

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.

Компьютерная программа “Электронный дневник студента” предоставляет студентам доступ к просмотру расписания занятий, внесению заметок к занятиям, просмотру своего профиля, получению информации о текущей успеваемости студента. Программа также предоставляет функционал администратора, а именно возможность создавать и редактировать учебные группы и их состав, составлять и редактировать расписание учебного процесса, проставлять баллы за лабораторные работы, зачеты и экзамены.

1.3. Определения, акронимы и сокращения.

БД - База данных

1.4. Краткое содержание.

Глава 2 описывает обзор системы, подробно рассказывает о типах пользователей, окружении системы, обзоре вариантов использования и нефункциональных требованиях.

Глава 3 описывает детальные требования к системе, включая функциональные требования, требования к пользовательским интерфейсам и требования к логической структуре БД.

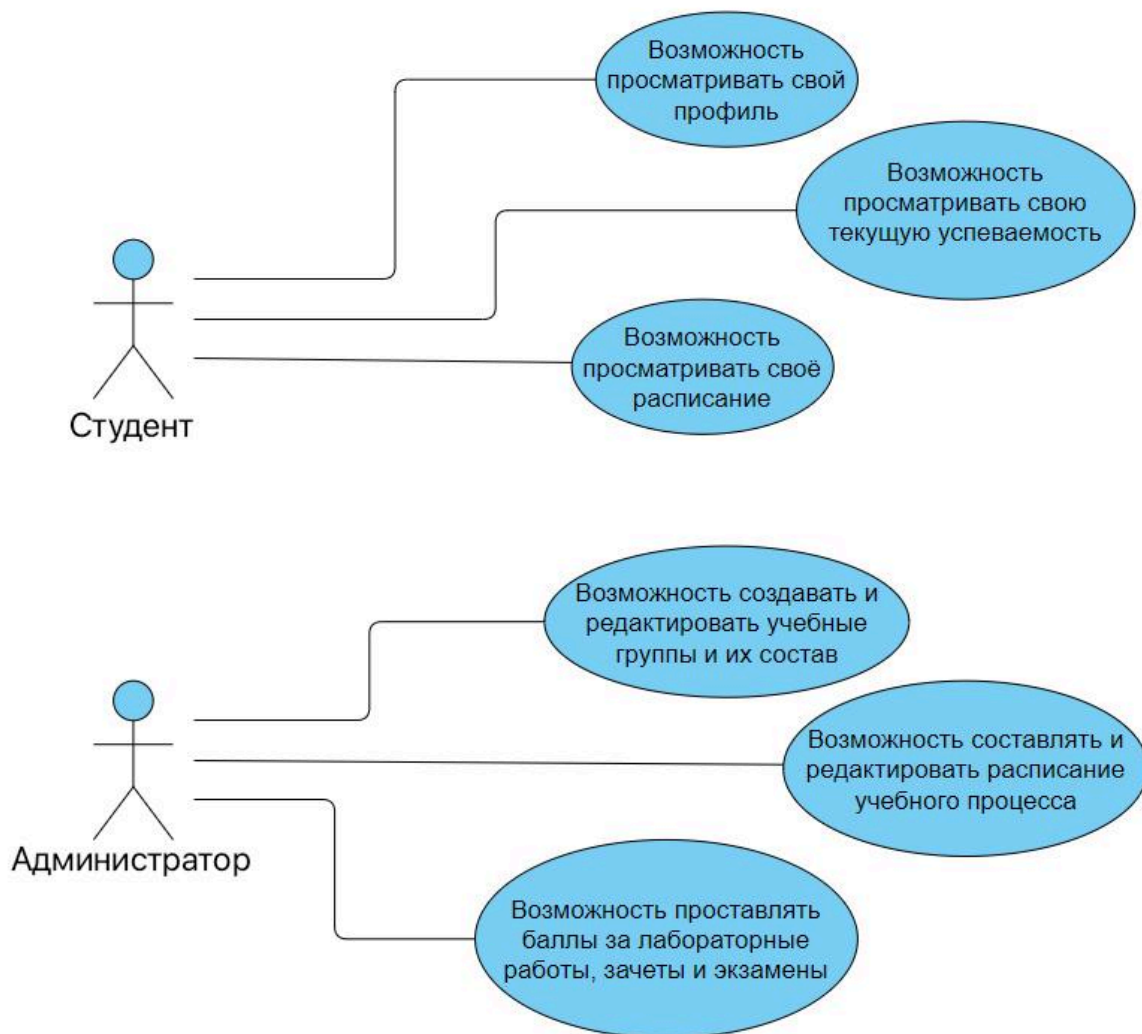
2. Обзор системы

2.1. Определение типов пользователей системы.

Студенты: пользователи, имеющие доступ к своему профилю, расписанию, информации о занятиях и успеваемости обучающегося

Администраторы: пользователи, имеющие возможность создавать и редактировать учебные группы и их состав, составлять и редактировать расписание учебного процесса, проставлять баллы за лабораторные работы, зачеты и экзамены.

2.2. Окружение системы.



2.3. Обзор вариантов использования.

Я, как студент, хочу иметь возможность просматривать свое расписание, чтобы знать когда проходят мои занятия.

Я, как студент, хочу иметь возможность добавлять заметки к занятиям, чтобы мне было проще планировать свой день, добавляя в эти заметки домашнее задание.

Я, как студент, хочу иметь возможность просматривать свой профиль, чтобы знать, в какой аккаунт я вошел.

Я, как администратор, хочу иметь возможность создавать и редактировать учебные группы, чтобы мне было проще работать с ними.

Я, как администратор, хочу иметь возможность составлять и редактировать расписание учебного процесса, чтобы где и когда я провожу занятия.

Я, как администратор, хочу иметь возможность проставлять баллы за лабораторные работы, зачеты и экзамены, для того чтобы студенты могли отслеживать свою успеваемость.

2.4. Нефункциональные требования.

Необходимо реализовать программу на языке программирования C#, с помощью интерфейса программирования приложений Windows Forms.

3. Детальные требования

3.1. Функциональные требования

Просмотр своего профиля:

Актеры: Студент

Описание: Студент может просматривать свой профиль.

Действия: Просмотр профиля студента.

Просмотр расписания:

Актеры: Студент

Описание: Студент может просматривать свое расписание занятий.

Действия: Просмотр расписания на текущий семестр.

Просмотр успеваемости:

Актеры: Студент

Описание: Студент может просматривать свою текущую успеваемость.

Действия: Просмотр текущей успеваемости.

Добавление заметок к занятиям:

Актеры: Студент

Описание: Студент может добавлять заметки к своим занятиям.

Действия:

1. Выбор занятия
2. Добавление заметки

Управление учебными группами:

Актеры: Администратор

Описание: Администратор может просматривать, создавать, редактировать и удалять учебные группы.

Действия:

1. Просмотр списка учебных групп
2. Создание новой учебной группы
3. Редактирование информации об учебной группе
4. Удаление учебной группы

Управление Расписанием:

Актеры: Администратор

Описание: Администратор может просматривать, создавать, редактировать и удалять расписание для учебных групп.

Действия:

1. Просмотр расписания для учебной группы
2. Создание нового расписания для учебной группы
3. Редактирование расписания для учебной группы
4. Удаление расписания для учебной группы

Проставление баллов:

Актеры: Администратор

Описание: Администратор может проставлять баллы за лабораторные работы, зачеты и экзамены.

Действия:

1. Выбор студента и учебного мероприятия
2. Проставление баллов за выполненное задание

3.2. Требования к пользовательским интерфейсам

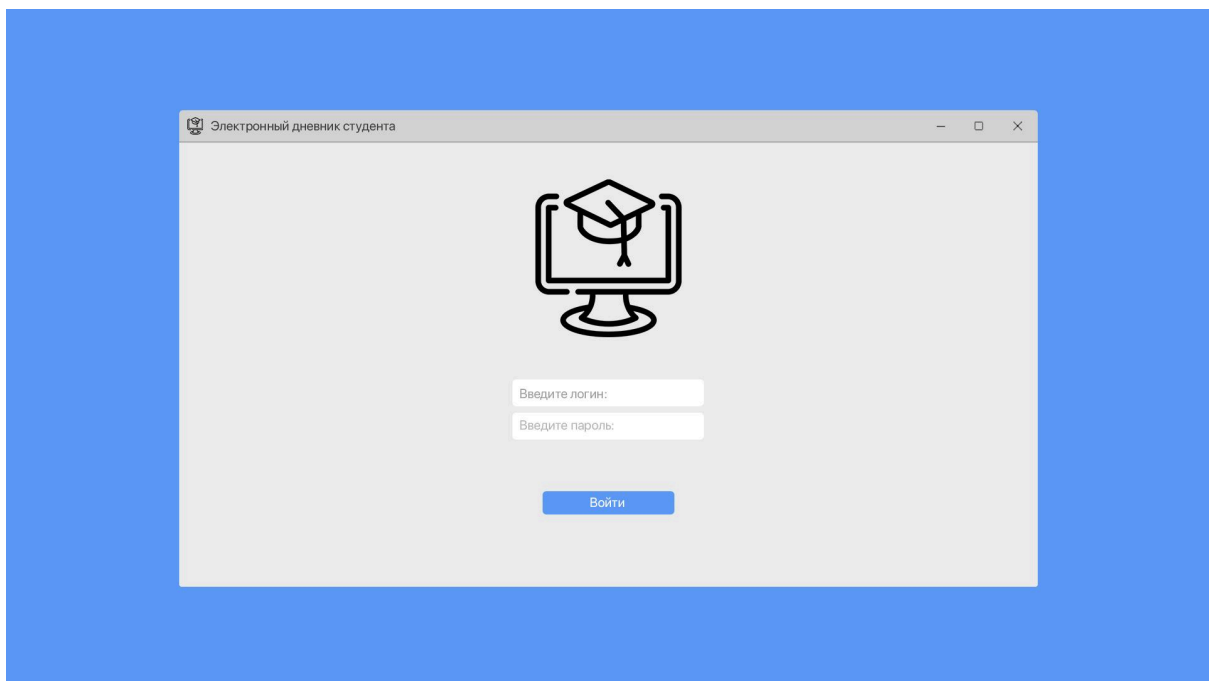


Рисунок 1 - окно входа, пользователь вводит данные и входит в аккаунт

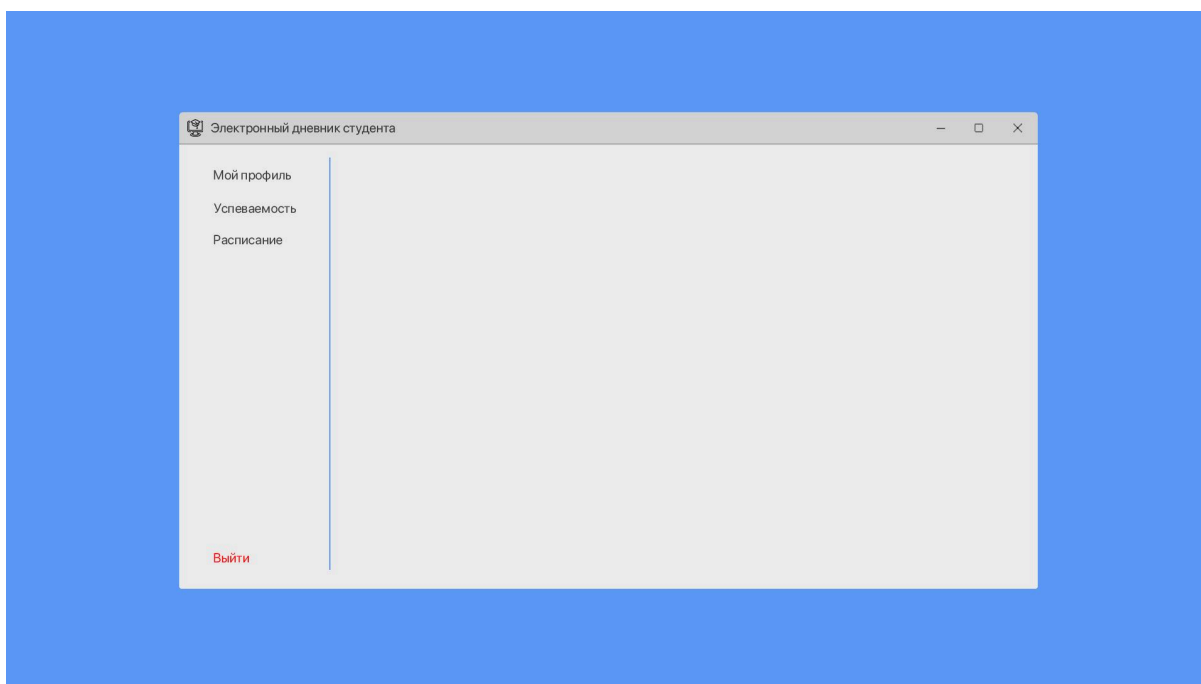


Рисунок 2 - Окно, встречающееся пользователю студенту, при входе в аккаунт

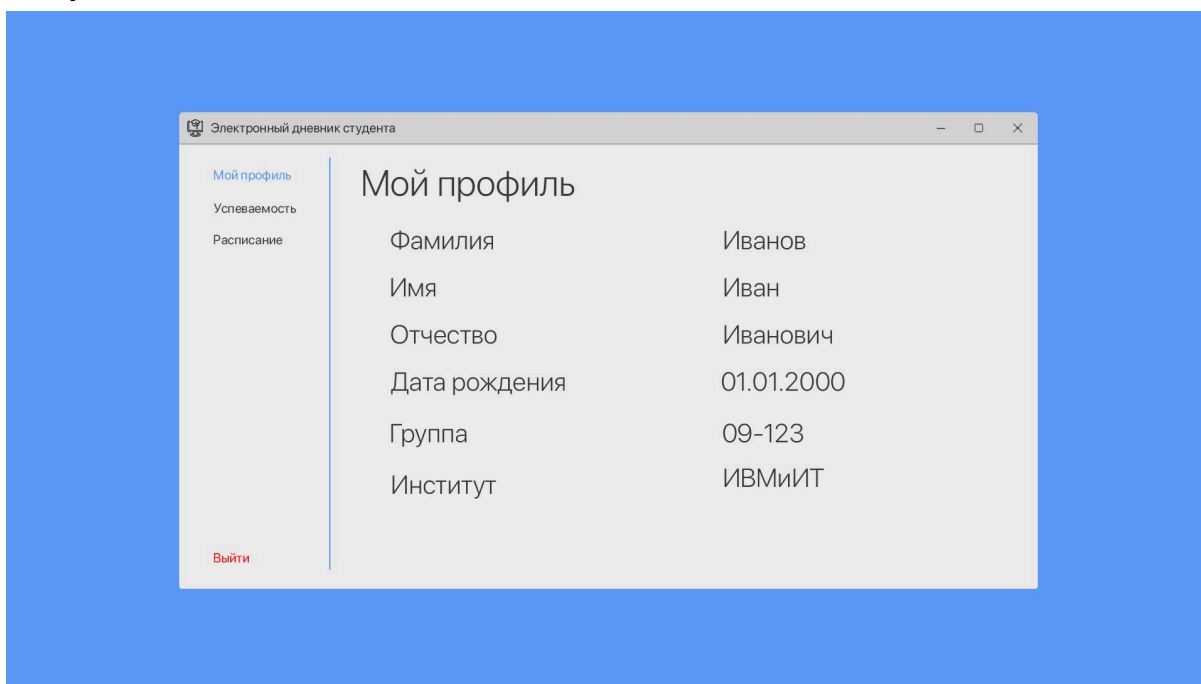


Рисунок 3 - Окно с данными о профиле студента

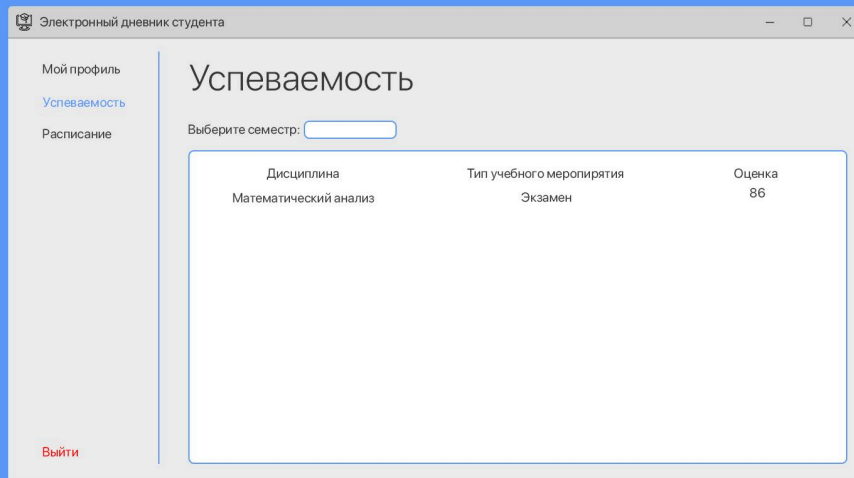


Рисунок 4 - Окно с успеваемостью студента, пользователь выбирает семестр и получает данные о своей успеваемости

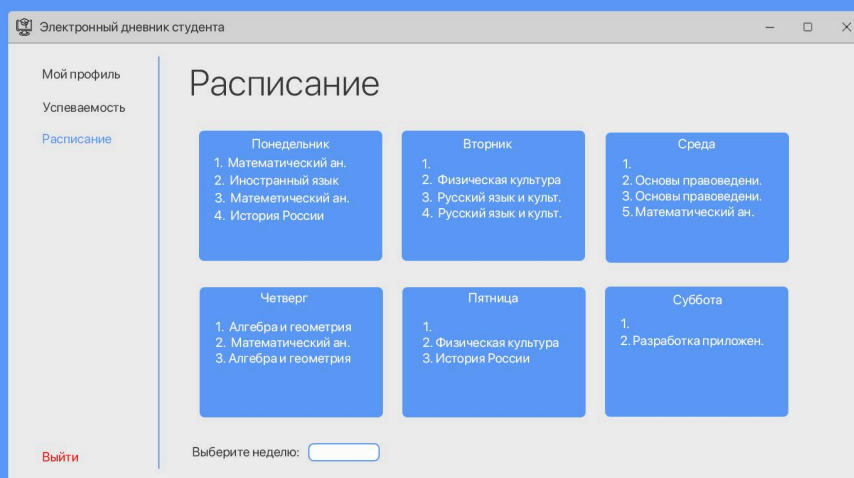


Рисунок 5 - Окно с расписанием занятий
Пользователь выбирает неделю (четную / нечетную) в белой панели и получает данные о своих занятиях на соответствующей неделе

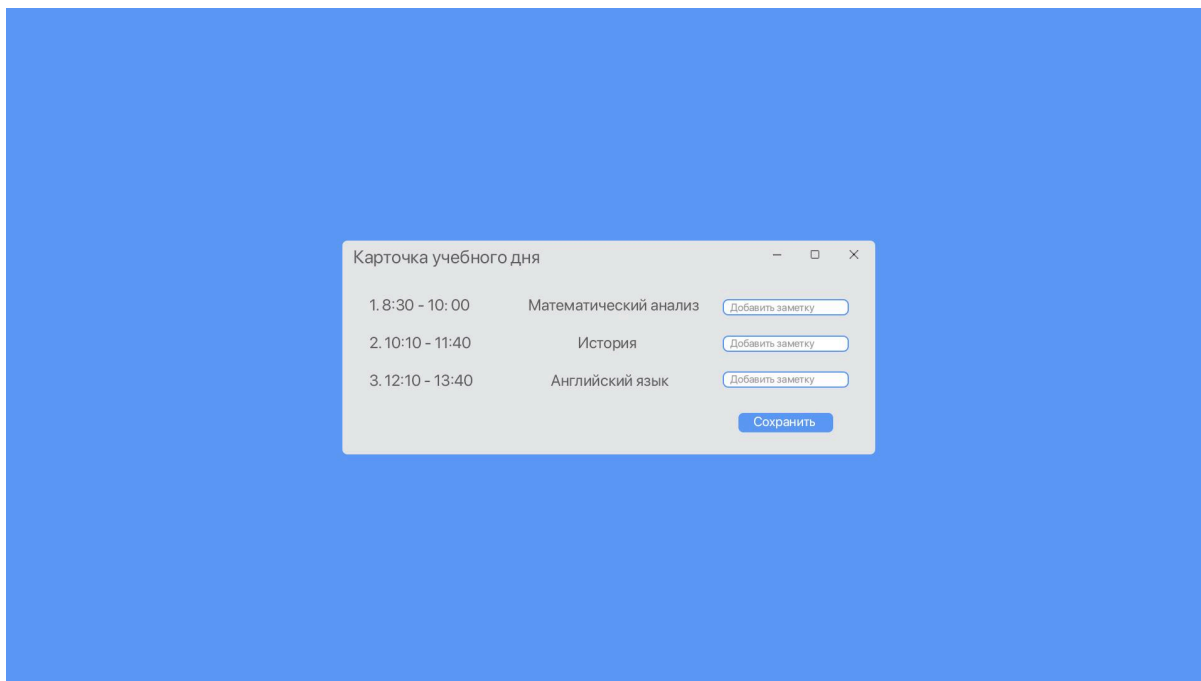


Рисунок 6 - При двойном нажатии на день в расписании вылезает новое окно с данными о занятиях в конкретный день. Пользователь имеет возможность оставлять заметки к занятиям

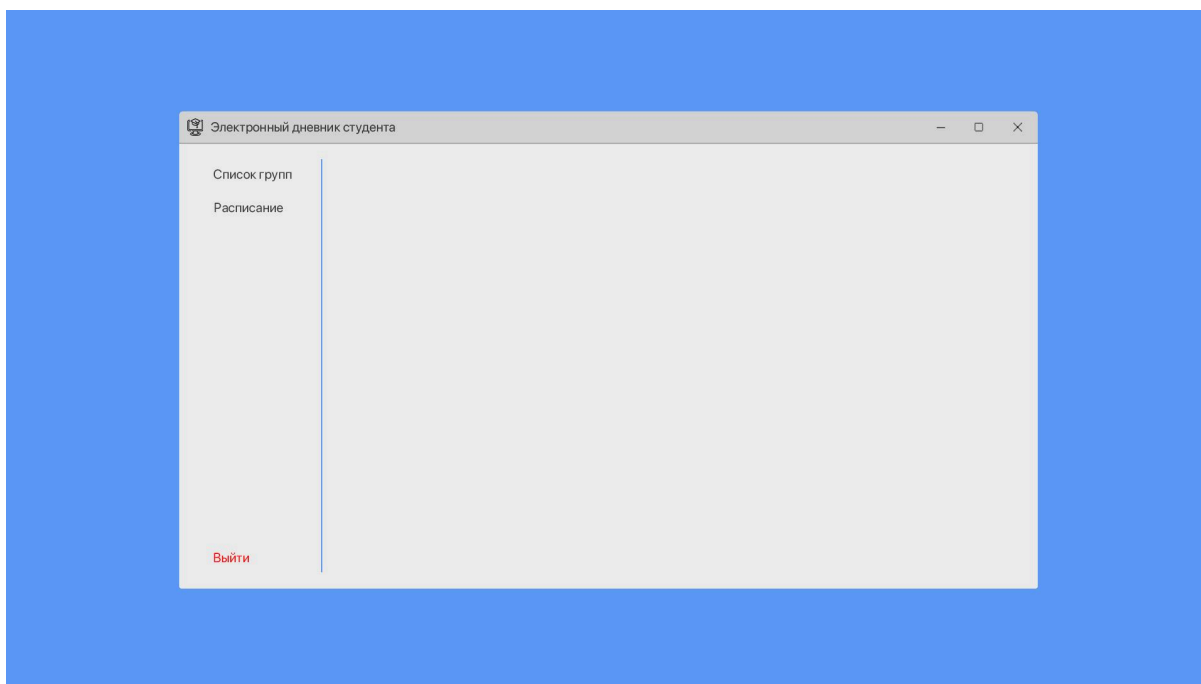


Рисунок 7 - Окно, встречающееся администратору, вошедшему в аккаунт

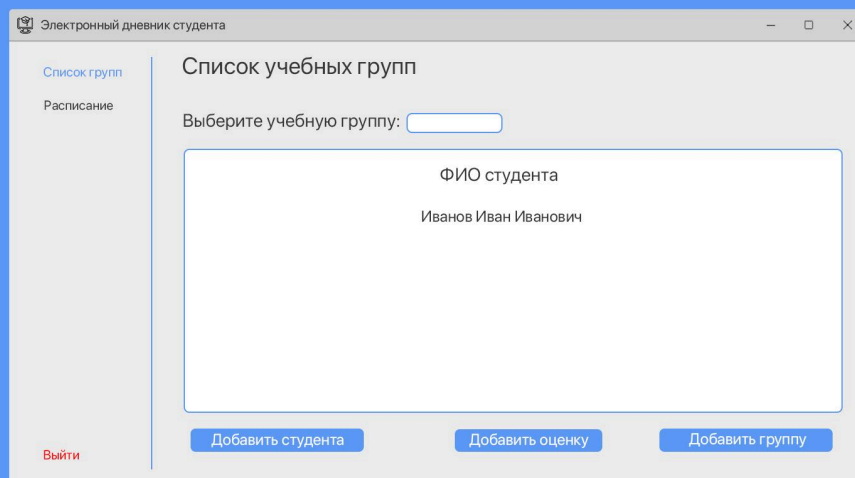


Рисунок 8 - Окно со списком учебных групп. Пользователь может создать учебную группу, добавить в открытую учебную группу студента по ID, добавить оценку студенту, а также выбрать учебную группу и просмотреть список её студентов

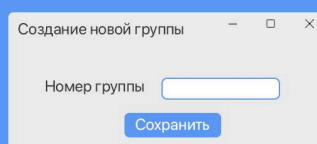


Рисунок 9 - Окно создания новой группы

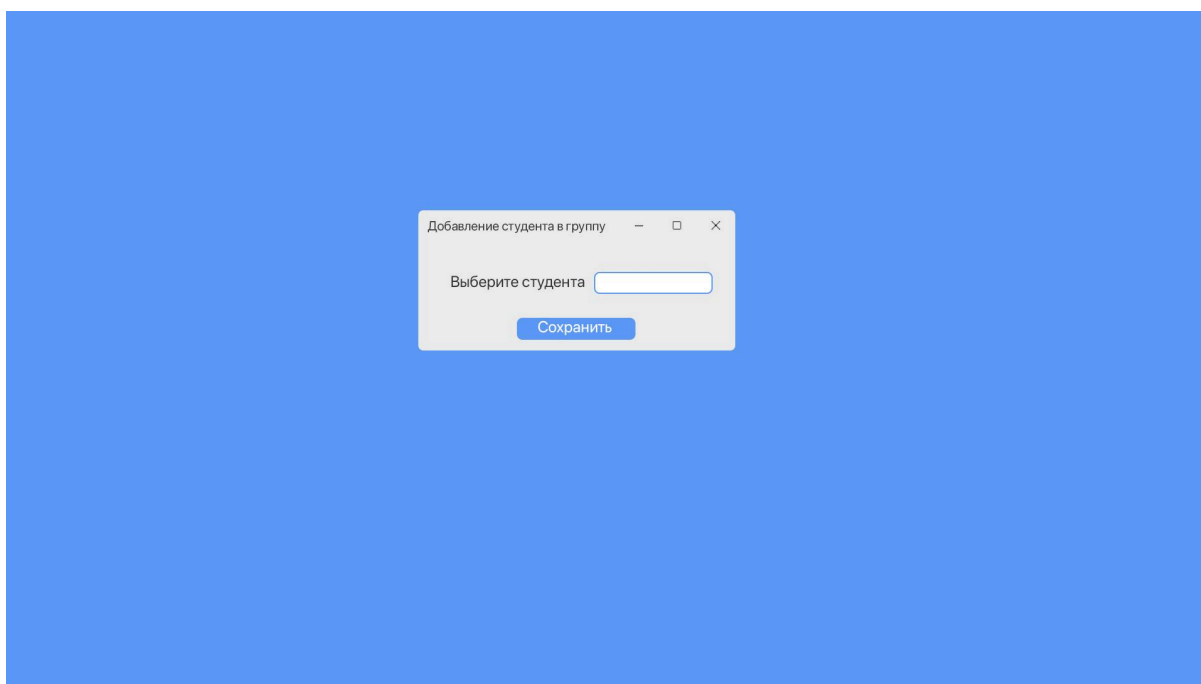


Рисунок 10 - Окно добавления студента в выбранную группу



Рисунок 11 - Окно добавления оценки студенту

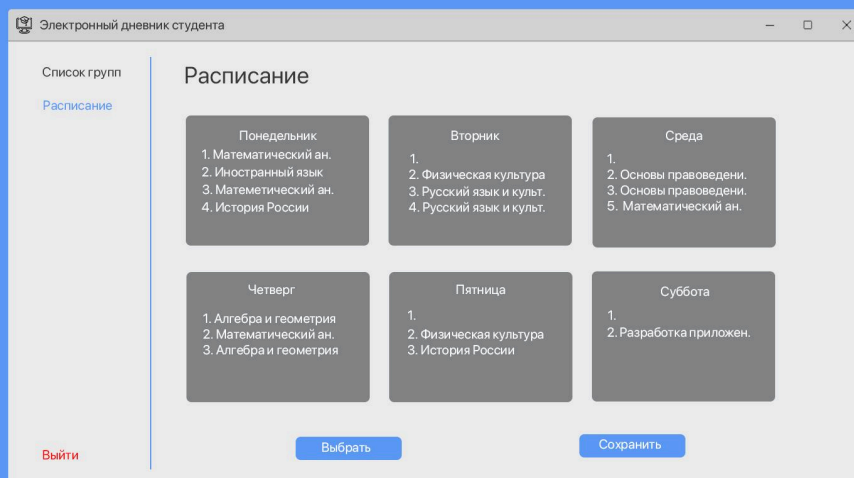


Рисунок 12 - Окно с расписанием. Пользователь-администратор выбирает учебную группу и получает данные о ее расписании занятий. Пользователь имеет возможность редактировать занятия, убирая и добавляя их. После завершения работы, пользователь должен сохранить изменения

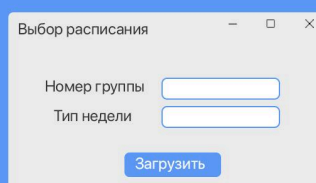


Рисунок 13 - Окно выбора расписания. Пользователь-администратор выбирает номер группы, тип недели и загружает расписание в приложение

3.3. Требования к логической структуре БД

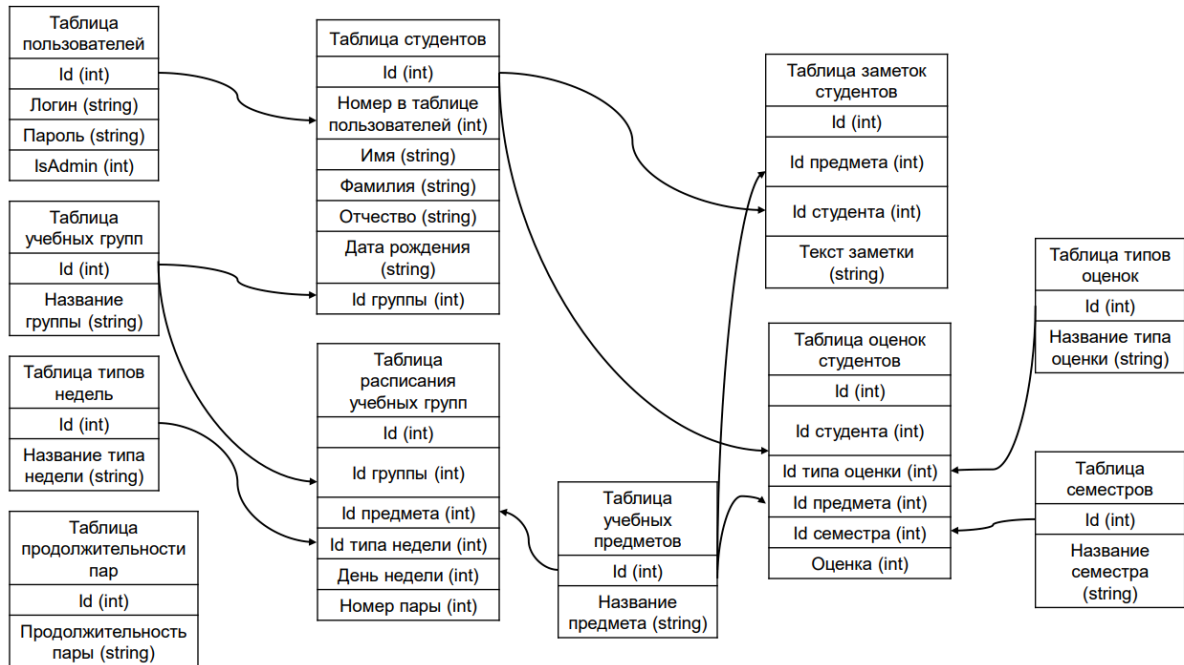


Рисунок 14 - Схема БД