Структура баз данных

3.4.1.2. Требования к запросам пользователей данных из базы

3.4.2. Требования к исходным кодам и языкам программирования

3.4.3. Требования к программным средствам, используемым программой

3.4.4. Требования к защите информации и программ

3.5. Специальные требования

4. Требования к программной документации

4.1. Предварительный состав программной документации

5. Технико-экономические показатели

5.1. Экономические преимущества разработки

6. Стадии и этапы разработки

6.1. Стадии разработки

6.2. Этапы разработки

6.3. Содержание работ по этапам

7. Порядок контроля и приемки

7.1. Виды испытаний

7.2. Общие требования к приемке работы

**1. Введение**

**1.1. Наименование программы**

Наименование программы: информационная система «Спортивная школа»

**1.2. Назначение и область применения**

Программа предназначена для обработки и хранения, следующей информации:

1.2.1. Информации о клиентах  
1.2.2. Информации о курсах  
1.2.3. Информации о сотрудниках  
1.2.4. Заключённые договора

Доступ к хранимой информации будет предоставляться через приложение.

**2. Требования к программе**

**2.1. Требования к функциональным характеристикам**

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:  
2.1.1. Разделение пользователей, подключаемых через Веб интерфейс на группы:  
2.1.1.1. Кадровиков  
2.1.1.2. Бухгалтеров  
2.1.1.3. Администраторов

2.1.1.4. Тренеров  
2.1.2. Возможность поиска (фильтрации) документов по заданным критериям

**2.2. Требования к надежности**

**2.2.1. Требования к обеспечению надежного функционирования программы**

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:   
а) организацией бесперебойного питания технических средств;   
б) использованием лицензионного программного обеспечения;   
в) регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;   
г) регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов

**2.2.2. Время восстановления после отказа**

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать 30-ти минут при условии соблюдения условий эксплуатации технических и программных средств.   
Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

**2.2.3. Отказы из-за некорректных действий пользователей системы**

Отказы программы вследствие некорректных действий пользователя при взаимодействии с программой

недопустимы.

**3. Условия эксплуатации**

**3.1. Климатические условия эксплуатации**

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации

**3.2. Требования к квалификации и численности персонала**

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 2 штатных единиц — системный администратор и конечный пользователь программы — оператор. Системный администратор должен иметь высшее профильное образование и сертификаты компании-производителя операционной системы. В перечень задач, выполняемых системным администратором, должны входить:   
а) задача поддержания работоспособности технических средств;   
б) задачи установки (инсталляции) и поддержания работоспособности системных программных средств — операционной системы;   
в) задача установки (инсталляции) программы.   
г) задача создания резервных копий базы данных.

**3.3. Требования к составу и параметрам технических средств**

3.3.1. В состав технических средств должен входить IВМ-совместимый персональный компьютер (ПЭВМ), выполняющий роль сервера, включающий в себя:

3.3.1.1. процессор Core i3 10100 3.6 Hz, не менее;   
3.3.1.2. оперативную память объемом, 4 Гигабайт, не менее;   
3.3.1.3. HDD, 100 Гигабайт, не менее;   
3.3.1.4. операционную систему Windows 8/10/11.

**3.4. Требования к информационной и программной совместимости**

**3.4.1. Требования к информационным структурам и методам решения**

База данных работает под управлением Microsoft SQL Server. Используется много поточный доступ к базе данных. Необходимо обеспечить одновременную работу с программой с той же базой, данной модулей экспорта внешних данных

**3.4.1.1. Структура баз данных**

Таблица пользователей – Employee

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| Id | Int | [уникальное] Идентификатор |
| Login | VarChar(30) | [уникальное] Логин |
| Password | VarChar(30) | Пароль |
| Name | nvarchar (255) | ФИО сотрудника |
| Phone | nvarchar (20) | Телефон сотрудника |
| Status | nvarchar(30) | Статус сотрудника |

Таблица ролей – Role

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| Id | Int | [уникальное] Идентификатор |
| Name | nvarchar(255) | Название роли |

Таблица курсов – Course

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| Id | Int | [уникальное] Идентификатор |
| Name | nvarchar(255) | Название курса |
| Description | nvarchar(max) | Описание курса |
| Duration | nvarchar(255) | Продолжительность курса |

Таблица документов – Document

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| Id | Int | [уникальное] Идентификатор |
| Name | nvarchar(max) | Название документа |
| Description | nvarchar(max) | Описание |
| Created | Date | Дата заключения договора |
| Expires | Date | Дата окончания действия договора |
| SignedBy | Int | Кем подписан |

Таблица ролей – Menu

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| Id | Int | [уникальное] Идентификатор |
| Name | nvarchar(255) | Название меню |

**3.4.1.2. Требования к запросам пользователей данных из базы**

Пользователи и администраторы работают с базой данных через приложение.

Администраторы системы должны иметь возможность редактировать таблицы, перечисленные п. 3.4.1.1. (добавление, редактирование)

Пользователи системы должны иметь возможность производить поиск по таблицам Document и Course, просматривать детальную информацию по каждому результату выборки

**3.4.2. Требования к исходным кодам и языкам программирования**

Для написания программы необходимо использовать язык программирования C#.

**3.4.3. Требования к программным средствам, используемым программой**

Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы Windows 7/8/10/11.

**3.4.4. Требования к защите информации и программ**

Требования к защите информации и программ не предъявляются.

**3.5. Специальные требования**

Программа должна обеспечивать одновременную работу пользователей посредством приложения.

**4. Требования к программной документации**

**4.1. Предварительный состав программной документации**

Состав программной документации должен включать в себя:   
4.1.1. техническое задание;  
4.1.2. программу и методики испытаний;

**5. Технико-экономические показатели**

**5.1. Экономические преимущества разработки**

Ориентировочная экономическая эффективность не рассчитываются. Аналогия не проводится ввиду уникальности предъявляемых требований к разработке.

**6. Стадии и этапы разработки**

**6.1. Стадии разработки**

Разработка должна быть проведена в три стадии:   
1. разработка технического задания;   
2. рабочее проектирование;   
3. внедрение.

**6.2. Этапы разработки**

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.   
На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. разработка программы;   
2. разработка программной документации;   
3. испытания программы.

На стадии внедрения должен быть выполнен этап разработки подготовка и передача программы.

**6.3. Содержание работ по этапам**

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:   
1. постановка задачи;   
2. определение и уточнение требований к техническим средствам;   
3. определение требований к программе;  
4. определение стадий, этапов и сроков разработки программы и документации на неё;   
5. согласование и утверждение технического задания.   
На этапе разработки программы должна быть выполнена работа по программированию (кодированию) и отладке программы.   
На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями к составу документации.   
На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:   
1. разработка, согласование и утверждение и методики испытаний;   
2. проведение приемо-сдаточных испытаний;   
3. корректировка программы и программной документации по результатам испытаний.   
На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах Заказчика.

**7. Порядок контроля и приемки**

**7.1. Виды испытаний**

Приемо-сдаточные испытания должны проводиться на объекте Заказчика в оговоренные сроки.   
Приемо-сдаточные испытания программы должны проводиться согласно разработанной Исполнителем и согласованной Заказчиком Программы и методик испытаний.   
Ход проведения приемо-сдаточных испытаний Заказчик и Исполнитель документируют в Протоколе проведения испытаний

**7.2. Общие требования к приемке работы**

На основании Протокола проведения испытаний Исполнитель совместно с Заказчиком подписывает Акт приемки-сдачи программы в эксплуатацию

# **Приложение Б**

Фрагменты исходного программного кода информационной системы «Спортивная школа»

Окно аутентификации:

namespace SportSchool

{

public partial class Authentication : Window

{

SportSchoolDataContext \_context = new();

public Authentication()

{

InitializeComponent();

}

private void EnterButton\_Click(object sender, RoutedEventArgs e)

{

foreach (var item in \_context.Employees.ToList())

{

if (item.Login.Equals(loginTextBox.Text)

&&

item.Password.Equals(passwordTextBox.Text))

{

MainWindow mainWindow = new(\_context, item.Id);

mainWindow.Show();

this.Close();

return;

}

}

MessageBox.Show("Логин и/или пароль некорректен",

"Ошибка",

MessageBoxButton.OK,

MessageBoxImage.Warning);

}

private void ExitButton\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

}

}

Окно добавления сотрудника:

public partial class AddCourse : Window

{

SportSchoolDataContext \_context;

List<TextBox> \_textBoxes = new();

public AddCourse(SportSchoolDataContext context)

{

InitializeComponent();

\_context = context;

\_textBoxes.Add(nameTextBox);

\_textBoxes.Add(durationTextBox);

\_textBoxes.Add(descriptionTextBox);

}

private void CloseButton\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

private void AddButton\_Click(object sender, RoutedEventArgs e)

{

if (FieldsValidator.IsValid(\_textBoxes))

{

Course course = new() { Name = nameTextBox.Text,

Description = descriptionTextBox.Text,

Duration = durationTextBox.Text };

\_context.Courses.Add(course);

\_context.SaveChanges();

MessageBox.Show("Данные успешно внесенны", "Успех",

MessageBoxButton.OK,

MessageBoxImage.Information);

this.Close();

}

else

{

MessageBox.Show("Заполнены не все поля", "Ошибка",

MessageBoxButton.OK,

MessageBoxImage.Error);

}

}

}

**ПРИЛОЖЕНИЕ В**

Руководство пользователя

Назначение программы

Программа предназначена для:

* Использования сотрудниками спортивной школы;
* Хранения информации о сотрудниках, документах, клиентах и курсах;
* Удобного получения необходимых сведений.

Версия 0.1 Производство: ООО "S. corp": 355000, г. Ставрополь, ул. Черняховского, д. 3. Программа выполнена по индивидуальному заказу ООО "Колледж Связи". Лицензия: Программа может использоваться исключительно авторизованным правообладателем.

Условия выполнения программы

Минимальные системные требования:

ОС: Windows 8/ 8.1/10/11 x32/x64

Процессор: Intel core2 Duo E5200

Оперативная память: 256 МБ ОЗУ

Место на диске: 100 МБ

Сеть: Широкополосное подключение к интернету

Рекомендуемые системные требования:

ОС: Windows 8/ 8.1/10/11 x32/x64

Процессор: Intel core2 Duo E7300

Оперативная память: 512 МБ ОЗУ

Место на диске: 100 МБ

Сеть: Широкополосное подключение к интернету

Начало работы

Для запуска информационной системы необходимо открыть исполнительный файл, находящийся на персональном компьютере сотрудника.

При запуске информационной системы пользователя встречает меню аутентификации. Пользователь не может сам зарегистрироваться в системе и должен ввести данные, которые были предоставлены ему уполномоченным лицом.

Если у пользователя нет доступ к информационной системе, то он должен обратиться к уполномоченному лицу для внесения информации о себе в базу данных информационной системы и получения доступа к её функционалу.

Основные функции

Пользователь получает доступ к различным таблицам, хранящим информацию о документах, пользователях, курсах, основываясь на ролях, которые он имеет.

Пользователь получает возможность выполнять следующие действия с этими записями:

* Просматривать
* Добавлять
* Удалять

Дополнительные возможности

Данной информационной системой не предусмотрены никакие дополнительные возможности.