**Введение**

Курсовая работа направлена на проектирование и создание многопользовательского приложения. В процессе выполнения курсовой работы студент отрабатывает технологический процесс разработки программного продукта, включающий в себя разработку технического задания, внешнее и внутреннее проектирование, кодирование, тестирование, разработку документации и развитие способности принимать обоснованные решения в ходе проектной деятельности и получать базовый опыт разработки программного обеспечения.

Для выполнения курсового проекта была выбрана тема «Учёт выполненных курсовых работ». Цель проекта заключается в разработке программного обеспечения, которое поможет вести учет оценок за курсовые работы студентов, выявлять неуспевающих и составлять отчеты по выбранным параметрам.

Программный продукт - это программа, которая может использоваться независимо от ее разработчиков для заданных целей на разных компьютерах, при условии, что они соответствуют системным требованиям.

1. **Выработка системных требований**

**Описание предметной области**

При сдаче студентом курсовой работы руководителю и выставлении оценки, руководитель добавляет в «Список выполненных курсовых работ», в котором описывается вся информация, необходимая для анализа и хранения (табельный номер студента, ФИО студента, группа, предмет, предметная область, тема, оценка и год). Этот документ является архивом информации о всех сданных курсовых работах студентами. В конце учебного года руководитель готовит отчет по сданным курсовым работам в группе. Так же есть возможность выводить список работ по определённому предмету и году. И работ по руководителям.  
 Благодаря, новой системе преподавателям не нужно будет идти в деканат и нести список, ведь они могу после сдачи курсовой работы студентом сразу же, через новое ПО проставить оценку студенту, в свою очередь благодаря БД, преподавателю даже не придётся заполнять данные о ученике.

Автоматизация задачи позволит повысить производительность работы руководителей курсовых работ, уменьшить количество ручной работы при оформлении отчетов, повысить точность и достоверность данных, уменьшить количество ошибок при подготовке документов, что будет способствовать повышению надежности и достоверности предоставляемой информации. Внедрение программы позволит сократить денежные, трудовые и временные затраты в организации процесса учета курсовых и поиска необходимой информации. Хранение информации о курсовых в базе данных и поиск по различным критериям существенно упростится. Удобный интерфейс позволит руководителю не только оперативно ввести информацию, но и узнать всю необходимую для него информацию.

**1.2 Цель проекта**

Обеспечить точность и своевременность учета выполненных работ, чтобы руководитель мог контролировать прогресс сдачи курсовых работ, управлять бюджетом и принимать информированные решения на основе актуальной информации.

**1.3 Описание проблемы**

Основной целью автоматизации данной предметной области является необходимость более удобной и быстрой обработки данных по сделанным курсовым работам. Для этого необходимо создать базу данных для хранения информации о студентах, предметах и курсовых работах на электронном носителе.

В университете учет курсовых работ, сделанных студентами, ведется посредством электронных таблиц Microsoft Excel и текстового редактора Microsoft Word. Отсутствие единой базы данных привело к усложнению учета и получения необходимой информации для преподавателей и директора. В связи с этой проблемой возникла актуальная необходимость осуществления автоматизации учета сделанных и защищенных курсовых работ студентами университета.

* 1. **Словарь предметной области**

Таблица 1 – Словарь терминов предметной области

|  |  |  |
| --- | --- | --- |
| № | Термин | Назначение |
| 1. | Студент | Человек, который обучается в университете |
| 2. | Сотрудник деканата | Сотрудник, который работает в деканате и ведёт работу с ведомостями |
| 3. | Ведомость | Документ, в котором выставляются оценки за зачет или экзамен всем студентам группы |
| 4. | Преподаватель | Работник университета, преподающий определенную дисциплину |
| 5. | Дисциплина | Учебный предмет в университете |
| 6. | Параметр оценки | Отметка, показывающая качество освоения дисциплины учащимся, значения: «2», «3», «4», «5», «нз» - не зачёт |

Таблица 2 – Словарь предметной области по методу Аббота

|  |  |  |
| --- | --- | --- |
| № | Существительное | Глагол |
| 1. | Студент | - |
| 2. | Сотрудник деканата | - |
| 3. | Ведомость | - |
| 4. | Преподаватель | - |
| 5. | Дисциплина | - |
| 6. | Параметр оценки | Распознать и вывести |

Таблица 3 – Объектно-ориентированный словарь

|  |  |  |  |
| --- | --- | --- | --- |
| № | Класс (Сущность/Актер) | Свойство (Состояние) | Метод (Функция/Поведение) |
| 1. | Студент | ФИО, группа | - |
| 2. | Сотрудник деканата | ФИО | - |
| 3. | Ведомость | Преподаватель, дисциплина, параметр оценки, список ФИО студентов, подпись преподавателя | - |
| 4. | Преподаватель | Фио | - |
| 5. | Дисциплина | Наименование, ЗЕТ, количество часов, критерий типа контроля | - |
| 6. | Параметр оценки | «2», «3», «4», «5», «нз» - не зачёт, | Запись, редактирование |

**1.5 Функциональность ПО**

**Формирование отобржения информации о выполненных курсовых работах**

Модуль должен, вывести на экран информацию о предметах, студентах, темой и оценкой за курсовую работу. Так же должен предоставлять пользователю возможность, двигать ползунок вверх и вниз, тем самым давать всю возможную информацию.

**Формирование отчётных документов**

Модуль должен давать возможность, формировать отчётные документы, для руководства, деканата и т.д. В данных документах будет выполенная курсовая работа с оценкой, так же эти документы можно будет распечатать или переслать почтой или прочими мессенджерами.

**Изменения итоговой оценки для задолжников**

Модуль должен давать возможность пользователю изменять информацию о выбранном предмете.

**Фильтрация списка предметов**

Модуль должен давать возможность пользователю отфильтровать список предметов по заданным критериям: по студентам, руководству, специальности и группе.

При фильтрации по студенту, необходимо ввести ФИО студента.

При фильтрации по специальности, необходимо узнать о существовании данной специальности в учебном заведении.

При фильтрации по группе, необходимо указать данную группу.

При фильтрации по руководству, необходимо указать год сдачи или тему или оценку.

**2. Формальное описание требований к программному продукту**

**2.1 Определение бизнес требований**

В данном разделе описан бизнес-процесс контроль успеваемости студентов.

Описание контекста модели:

Предмет моделирования – процесс автоматизации учёта выставления оценок за курсовую работу у студентов.

Область моделирования – модель охватывает процессы, связанные с записью информации об оценках за курсовую работу студентов в базу данных.

Цель моделирования – детальное представление работы системы

Точка зрения – Стдуент

Диаграмма (А-0) является вершиной структуры и представляет собой общее описание системы и её взаимодействия с внешней средой (Рис. 1).

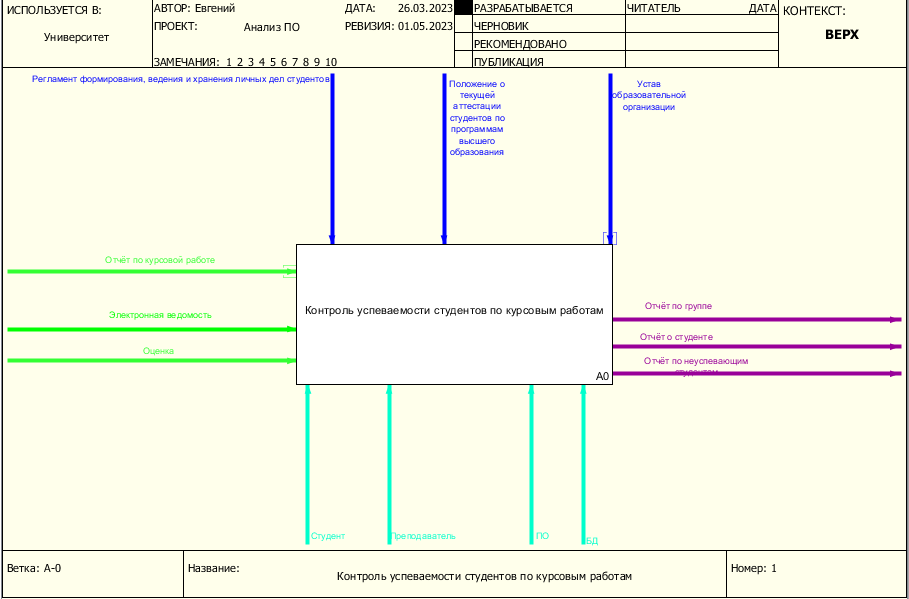


Рисунок 1 – Процесс «Контроль успеваемости студентов по курсовым работам»

Процесс «Контроль успеваемости по курсовым работам» (см. рисунок 2):

Основные бизнес-функции процесса:

1. Загрузка электронной ведомости в программу
2. Контроль успеваемости студентов
3. Запись данных в БД
4. Отчёты

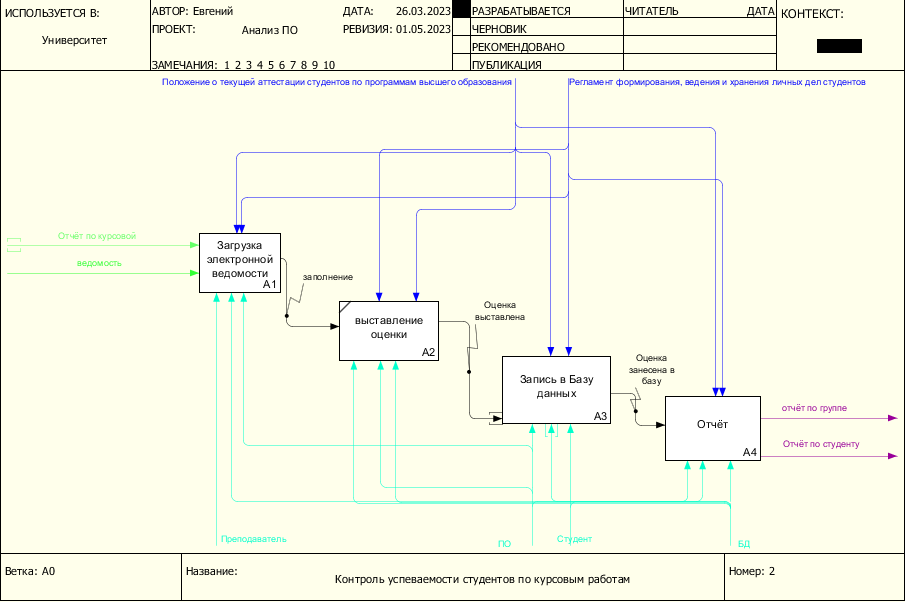


Рисунок 2 – Декомпозиция обобщенного процесса «Контроль успеваемости студентов»

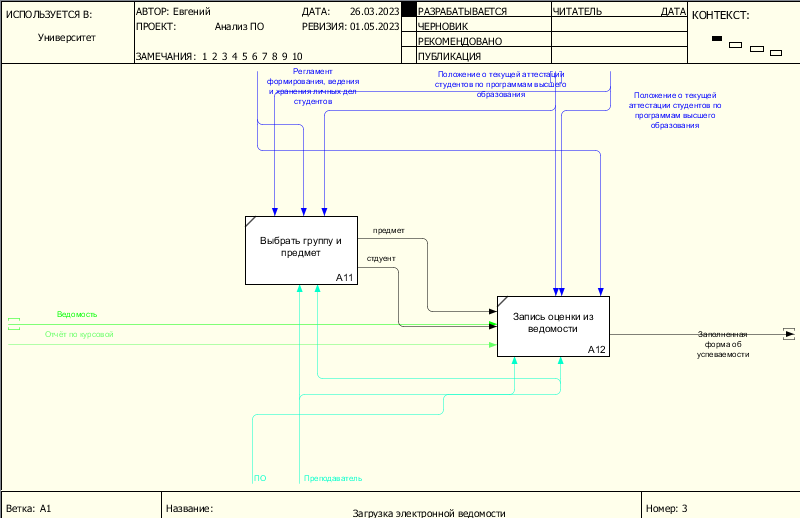


Рисунок 3 – Детализация бизнес-функции процесса «Загрузка электронной ведомости в программу»

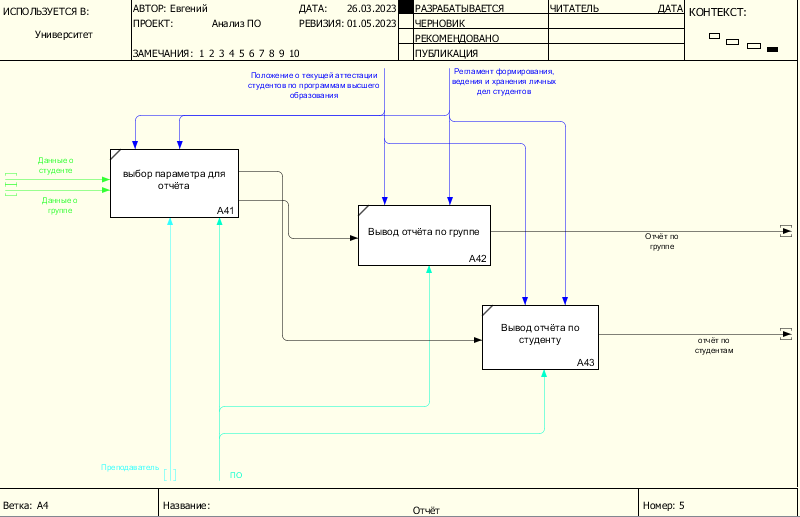
****

Рисунок 4 – Детализация бизнес-функции процесса «Отчёты»

**2.2 Определение требований пользователя**

Опишем функциональное назначение системы с помощью диаграммы вариантов использования, таким образом, показав, что будет делать разрабатываемся система.

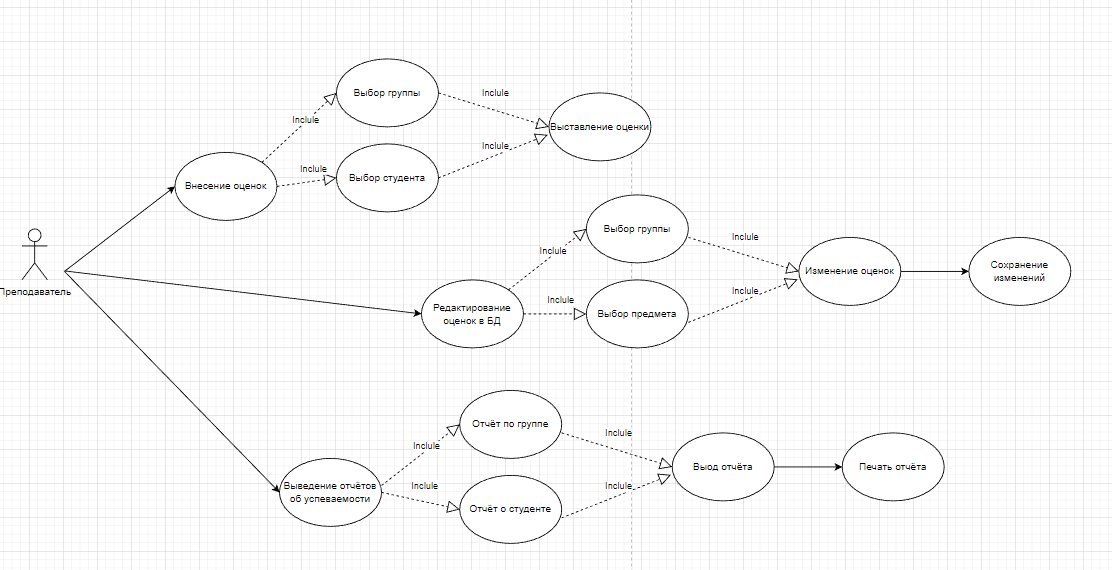


Рисунок 5 – Диаграмма функциональности разрабатываемого приложения

**Спецификации варианта использования метода «Загрузка списка студентов»**

**Цель:** Загрузить список студентов

**Активные субъекты:** Преподаватель.

**Предусловие:** Когда программа запускается, открывается главная форма, на которой находятся пункты меню, такие как "Внесение оценок", "Редактирование оценок в БД" и "Выведение отчётов об успеваемости". При нажатии на кнопку "Внесение оценок" открывается вторая форма, названная "Внесение оценок", на которой можно выбрать группу и предмет, а затем занести оценки в базу данных. Список доступных групп для выбора подгружается при переходе с главной формы на форму "Внесение оценок" через кнопку "Внесение оценок".

**Основной сценарий:**

1. Нажатие кнопки внесения оценок
2. Запрос списка студентов
3. Получение данных
4. Отображение списка студентов

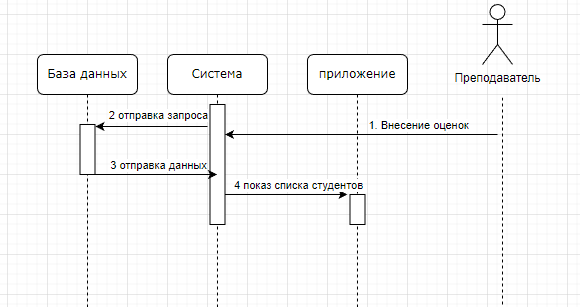


Рисунок 6 – Описание функции «Загрузка списка студентов»

**Спецификации варианта использования метода «Выбор группы»**

**Цель:** Выбрать группу ивывести соответствующий список студентов и список предметов для выбранной группы.

**Активные субъекты:** Преподаватель

**Предусловие:** Перед выбором группы уже загружен список групп.

**Основной сценарий:**

1. Выбор нужной группы из списка
2. Отправка запроса
3. Запрос по предметам
4. Отправка данных
5. Получение данных предметов
6. Отображение списка студентов группы
7. Отображение предметов

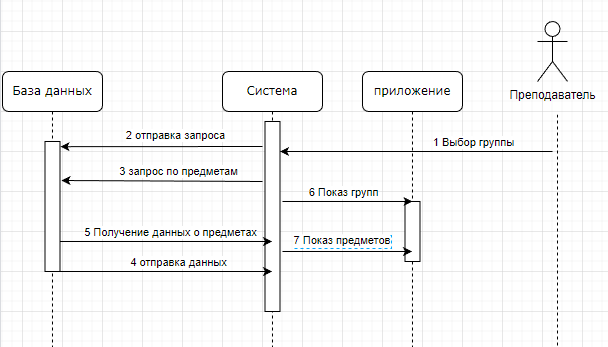


Рисунок 7 – Описание функции «Выбор группы»

**Спецификации варианта использования метода «Выбор предмета»**

**Цель:** Выбрать предмет.

**Активные субъекты:** Преподаватель.

**Предусловие:** Перед этим этапом производится загрузка списка групп, студентов и предметов. Однако список предметов становится доступным только после выбора конкретной группы.

**Основной сценарий:**

1. Выбор нужного предмета
2. Отработка запроса
3. Отображение предмета

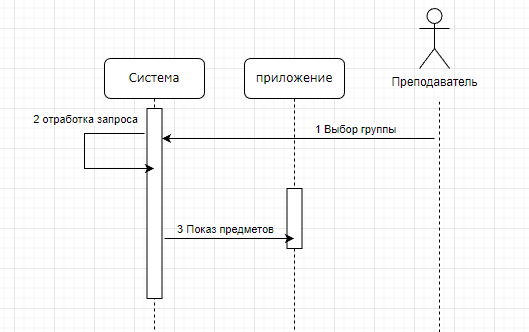


Рисунок 8 – Описание функции «Выбор предмета»

**Спецификации варианта использования метода «Загрузка файла ведомости»**

**Цель:** Открыть файл ведомости

**Активные субъекты:** Преподаватель

**Предусловие:** Перед этим выбрана группа и предмет.

**Основной сценарий:**

1. Загрузка файла
2. Загрузка
3. Отображение оценок студентов

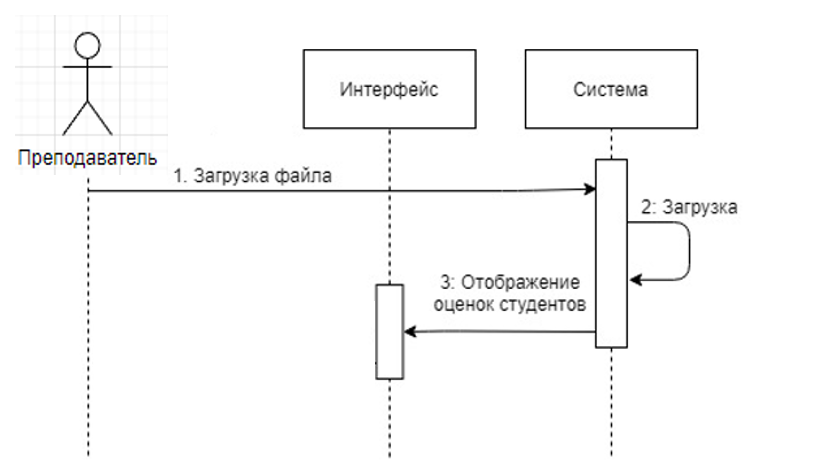


Рисунок 9 – Описание функции «Загрузка файла ведомости»

Для разрабатываемой системы представлена диаграмма сущность-связь по Чену на рисунке (10). На диаграмме активные части системы обозначены прямоугольниками, а атрибуты - овалами.

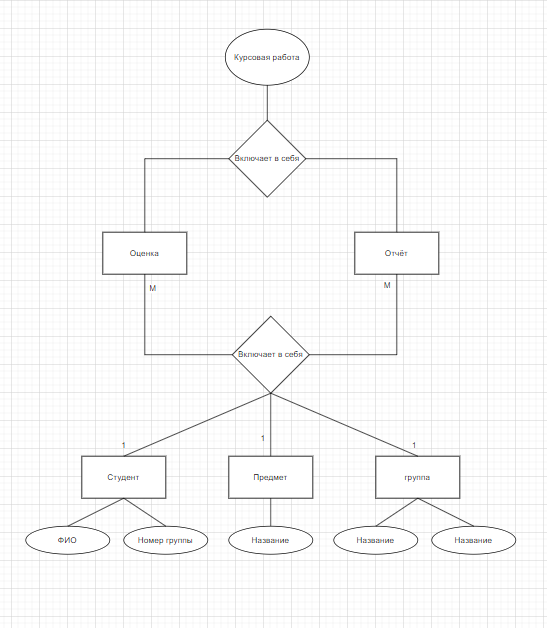


Рисунок 10 – Диаграмма сущностей-связей по Чену

**2.3 Описание поведения разрабатываемой системы**

Составим диаграмму активностей, которая будет описывать процесс редактирования оценок.

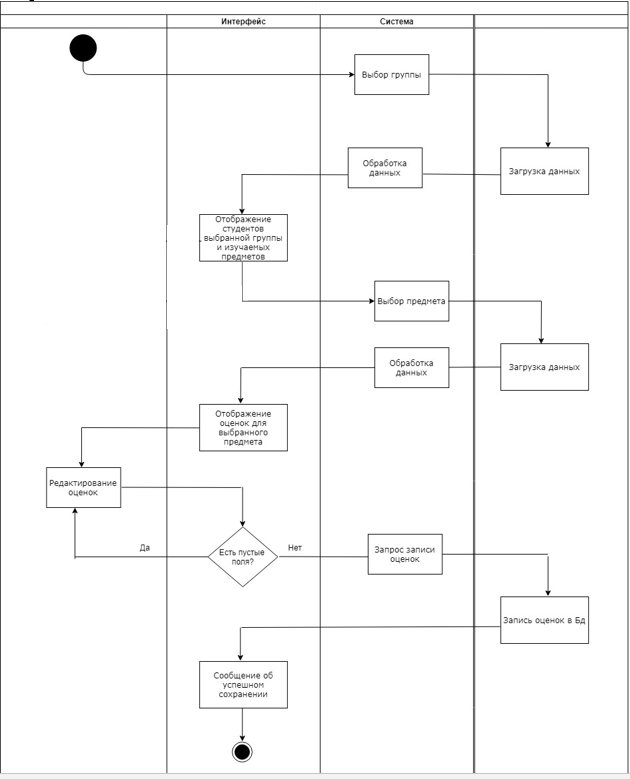


Рисунок 11 – Диаграмма активностей процессов «Редактирование оценок в БД»

**3. Техническое задание**

1. ВВЕДЕНИЕ

Это техническое задание описывает создание программного продукта, который будет использоваться для автоматизации процесса записи информации о студенческой успеваемости в электронную базу данных из текстового формата ведомости. В настоящее время этот процесс не автоматизирован, и информация о успеваемости хранится либо в электронных таблицах MS Excel, либо на бумажных носителях, что затрудняет быструю обработку данных и уведомление студентов о их успеваемости.

Для разработки программного продукта требуется использовать стандартизованные и легко обслуживаемые решения, которые будут реализованы в виде открытой системы с возможностью добавления новых функций.

2. ОБЩИЕ СВЕДЕНИЯ

**2.1 Наименования и шифры**

2.1.1 Название системы

Название системы - "Система автоматизации контроля учёта выполненных курсовых работ".

2.1.2 Код системы

Код системы - ISA SPC.

**2.2** **Информация о заказчиках и исполнителях**

2.2.1 Заказчик: Иркутский национальный исследовательский технический университет (ИрНИТУ)

Адрес: 664033, г. Иркутск, ул. Лермонтова, 83.

2.2.2 Исполнитель

Исполнителем является студентка Завьялова Радиана Аркадьевна.

**2.3** **Основание для разработки**

2.3.1 Основа для разработки автоматизированной системы

Основанием для разработки автоматизированной системы, изложенной в настоящем техническом задании, является учебный план Института информационных технологий и анализа данных.

2.3.2 Основание для разработки документа

Основанием для разработки данного документа является раздел 6 договора № 02-09.

2.3.3 Нормативные документы

Настоящее Техническое задание разработано в соответствии с требованиями ГОСТ 34.602-89 "Информационная технология. Комплексные стандарты на автоматизированные системы. Техническое задание на создание автоматизированных систем".

При разработке и вводе в эксплуатацию автоматизированной системы исполнитель должен соблюдать требования следующих государственных стандартов:

ГОСТ 34.601-90. Информационные технологии. Комплексные стандарты на автоматизированные системы. Автоматизированные системы. Этапы создания;

ГОСТ 34.603-92. Информационные технологии. Виды испытаний автоматизированных систем;

ГОСТ 19. Единая система программной документации;

РД 50-34.698-90. Методические указания. Информационные технологии. Комплексные стандарты на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

**2.4** **Сроки выполнения работы**

Разработка началась 2 марта 2021 года.

Разработка будет завершена 9 октября 2021 года.

**2.5 Информация об источниках финансирования и процедурах**

Финансирование разработки информационной системы не предусмотрено, так как она разрабатывается в рамках курсовой работы.

**2.6** **Процедуры представления документов и предоставления их заказчику**

Процедура документирования и представления работы заказчику осуществляется в соответствии с договором. Приемка системы осуществляется комиссией, создаваемой заказчиком.

3. НАЗНАЧЕНИЯ И ЦЕЛИ СОЗДАНИЯ

**3.1. Назначение**

Информационная система "Система автоматизации контроля учёта выполненных курсовых работ" используется для автоматизации процесса записи информации из текстового формата ведомости в электронную базу данных.

**3.2 Цели создания**

3.2.1 Общие цели проекта

Цели создания и внедрения "Система автоматизации контроля учёта выполненных курсовых работ" – сокращение временных и трудовых ресурсов для фиксации успеваемости студентов.

3.2.2 Цели текущего этапа

Целью работ по данному этапу проекта является создание первой экспериментальной версии информационной системы.

4. ХАРАКТЕРИСТИКА ОБЪЕТОВ АВТОМАТИЗАЦИИ

**4.1 Краткие сведения об объекте автоматизации**

Объектом автоматизации являются процесс контроля успеваемости студентов.

**4.2 Сведения об условиях эксплуатации**

Автоматизированная система будет эксплуатироваться в составе программного комплекса Заказчика.

5. ТРЕБОВАНИЯ К ПРОГРАММЕ

**5.1 Требования к функциональным характеристикам**

5.1.1 Программа должна обеспечивать возможность выполнения следующих функций:

Ввод оценок учащихся из классных журналов в базу данных путем выбора необходимой группы, предмета и загрузки соответствующего файла классного журнала в программу;

редактировать оценки в базе данных, выбрав необходимую группу и предмет;

Выводить отчеты по успеваемости студентов: отчет по группе, отчет по конкретному студенту, отчет по неуспевающим студентам.

5.2 Требования к надежности

Обеспечить блокировку некорректных действий пользователя при работе с системой, в частности, ввод данных не того типа, заполнение не всех обязательных полей.

**5.3 Требования к эргономике и технической эстетике**

Система должна иметь графический интерфейс. Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами, должен обеспечивать быстрое отображение экранных форм. Элементы навигации должны быть удобными для пользователя.

**5.4 Требования к составу и параметрам технических средств**

Система должна работать на IBM-совместимых персональных компьютерах. Системные требования представлены в таблице 2.2.

Таблица 1 - Системные требования к программному продукту

|  |  |
| --- | --- |
| Компонент | Минимальные требования |
| Процессор | 1,8 гигагерц (ГГц) |
| ОЗУ | 1 гигабайт (ГБ) |
| ВЗУ | Раздел с файловой системой NTFS и 2 ГБ свободного места |
| Экран | Монитор с разрешением 1024х768 или выше |

**5.5 Требования к информационной и программной совместимости**

Система должна работать под управлением семейства операционных систем Window (Windows 7, Windows 8, Windows 10, Windows 11).

**5.6 Требования к квалификации персонала:**

1. Пользователи программного продукта должны иметь базовые навыки работы с операционными системами Microsoft (любая из версий: Microsoft Windows XP,7,8,10).
2. Техническое обслуживание и администрирование оборудования должно выполняться специалистами, имеющими соответствующую квалификацию и навыки выполнения работ.

6. СОСТАВ И СОДЕРЖАНИЕ РАБОТ ПО СОЗДАНИЮ СИСТЕМЫ

Таблица 2 – Содержание работ по созданию системы

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название этапа | Срок | Отчетность |
| 1 | Выработка системных требований (постановка задачи) | 20.04.23 -01.05.23 | Постановка задачи |
| 2 | Формальное описание требований к программному продукту | 20.04.23 -01.05.23 | Техническое задание |
| 3 | Архитектурное и интерфейсное проектирование программного продукта | 20.04.23 -01.05.23 | Проектирование ПП |
| 4 | Анализ и управление рисками | 20.04.23 -01.05.23 | Пояснительная записка к КР |
| 5 | Программирование | 20.04.23 -01.05.23 | Программный код |
| 6 | Тестирование (автоматизированное) | 20.04.23 -01.05.23 | Код, пояснительная записка к КР |
| 7 | Разработка документации | 20.04.23 -01.05.23 | Документация |
| 8 | Планирование и управление процессом разработки | Планирование:  20.04.23 -01.05.23  Управление процессом разработки:  В течение всего проекта | План |

По окончанию каждого этапа заказчику будет предоставлен отчёт о выполненной работе.

7. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

**7.1 Этапы сдачи-приемки работ осуществляются согласно установленному Календарному графику.**

Подтверждением завершения работы по каждому этапу является Отчет о завершении работ.

**7.2 Готовые подсистемы "Система автоматизации контроля учёта выполненных курсовых работ", созданные в рамках проекта, проходят предварительные испытания для проверки работоспособности и соответствия требованиям ТЗ.**

Предварительные испытания проводятся представителями Заказчика на основе программы тестирования, разработанной Исполнителем и согласованной с Заказчиком. Если необходимо, Исполнитель корректирует проектную и эксплуатационную документацию и устраняет выявленные недостатки программного обеспечения.

**7.3 Передача системы в опытную эксплуатацию**

Осуществляется при наличии положительного заключения, вынесенного по результатам предварительных испытаний, и производится приемо-сдаточной комиссией. Опытная эксплуатация должна проводиться на технических средствах Заказчика.

8. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

Разрабатываемые программные модули должны быть самодокументированы, т.е. тексты программ должны содержать все необходимые комментарии.

В состав сопровождающей документации должны входить:

* Пояснительная записка, содержащая описание разработ­ки.
* Руководство пользователя ИС.

**4. Проектирование программного продукта**



**Архитектура ПО**

* + 1. **Проектирование классов разрабатываемой системы**

Для создания программы будет использоваться язык программирования C# в среде разработки Visual Studio, которая является многофункциональной программой для написания, отладки и сборки кода, а также публикации приложений. Visual Studio включает в себя не только стандартный редактор и отладчик, но и компиляторы, средства автозавершения кода, графические конструкторы и многие другие функции для упрощения процесса разработки.

Для удобства взаимодействия с базой данных, которая будет создана в MS SQL Server, будет использоваться язык программирования C#. SQL Server предоставляет широкие возможности для хранения и доступа к данным, включая поддержку языка XML и протокола HTTP, а также средства повышения быстродействия и доступности

Для связи программы с базой данных будет использован ADO.NET, набор классов, предоставляющих доступ к данным в реляционном формате, включая реляционные базы данных, такие как Microsoft SQL Server. Архитектура разрабатываемой системы определяется набором функций, которые осуществляют преобразования при выполнении системой своего назначения.

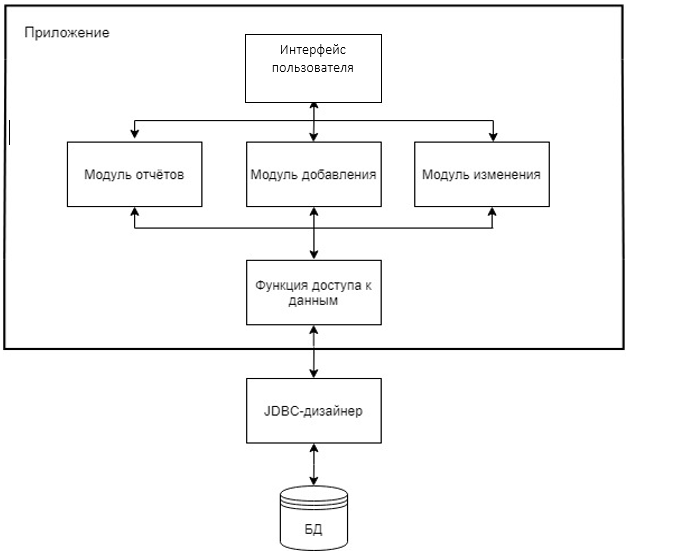


Рисунок 12 – Архитектура ИС

По изображению можно увидеть, что система, над которой работается, будет построена на двух уровнях: приложение и база данных будут иметь свои собственные подуровни. Для соединения между приложением и базой данных будет использоваться JDBC-драйвер. Данные будут синхронизироваться между приложением и базой данных через модуль "функции доступа к данным".

Модуль отчетов будет ответственным за создание отчетов о успеваемости.

Модуль добавления будет отвечать за добавление новых оценок в базу данных.

Модуль изменения будет использоваться для редактирования уже существующих оценок.