**Министерство науки и высшего образования РФ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Казанский (Приволжский) федеральный университет»  
Институт вычислительной математики и информационных технологий**

Проект 1. Электронный дневник студента.

Аналитик: Кельмурзин Данил Анатольевич  
Команда: 8  
Дедлайн проекта: 30.03.2024 23:59  
Руководители: Фадеева Екатерина Владимировна, Латыпова Дина Сергеевна

**I. Введение**

* **1. Цель**

Создать «Электронный дневник студента». Программа «Электронный дневник студента» предоставляет студентам информацию об учебном процессе и возможность администраторам редактировать его.

**2. Краткая сводка возможностей**

1. Регистрация/авторизация
2. Личные кабинеты с разным уровнем прав(Студент/Администратор)
3. Редактирование личных кабинетов
4. Просмотр, создание заметок и редактирование расписания в зависимости от прав пользователя
5. Редактирование и создание учебных групп и их состава (право администратора)
6. Выставление оценок отдельно за лабораторные работы, зачеты, экзамены (право администратора)
7. Уведомления о приближающихся зачётах/экзаменах (за сутки до зачёта/экзамена) (функция студента)

* **3. Определения, акронимы и сокращения**

БД – База Данных

СУБД – Система Управления Базами Данных

Юскейс или ВИ – Use Case

* **4. Краткое содержание**

Далее будет расписаны роли пользователей, Use Case, User Stories, макет, БД

**II. Обзор системы**

* **1. Определение типов пользователей**

В системе существуют следующие роли: администратор, ученик

Администратор имеет доступ к

1. Создание и редактирование учебных групп и их состава.
2. Составление и редактирование расписания учебного процесса.
3. Проставление баллов за лабораторные работы, зачёты и экзамены.

Ученик имеет доступ к

1. Просмотр своего расписания в удобном формате.
2. Просмотр карточки занятия: информация о типе занятия, дата и время и т.д.
3. Внесение заметок к карточкам занятий.
4. Возможность просмотра и редактирования информации в профиле.

* **2. Окружение системы**

См. в вложение. Рис. 1

* **3. Обзор вариантов использований**

User stories:

1. Я как студент, хочу иметь возможность устанавливать заметки под любым предметом в расписании, чтобы я мог зафиксировать важную информацию или какое-нибудь дз.

2. Я как студент, хочу иметь возможность смотреть актуальную информацию о расписании, чтобы я всегда знал какая сейчас пара

3. Я как студент, хочу иметь возможность детально изучить информацию о паре: полное название предмета, время начала и конца, ФИО преподавателя и мои заметки

4. Я как студент, хочу иметь возможность быстрого редактирования профиля, если там была какая-нибудь ошибка в данных

5. Я как студент, хочу иметь безопасный личный кабинет со своим логином и паролем, чтобы моя информация не попала в руки третьих лиц

6. Я как студент, хочу иметь возможность получать уведомления о приближающихся зачётах/экзаменах (за сутки до зачёта/экзамена) на свою почту

7. Я как администратор, хочу иметь возможность ставить отметки отдельно для лабораторных работ, зачетов и экзаменов, чтобы студент всегда был в курсе о своей успеваемости

8. Я как администратор, хочу иметь возможность редактировать расписание пар, чтобы студенты имели всегда актуальную информацию

9. Я как администратор, хочу иметь возможность редактировать и создавать учебные группы и их состав.

10. Я как администратор, хочу иметь возможность прикреплять отдельную группу к какому-нибудь расписанию и иметь список этих объединений, чтобы легче в дальнейшем редактировать информацию

11. Я как администратор, хочу иметь безопасный личный кабинет со своим логином и паролем, чтобы моя информация не попала в руки третьих лиц и студенты не могли войти с моими правами

12. Я как администратор, хочу иметь особенные права, чтобы без проблем изменять и редактировать.

* **4. Нефункциональные требования.**

Язык – С#

Среда – Visual Studio

СУБД - MySQL

1. **Детальные требования**

* **1. Функциональные требования**

|  |  |
| --- | --- |
| Имя | UC 1. Сохранить запись заметки |
| Приоритет | 2-Should |
| Область действия | Карточка занятия в программе «Электронный дневник студента». |
| Актор | Студент |
| Описание | 1. Студент нажимает на предмет  2. Система открывает окно «Карточка занятия»  3. Пользователь записывает данные в поле «Заметка» и нажимает кнопку «Сохранить»  4. Система сохраняет заметку |
| Результат | Сохраненная заметка |

|  |  |
| --- | --- |
| Имя | UC 2. Возможность просматривать расписание |
| Приоритет | 1-Must |
| Область действия | Окно «Расписание» в программе «Электронный дневник студента». |
| Актор | Студент/Администратор |
| Описание | 1. Из окна «Главная страница» пользователь нажимает кнопку «Расписание»  2.1. Если роль пользователя – администратор, то программа выводит список расписаний всех групп  2.2. Если роль пользователя – студент, то программа проверяет номер группы студента из профиля и показывает расписание этой группы |
| Результат | Расписание |

|  |  |
| --- | --- |
| Имя | UC 3. Возможность получить дополнительную информацию о предмете |
| Приоритет | 2-Should |
| Область действия | Карточка занятия в программе «Электронный дневник студента». |
| Актор | Студент/Администратор |
| Описание | 1. Студент нажимает на предмет  2. Система открывает окно «Карточка занятия»  3. На этом окне расписана вся дополнительная информация: полное название предмета, ФИО преподавателя, время начала и конца пары, дата и заметка |
| Результат | Дополнительная информация в окне «Карточка занятия» |

|  |  |
| --- | --- |
| Имя | US 4. Просмотр и редактирование профиля |
| Приоритет | 1-Must |
| Область действия | Окно «Главное окно» |
| Актор | Студент/Администратор |
| Описание | 1. Студент/Администратор открывает окно «Главное окно»  2. В окне профиля есть поля информации: id (уникальный номер студента/администратора), ФИО, номер группы (если студент), номер телефона, почта, роль (студент/администратор)  3. При нажатии кнопки «Редактирование» пользователь получает возможность изменить поля (все кроме id и группы)  4. Пользователь сохраняет изменения при нажатии кнопки «сохранить» |
| Результат | Просмотр и редактирование профиля |

|  |  |
| --- | --- |
| Имя | US 5. Получать уведомления о приближающихся зачётах/экзаменах |
| Приоритет | 3-Could |
| Область действия | Почта студента |
| Актор | Студент |
| Описание | 1. Система за день до экзамена/зачета отправляет на почту, указанную в профиле студента, сообщение, которое предупреждает о скорых экзаменов |
| Результат | Уведомление на почту |

|  |  |
| --- | --- |
| Имя | US 6. Возможность ставить отметки отдельно для лабораторных работ, зачетов и экзаменов |
| Приоритет | 1-Must |
| Область действия | Окно «Выставление баллов» |
| Актор | Администратор |
| Описание | 1. Администратор нажимает на кнопку «Выставить баллы», которое видно только при соответствующей роли в главном окне  2. Открывается окно, требующие указать группу и предмет  3. Система проверяет данные, внесенные пользователем  4.1. В случае провала система выводит сообщение о соответствующей ошибки  4.2. В случае успеха система выводит таблицу «\*Имя группы\* + \*Имя предмета\*» с такими столбцами: ФИО студентов, баллы за лабораторную работу, баллы за зачет, баллы за экзамен.  5. Администратор может редактировать последние 3 колонки  6. После нажатия на кнопку «Сохранить» система сохраняет внесенные пользователем данные, и студенты могут увидеть эти баллы в окне «Успеваемости» |
| Результат | Возможность ставить отметки отдельно для лабораторных работ, зачетов и экзаменов |

|  |  |
| --- | --- |
| Имя | US 7. Возможность редактировать расписание пар |
| Приоритет | 1-Must |
| Область действия | Окно «Редактирование расписания» |
| Актор | Администратор |
| Описание | 1. Система делает кнопку «Редактирование расписания» в главном окне видимой, если у пользователя роль администратора  2. При нажатии на эту кнопку, система выводит список номеров групп  3. При нажатии на номер группы, система открывает расписание этой группы  4. Администратор вводит изменения и нажимает кнопку «Сохранить»  5. Система сохраняет изменения и студенты могут увидеть эти изменения |
| Результат | Редактирование и сохранение расписание |

|  |  |
| --- | --- |
| Имя | US 8. Возможность редактировать и создавать учебные группы и их состав. И их просмотр. |
| Приоритет | 1-Must |
| Область действия | Окно «Группы и их состав» |
| Актор | Администратор |
| Описание | 1. Система делает кнопку «Группы» видимой, если у пользователя роль администратора  2. При нажатии по ней, система открывает окно «Группы и их состав»  3. В этом окне будет находится список «Добавленные группы» с уникальными номерами уже созданных групп и кнопка «Создать новую группу»  4.1. Если администратор нажмет на номер группы из списка, то система выведет окно список студентов этой группы  4.2 Если администратор нажмет на кнопку «Добавить группу», то система выведет окно «Добавить группу», в котором есть поле номера группы, список с добавленными администратором именами, список студентов, не принадлежащих никакой группе, и кнопка «Добавить»  5. При нажатии на ФИО из списка студентов, не принадлежащих никакой группе, система добавляет этого студента в список добавленных  5. При нажатии кнопки «Добавить», система переносит значения поля номера группы и списка добавленных в список групп |
| Результат | Возможность редактировать и создавать учебные группы и их состав. И их просмотр. |

|  |  |
| --- | --- |
| Имя | US 9. Регистрация/Авторизация |
| Приоритет | 1-Must |
| Область действия | Окно «Регистрация»/ «Авторизация» |
| Актор | Пользователь |
| Описание | 1. При запуске приложения будет открываться окно с выбором: авторизация и регистрация  2.1.1 При нажатии кнопки «Регистрация», система выведет кнопку «Зарегистрироваться» и такие поля: Имя, Фамилия, Отчество, Группа, Почта (Будет использоваться в качестве логина), Пароль, Код для администратора (поле может быть пустым)  2.1.2 При нажатии кнопки «Зарегистрироваться», система заполнит профиль, если код был правильный, то профиль будет иметь роль Администратора, если код не совпал, то система выдаст окно с соответствующей ошибкой, если поле будет пустым, то профиль будет иметь роль Студента  2.2.1 При нажатии на кнопку «Авторизация» система выведет поля Логин и Пароль  2.2.1.1 Если поля совпали с данными, то система открывает окно «Главная страница»  2.2.1.2 Если поля не совпали с данными, то система выведет окно с соответствующей ошибкой |
| Результат | Регистрация/Авторизация |

* **2. Требования к пользовательскому интерфейсу**

См. в вложение. Рис. 2-10

* **3. Требования к логической структуре БД**

\*Вторая строка – тип данных соответствующего столбца\*

Пользователь (Таблица заполняется с помощью регистрации)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| user\_id | user\_name | user\_surname | user\_patronymic | user\_role | user\_email | InTheGroupOrNot | user\_telephone | user\_mark\_  laba | user\_mark\_  test | user\_mark\_  exam |
| INT UNSIGNED | VARCHAR  (128) | VARCHAR  (128) | VARCHAR  (128) | INT(1) | VARCHAR  (128) | TINYINT(1) | VARCHAR(11) | INT UNSIGNED  (100) | INT UNSIGNED  (100) | INT UNSIGNED  (100) |

Группы (Заполняется администратором)

|  |  |  |
| --- | --- | --- |
| group\_id | group\_number | group\_user\_id |
| INT UNSIGNED | INT UNSIGNED | INT UNSIGNED |

Расписание Дня (Заполняется администратором)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| schedule\_id | schedule\_data | sheduleDay\_NumberInWeek | sheduleDay\_NumberGroup | schedule\_lesson\_name |
| INT UNSIGNED | DATE |  |  | VARCHAR(128) |

Предмет (lesson\_note заполняется студентом, lesson\_date не может повторяться и устанавливается в зависимости от расписания)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| lesson\_id | lesson\_name | lesson\_professor | lesson\_date | lesson\_note |
| INT UNSIGNED | VARCHAR  (128) | VARCHAR  (128) | DATETIME | VARCHAR(246) |

Оценка (Ставится администратором)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| mark\_id | mark\_users\_id | mark\_lesson\_name | mark\_laba | mark\_test | mark\_exam |
| INT UNSIGNED | INT UNSIGNED | VARCHAR  (128) | INT UNSIGNED  (100) | INT UNSIGNED  (100) | INT UNSIGNED  (100) |

Схему таблиц БД см. в вложение рис.12

***Вложения***

Рис.1

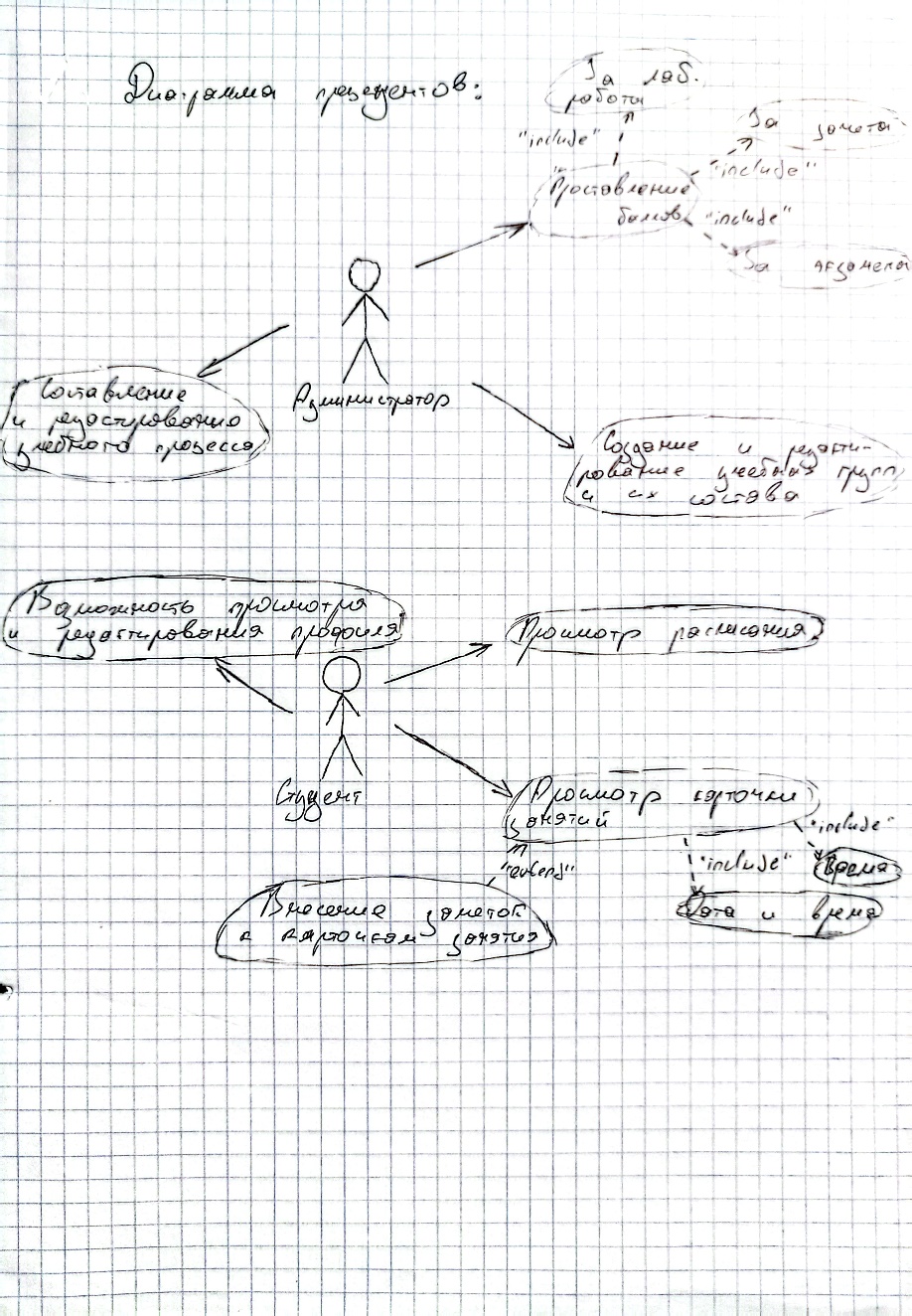


Рис. 2-3 (Регистрация/Авторизация)

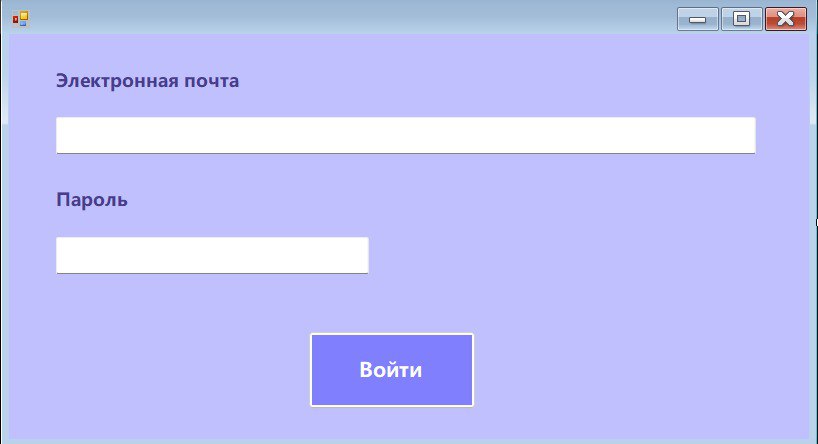
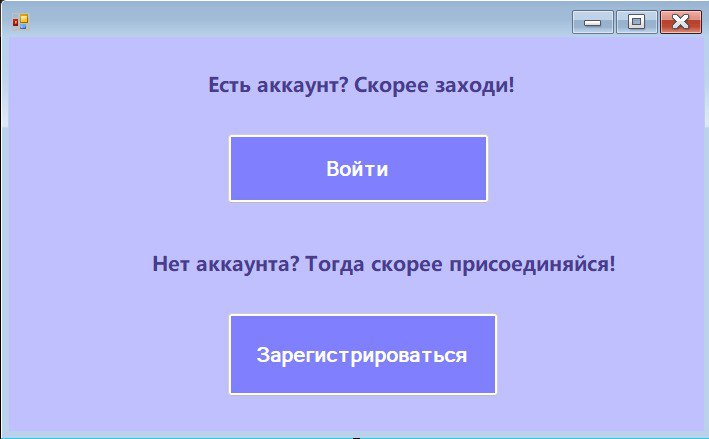
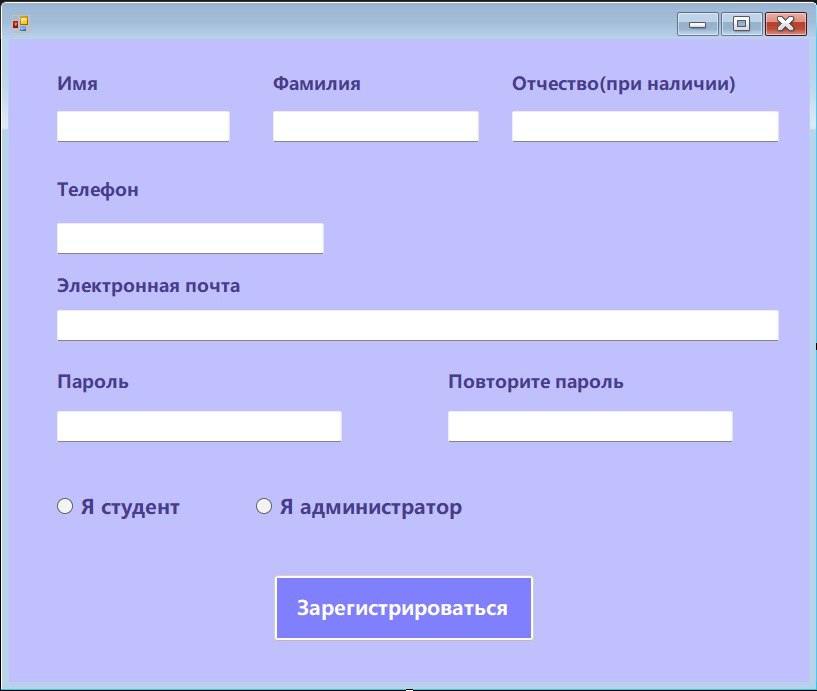


Рис. 4-5 (Профиль пользователя, администратора)

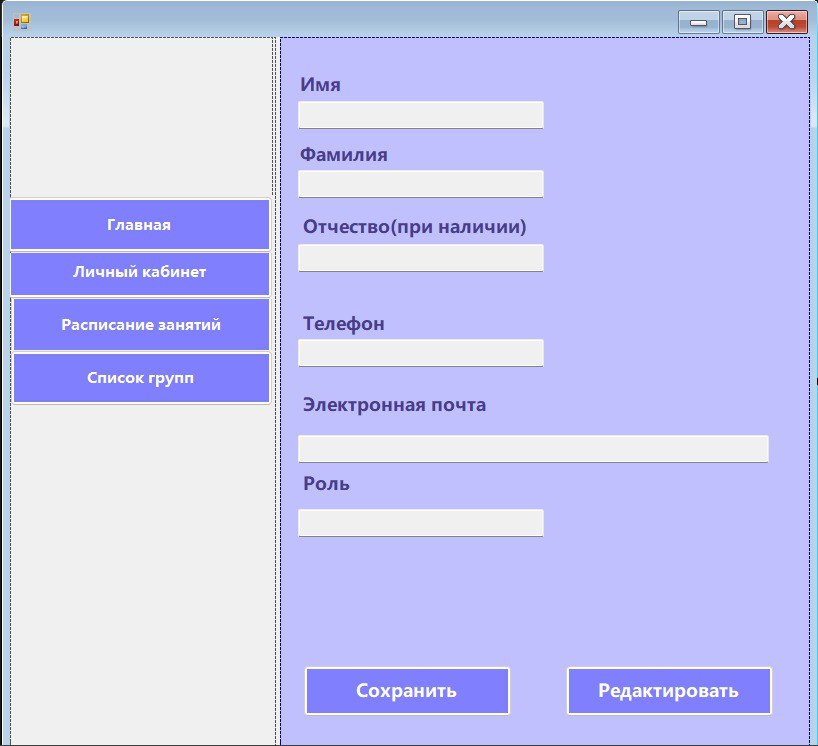
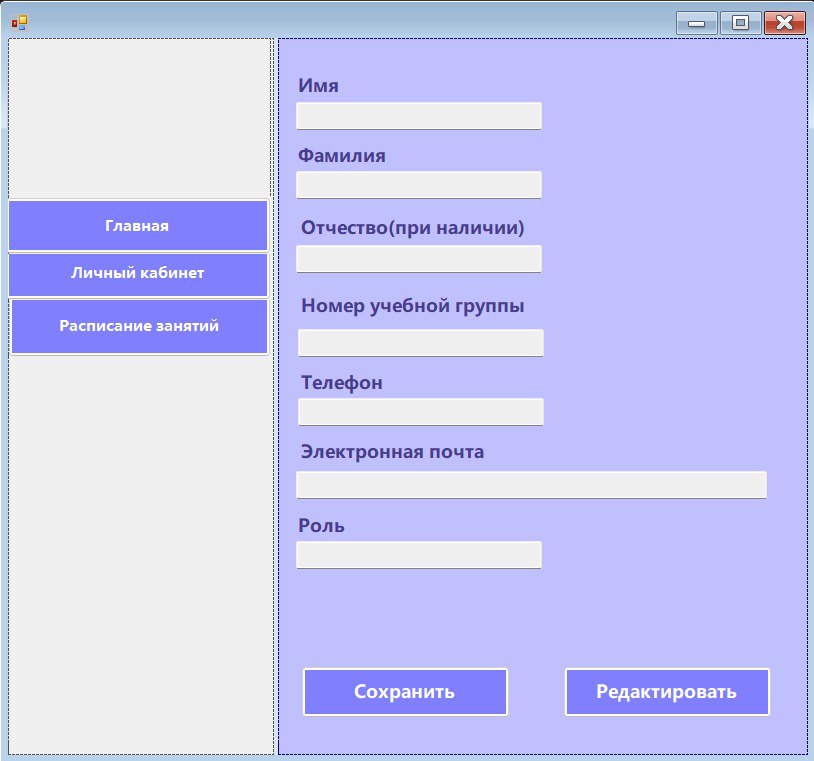


Рис. 6 (Главное окно)

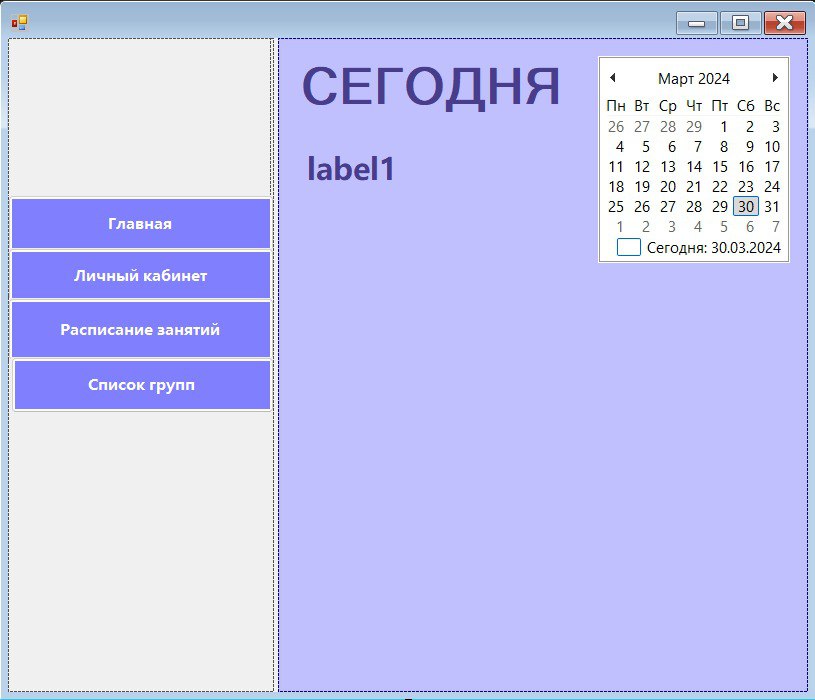


Рис. 7-8 (Расписание администратора и студента)

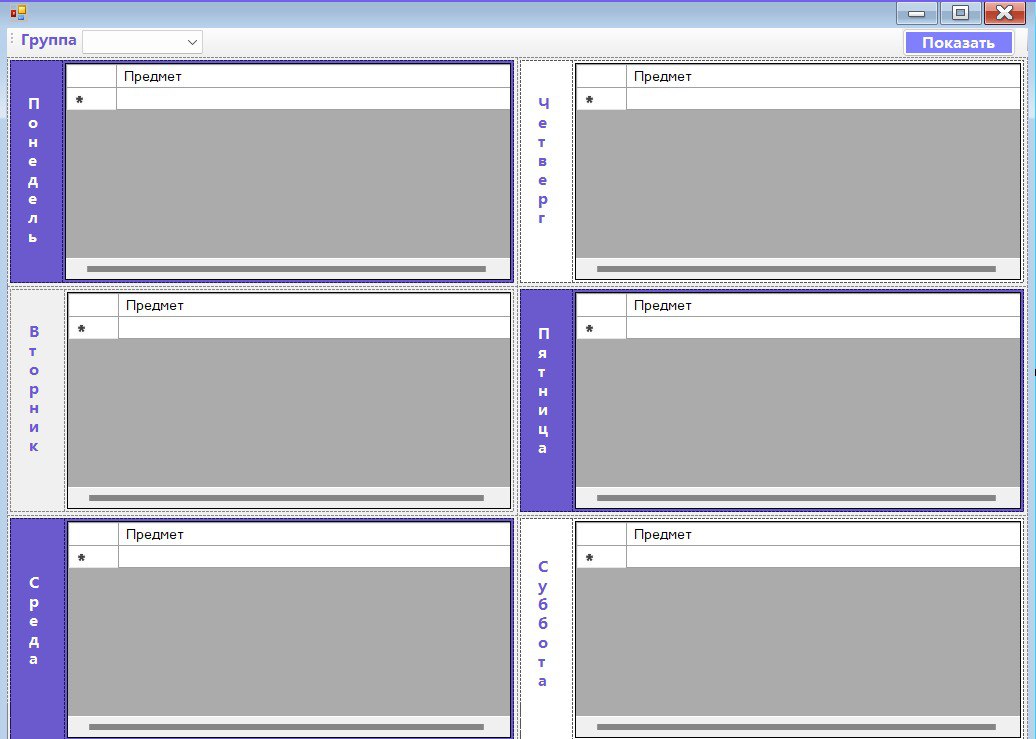
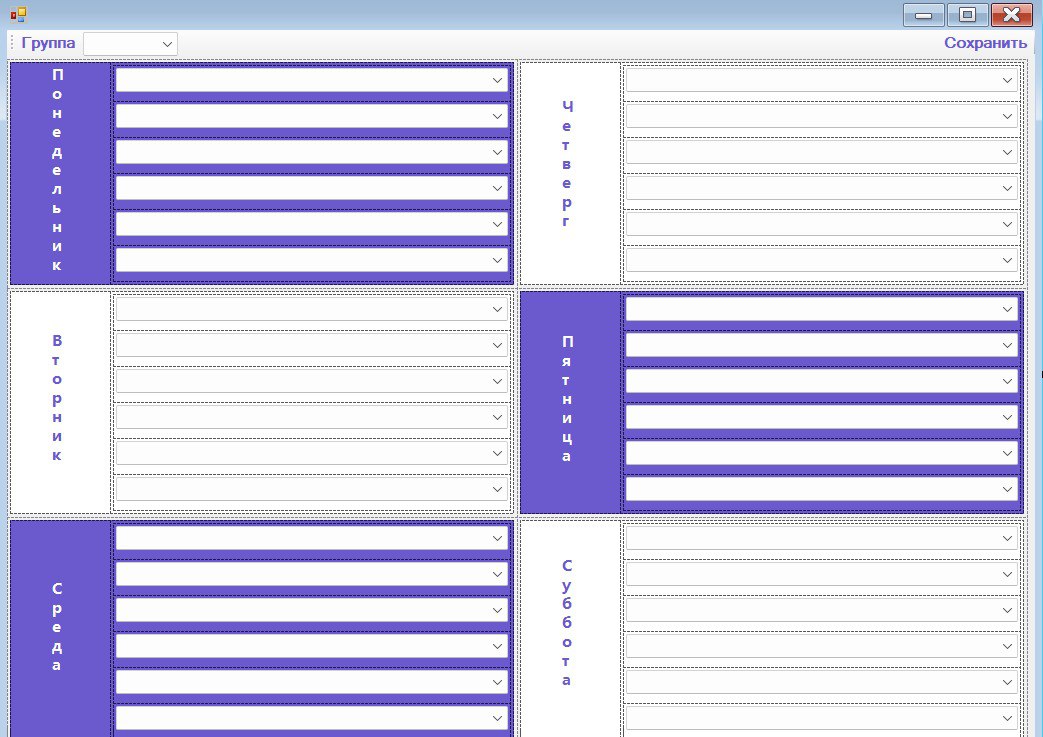


Рис.9

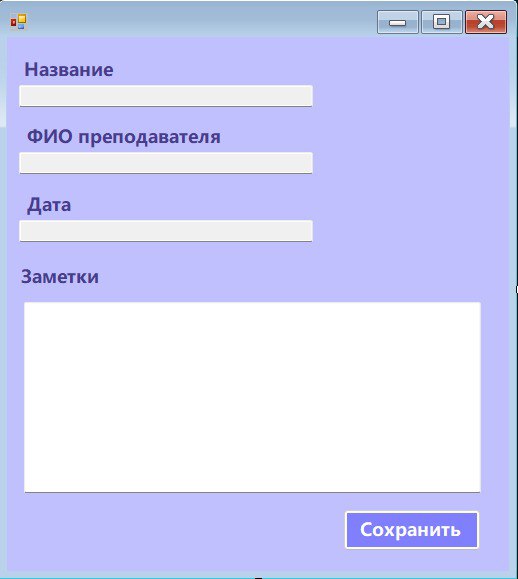


Рис. 10-11(Таблица успеваемости для студента и адмнистратора)

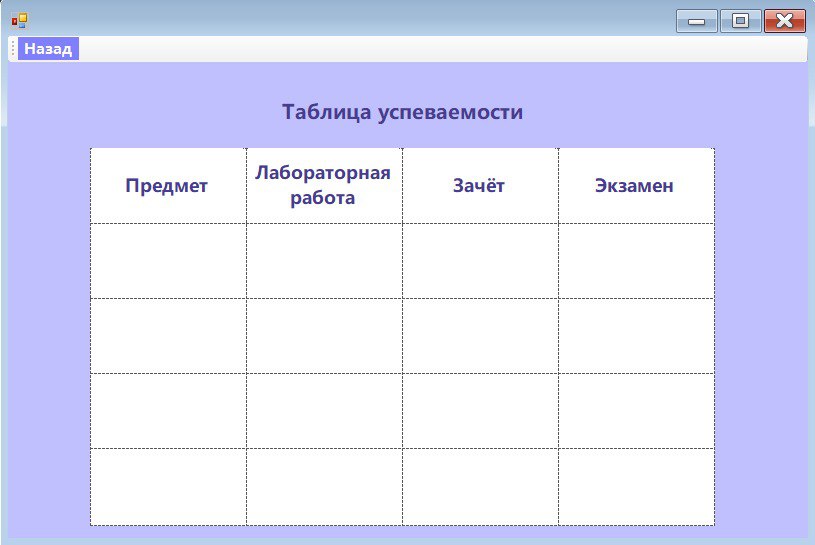


Рис.12(Диаграмма базы данных)

