Казанский (Приволжский) федеральный университет

Институт вычислительной математики и информационных технологий

Прикладная математика

Техническое задание к проекту

“Электронный дневник студента”

Работу выполнил:

Раков Р.О.

Команда №6

Содержание.

[1. Введение. 3](#_heading=h.vpdo8zy1gunw)

[1.1. Цель. 3](#_heading=h.1fob9te)

[1.2. Краткая сводка возможностей. 3](#_heading=h.3znysh7)

[1.3. Определения, акронимы и сокращения. 3](#_heading=h.2et92p0)

[1.4. Краткое содержание. 4](#_heading=h.tyjcwt)

[2. Обзор системы. 5](#_heading=h.3dy6vkm)

[2.1. Определение типов пользователей системы. 5](#_heading=h.1t3h5sf)

[2.2. Окружение системы. 5](#_heading=h.4d34og8)

[2.3. Обзор вариантов использований. 5](#_heading=h.17dp8vu)

[2.4. Нефункциональные требования. 7](#_heading=h.3rdcrjn)

[3. Детальные требования. 8](#_heading=h.26in1rg)

[3.1. Функциональные требования. 8](#_heading=h.lnxbz9)

[3.2. Требования к пользовательским интерфейсам. 12](#_heading=h.35nkun2)

[3.3. Требования к логической структуре БД. 18](#_heading=h.1ksv4uv)

# 

# 

# 1. Введение.

## 1.1. Цель.

Цель документа - представить детальное описание создаваемого приложения. Он описывает цели и возможности системы, пользователей системы, сценарии использования (пользовательские истории) и строение системы.

## 1.2. Краткая сводка возможностей.

Программа "Электронный дневник студента" предназначена для отображения информации об учебном процессе в университете. Предусмотрены функционал для студента и администратора.

## 1.3. Определения, акронимы и сокращения.

БД(База данных) — совокупность данных, хранимых в соответствии со схемойданных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных.

C# — объектно-ориентированный язык программирования общего назначения.

IDE (*Integrated Development Environment* — «Интегрированная среда разработки») — система программных средств, используемая программистами для разработки программного обеспечения.

SQL (*Structured Query Language* — «язык структурированных запросов») — декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных.

MySQL — свободная реляционная система управления базами данных

## 1.4. Краткое содержание.

Документ содержит информацию о типах пользователей, окружении системы, вариантах использования, нефункциональных требованиях, функциональных требованиях, требованиях к пользовательским интерфейсам и требованиях к логической структуре БД.

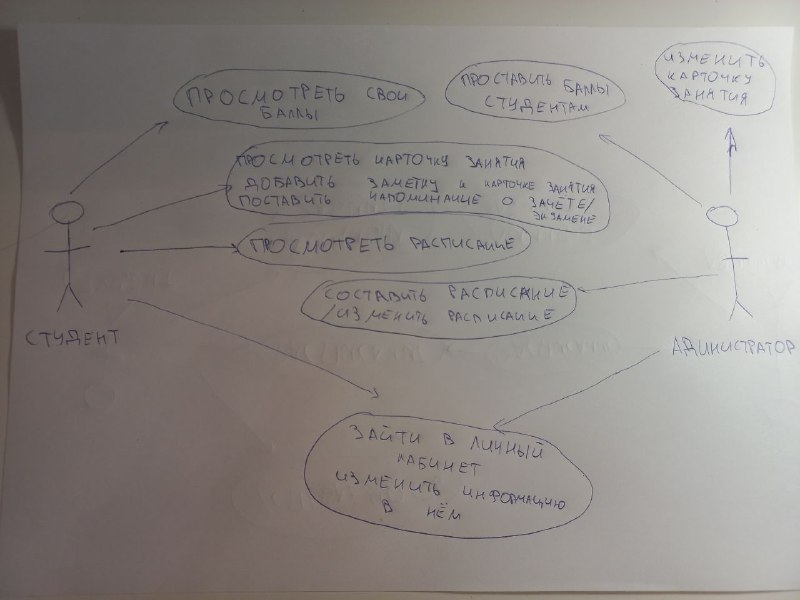
# 2. Обзор системы.

## 2.1. Определение типов пользователей системы.

- Администратор

- Студент

## 2.2. Окружение системы.



## 2.3. Обзор вариантов использований.

User Story 1: Как пользователь, я хочу иметь возможность войти в электронный дневник, чтобы выполнять дальнейшие действия.

User Story 2: Как администратор, я хочу иметь возможность создавать и редактировать учебные группы и их состав, чтобы управлять данными.

User Story 3: Как администратор, я хочу иметь возможность составлять и редактировать расписание учебного процесса, чтобы обеспечить организованное проведение занятий.

User Story 4: Как администратор, я хочу иметь возможность проставлять баллы за лабораторные работы, зачёты и экзамены, чтобы вести учет успеваемости студентов.

User Story 5: Как администратор, я хочу иметь возможность просматривать и редактировать информацию в своем профиле, чтобы поддерживать актуальные данные.

User Story 6: Как студент, я хочу иметь возможность просматривать свое расписание в удобном формате, чтобы быть в курсе занятий.

User Story 7: Как студент, я хочу иметь возможность просматривать карточку занятия, чтобы узнать больше информации о нем.

User Story 8: Как студент, я хочу иметь возможность просматривать свои баллы, чтобы быть в курсе своей успеваемости.

User Story 9: Как студент, я хочу иметь возможность настраивать уведомления на почту о приближающихся зачётах/экзаменах, чтобы быть информированным о предстоящих событиях.

User Story 10: Как студент, я хочу иметь возможность вносить заметки к карточкам занятий, чтобы сохранять дополнительную информацию.

User Story 11: Как студент, я хочу иметь возможность просматривать и редактировать информацию в своем профиле, чтобы поддерживать актуальные данные.

## 2.4. Нефункциональные требования.

Для создания системы будет использоваться IDE Visual Studio.Программа будет написана на языке программирования C#. База данных будет написана на языке программирования SQL.Для создания схемы базы данных будет использоваться MySQL.

# 3. Детальные требования.

## 3.1. Функциональные требования.

User Story 1: Как пользователь, я хочу иметь возможность войти в электронный дневник, чтобы выполнять дальнейшие действия.

Use Case 1:

1)Пользователь вводит адрес своей электронной почты и пароль.

2)Пользователь нажимает на кнопку “Войти”.

User Story 2: Как администратор, я хочу иметь возможность создавать и редактировать учебные группы и их состав, чтобы управлять данными.

Use Case 2:

1)Администратор нажимает на кнопку “Учебные группы”.

2)Администратор выбирает нужную группу.

2)Администратор нажимает на кнопку “Изменить состав группы”.

3)Администратор добавляет или удаляет студентов из учебной группы.

User Story 3: Как администратор, я хочу иметь возможность составлять и редактировать расписание учебного процесса, чтобы обеспечить организованное проведение занятий.

Use Case 3:

1)Администратор нажимает на кнопку “Расписание”.

2)Администратор нажимает на ячейку занятия, информацию о котором он хочет изменить.

3)Администратор нажимает на кнопку “Редактировать”.

4)Администратор изменяет информацию о данном занятии.

User Story 4: Как администратор, я хочу иметь возможность проставлять баллы за лабораторные работы, зачёты и экзамены, чтобы вести учет успеваемости студентов.

Use Case 4:

1)Администратор нажимает на кнопку “Учебные группы”.

2)Администратор выбирает нужную группу.

3)Администратор выбирает студента, которому хочет проставить баллы.

4)Администратор проставляет баллы за лабораторные работы, зачёты и экзамены.

User Story 5: Как администратор, я хочу иметь возможность просматривать и редактировать информацию в своем профиле, чтобы поддерживать актуальные данные.

Use Case 5:

1)Администратор нажимает на кнопку “Личный кабинет”.

2)Администратор нажимает на кнопку “Изменить” рядом с той информацией, которую хочет отредактировать.

3)Администратор редактирует информацию в своем профиле.

User Story 6: Как студент, я хочу иметь возможность просматривать свое расписание в удобном формате, чтобы быть в курсе занятий.

Use Case 6:

1)Студент нажимает на кнопку “Расписание”.

2)Студент просматривает свое расписание.

User Story 7: Как студент, я хочу иметь возможность просматривать карточку занятия, чтобы узнать больше информации о нем.

Use Case 7:

1)Студент нажимает на кнопку “Расписание”.

2)Студент нажимает на ячейку занятия, информацию о котором он хочет получить.

3)Студент просматривает карточку занятия.

User Story 8: Как студент, я хочу иметь возможность просматривать свои баллы, чтобы быть в курсе своей успеваемости.

Use Case 8:

1)Студент нажимает на кнопку “Успеваемость”.

2)Студент просматривает свои баллы.

User Story 9: Как студент, я хочу иметь возможность настраивать уведомления на почту о приближающихся зачётах/экзаменах, чтобы быть информированным о предстоящих событиях.

Use Case 9:

1)Студент нажимает на кнопку “Расписание”.

2)Студент нажимает на ячейку занятия,уведомление о зачете/экзамене которого хочет получить.

3)Студент нажимает на кнопку “Отправить напоминание о зачете/экзамене”.

User Story 10: Как студент, я хочу иметь возможность вносить заметки к карточкам занятий, чтобы сохранять дополнительную информацию.

Use Case 10:

1)Студент нажимает на кнопку “Расписание”.

2)Студент нажимает на ячейку занятия,к карточке которого хочет внести заметку.

3)Студент вводит заметку в текстовое окно “Добавьте заметку”.

User Story 11: Как студент, я хочу иметь возможность просматривать и редактировать информацию в своем профиле, чтобы поддерживать актуальные данные.

Use Case 11:

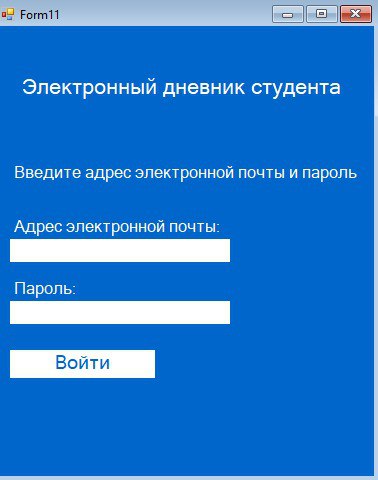
1)Студент нажимает на кнопку “Личный кабинет”.

2)Студент нажимает на кнопку “Изменить” рядом с той информацией, которую хочет отредактировать.

3)Студент редактирует информацию в своем профиле.

## 3.2. Требования к пользовательским интерфейсам.

Вход в приложение:

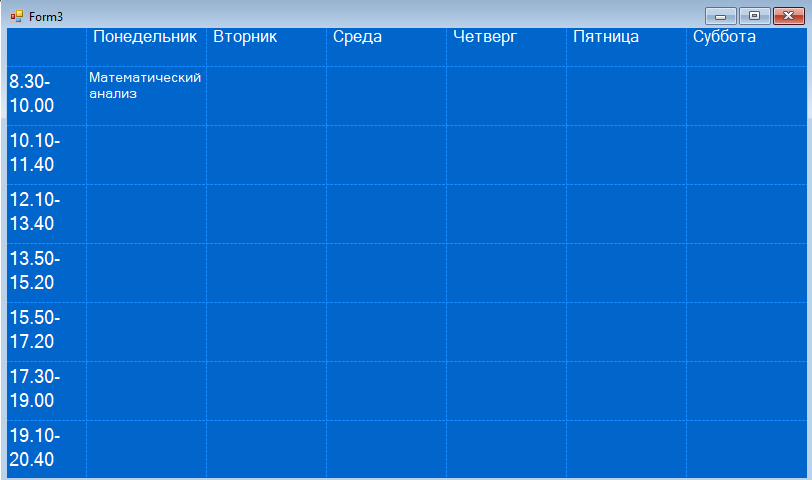


Функционал администратора:

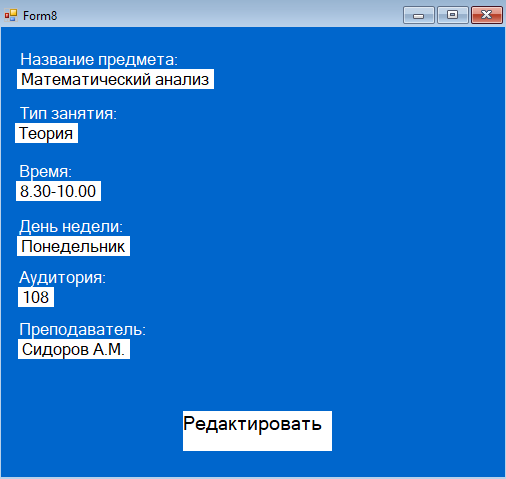
Администратор выбирает нужную опцию:



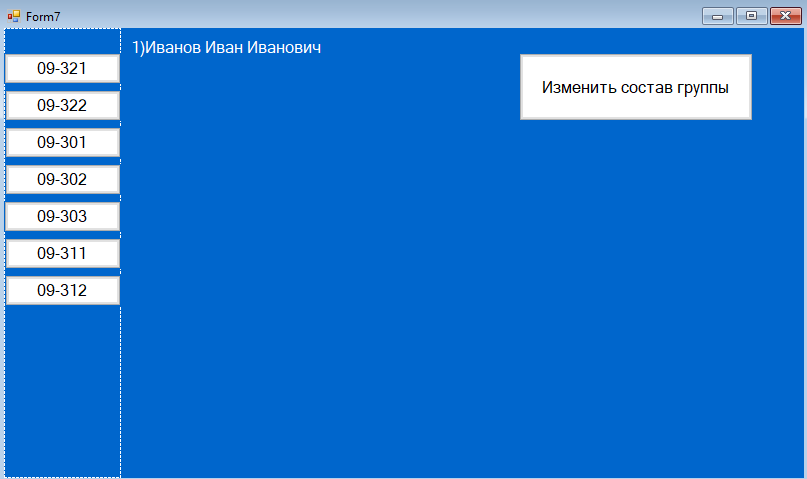
При нажатии на кнопку “Расписание” появляется следующее окно:



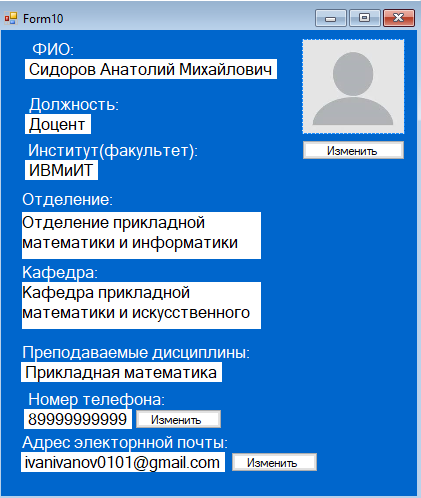
При нажатии на конкретную ячейку занятия появляется следующее окно:



При нажатии на кнопку “Учебные группы” появляется следующее окно:



При нажатии на кнопку “Личный кабинет” появляется следующее окно:

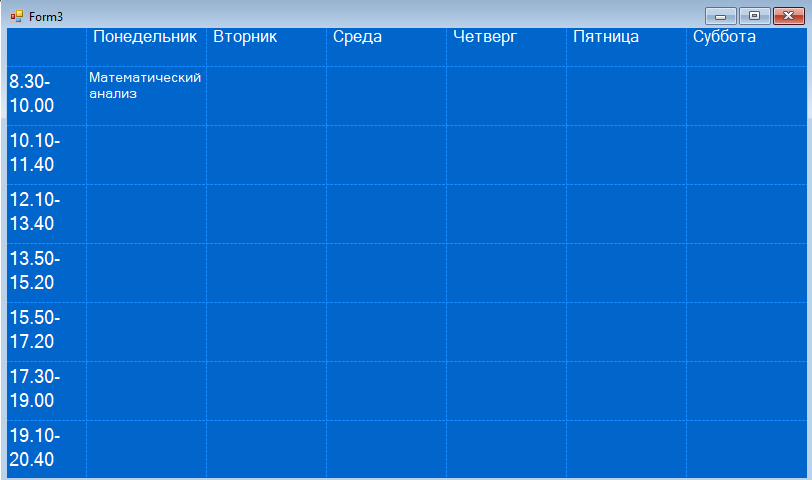


Функционал студента:

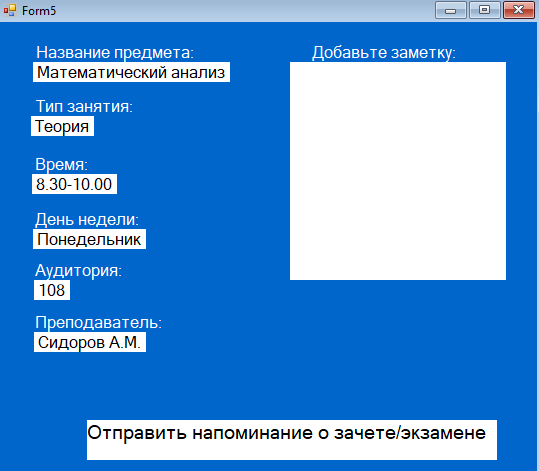
Студент выбирает нужную опцию:



При нажатии на кнопку “Расписание” появляется следующее окно:



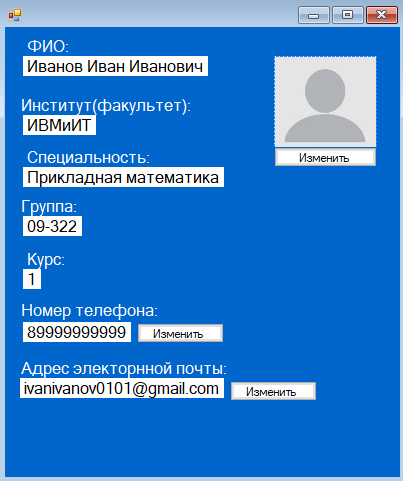
При нажатии на конкретную ячейку занятия появляется следующее окно:



При нажатии на кнопку “Успеваемость” появляется следующее окно:



При нажатии на кнопку “Личный кабинет” появляется следующее окно:



## 3.3. Требования к логической структуре БД.

