**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).

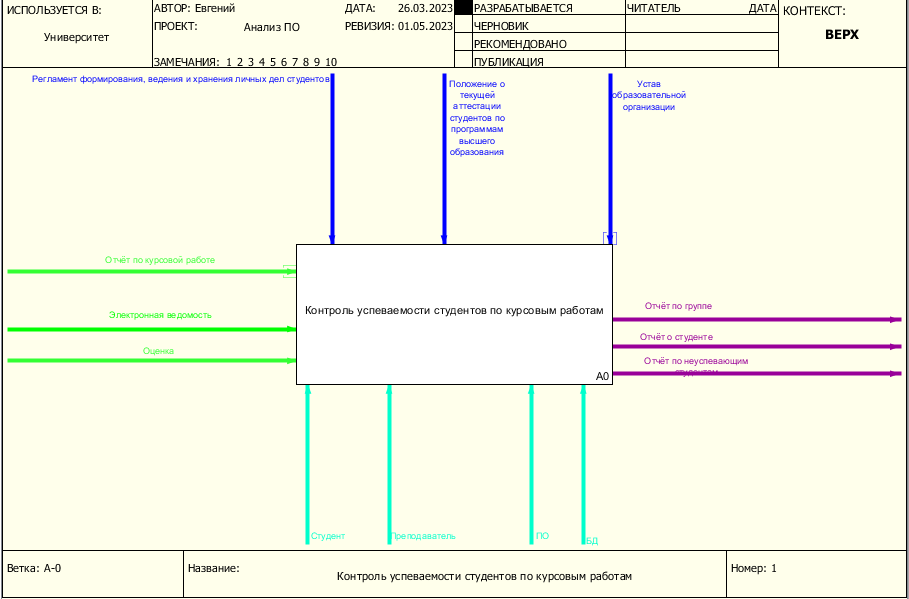


Рисунок 1 – Процесс «Контроль успеваемости студентов по курсовым работам»

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

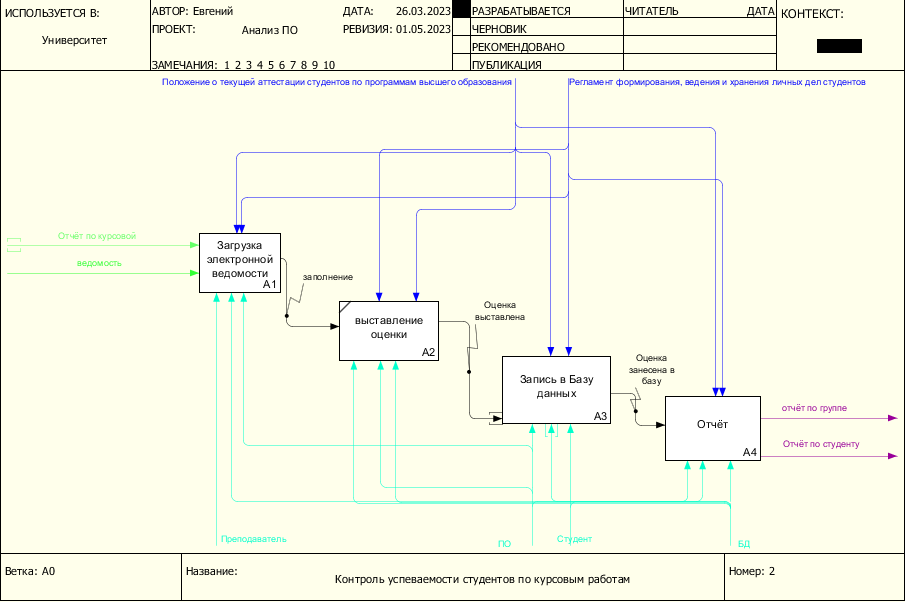


Рисунок 2 – Декомпозиция обобщенного процесса «Контроль успеваемости студентов»

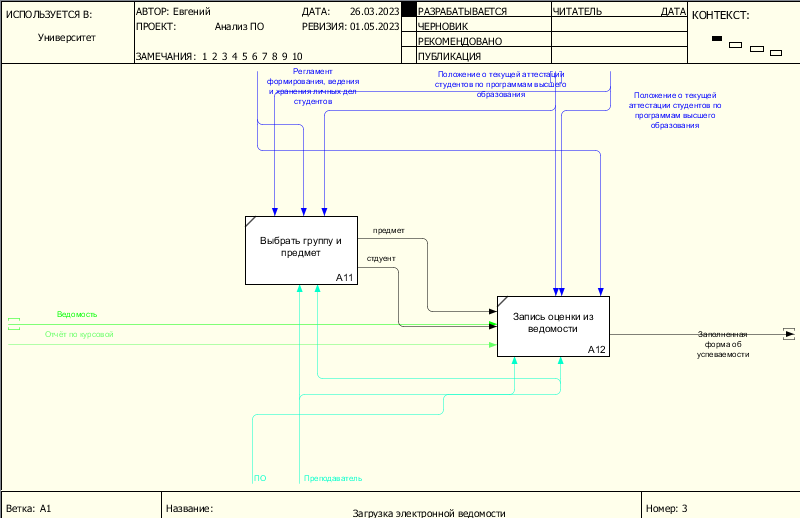


Рисунок 3 – Детализация бизнес-функции процесса «Загрузка электронной ведомости в программу»

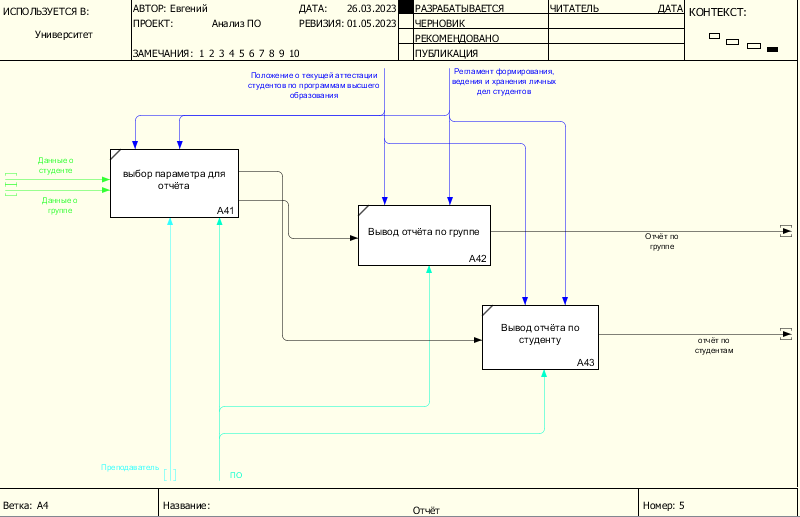
****

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.

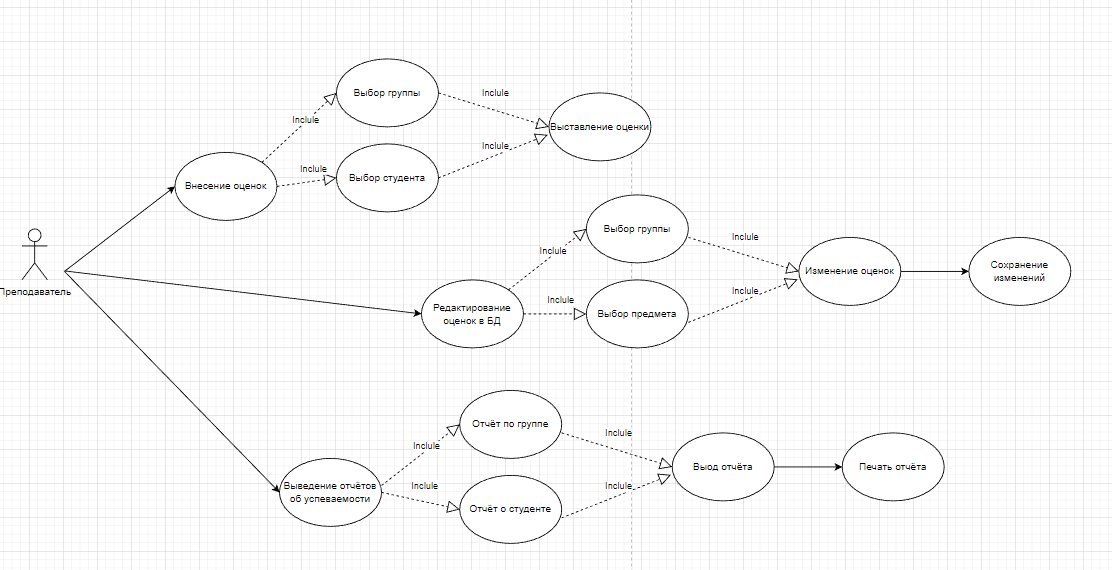


Рисунок 5 – Диаграмма функциональности разрабатываемого приложения

**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

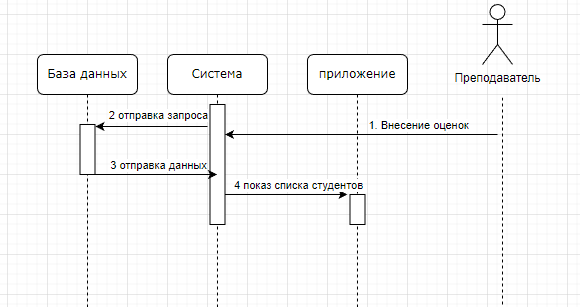


Рисунок 6 – Описание функции «Загрузка списка студентов»

**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

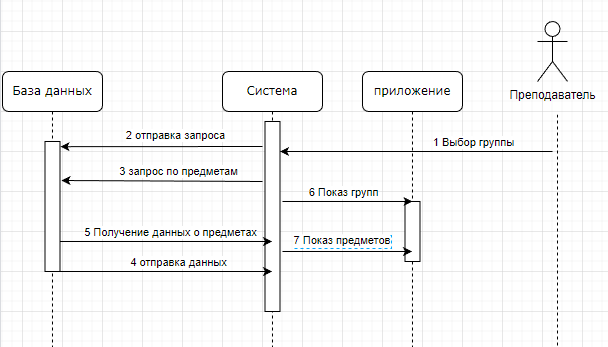


Рисунок 7 – Описание функции «Выбор группы»

**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

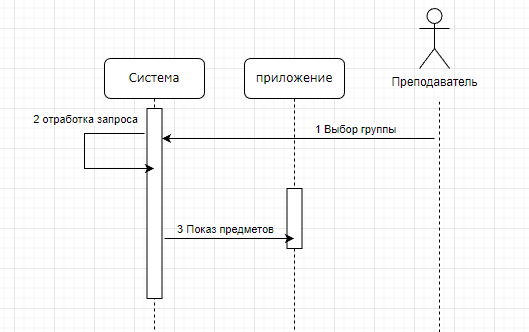


Рисунок 8 – Описание функции «Выбор предмета»

**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

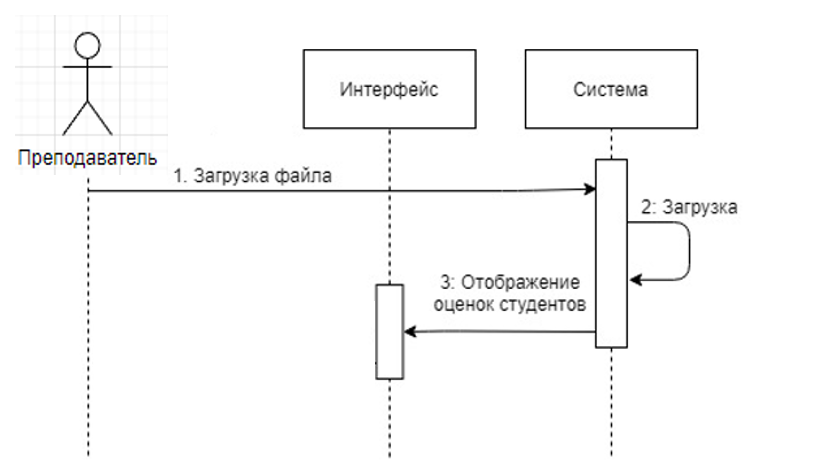


Рисунок 9 – Описание функции «Загрузка файла ведомости»

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

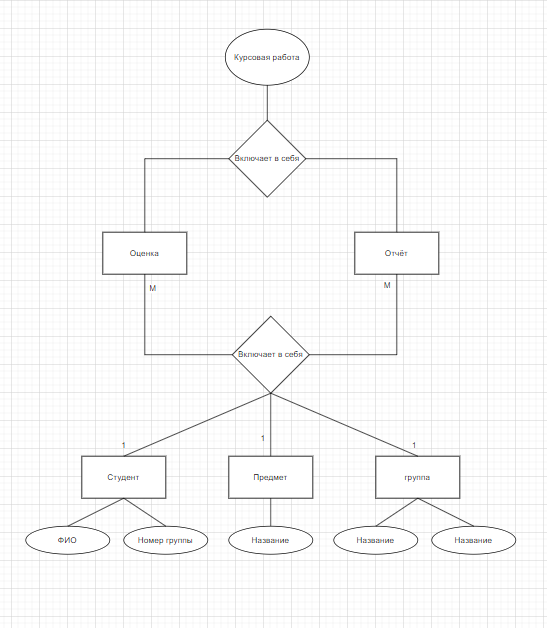


Рисунок 10 – Диаграмма сущностей-связей по Чену