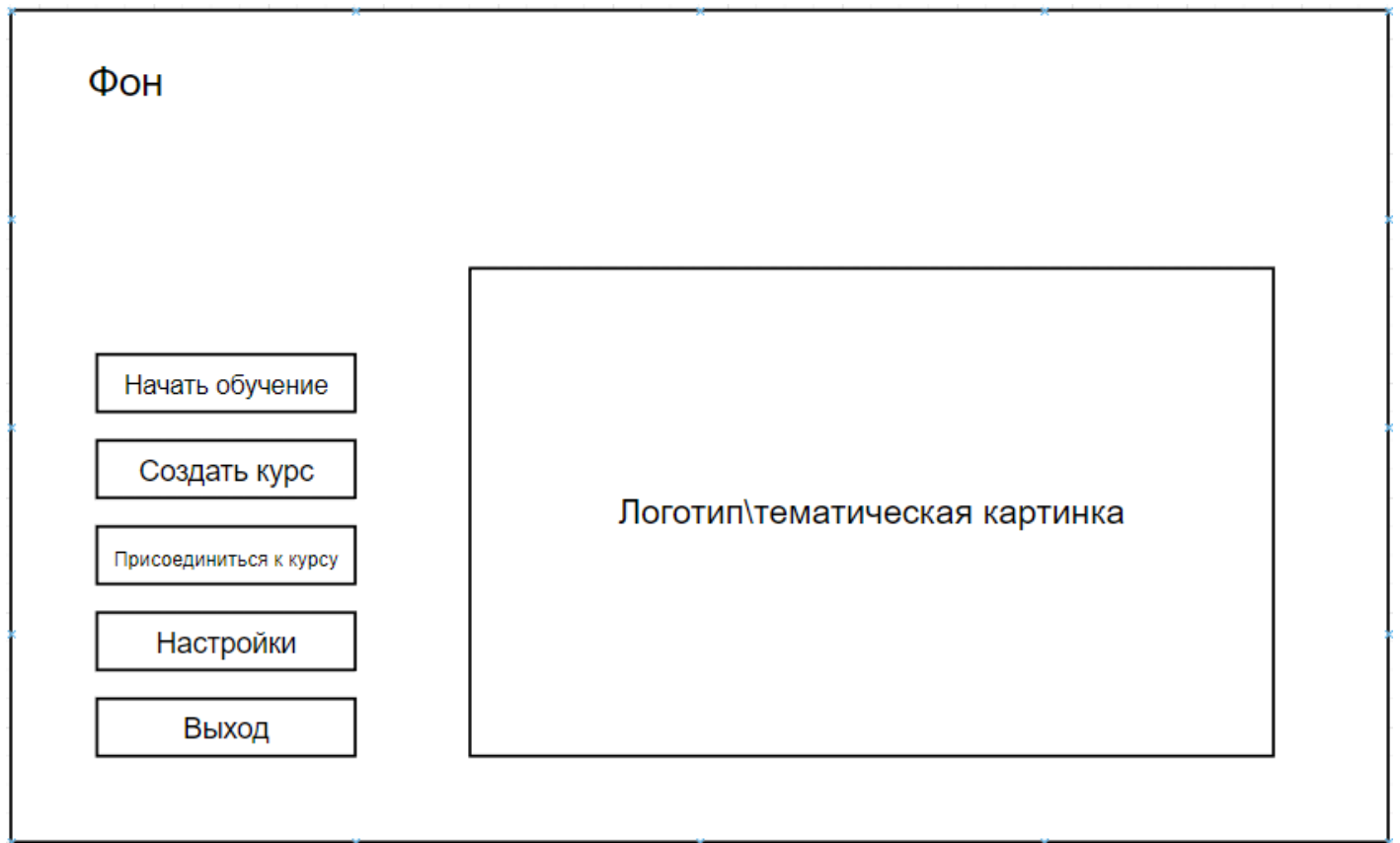


Разработка базового расписания

Прототипы экранных форм:

1. Прототип главного экрана: Главный экран состоит из фона, тематической картинки и группы кнопок: начать обучение, создать курс, присоединится к курсу, настройки, выход.

При нажатии кнопки «Начать обучение» открывается окно с выбором базовых тем и разделов (прототип 2), при нажатии кнопки «создать курс» открывается окно в котором представлен конструктор для создания курса (прототип 3), при нажатии кнопки «присоединиться к курсу» открывается окно с выбором курса, при нажатии кнопки «настройки» открывается окно с возможными настройками приложения (прототип 4), при нажатии кнопки «Выход» программа закрывается.



2. Прототип экрана с выбором базового курса: экран состоит из двух выпадающих списков: выбор темы, выбор раздела. Двух кнопок начать и кнопки назад в виде стрелки



При нажатии кнопки «Назад» открывается главное меню (прототип 1).

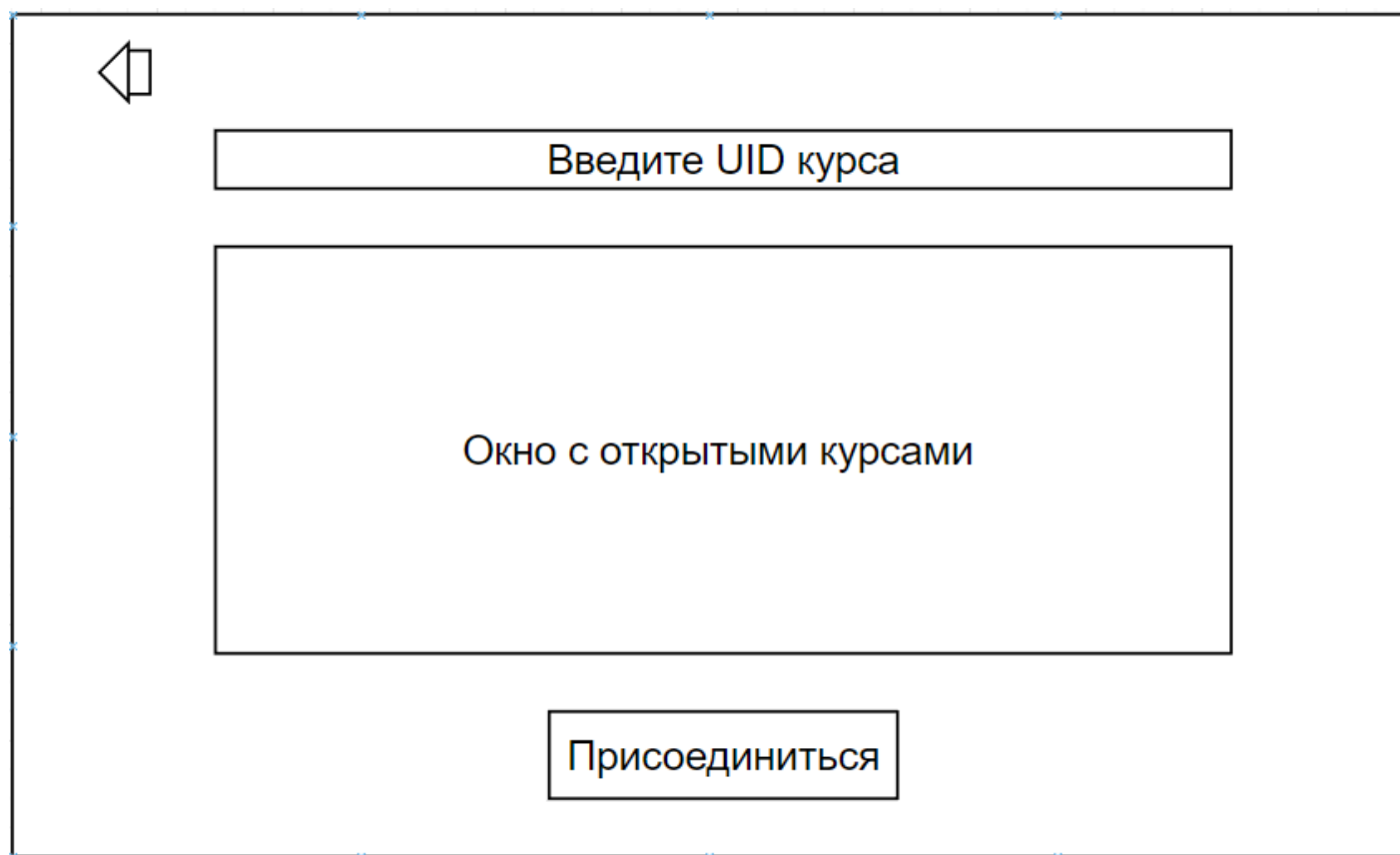
3. Прототип экрана с созданием курса: экран состоит из 4 выпадающих списков: выбрать тему, выбрать раздел, кол-во практик, кол-во лаб. работ. Поля для ввода названия курса, 3-х кнопок: назад в виде стрелки, кнопка в виде логического элемента, определяет будет ли курс закрытым, и кнопки создать.



Закрытый курс

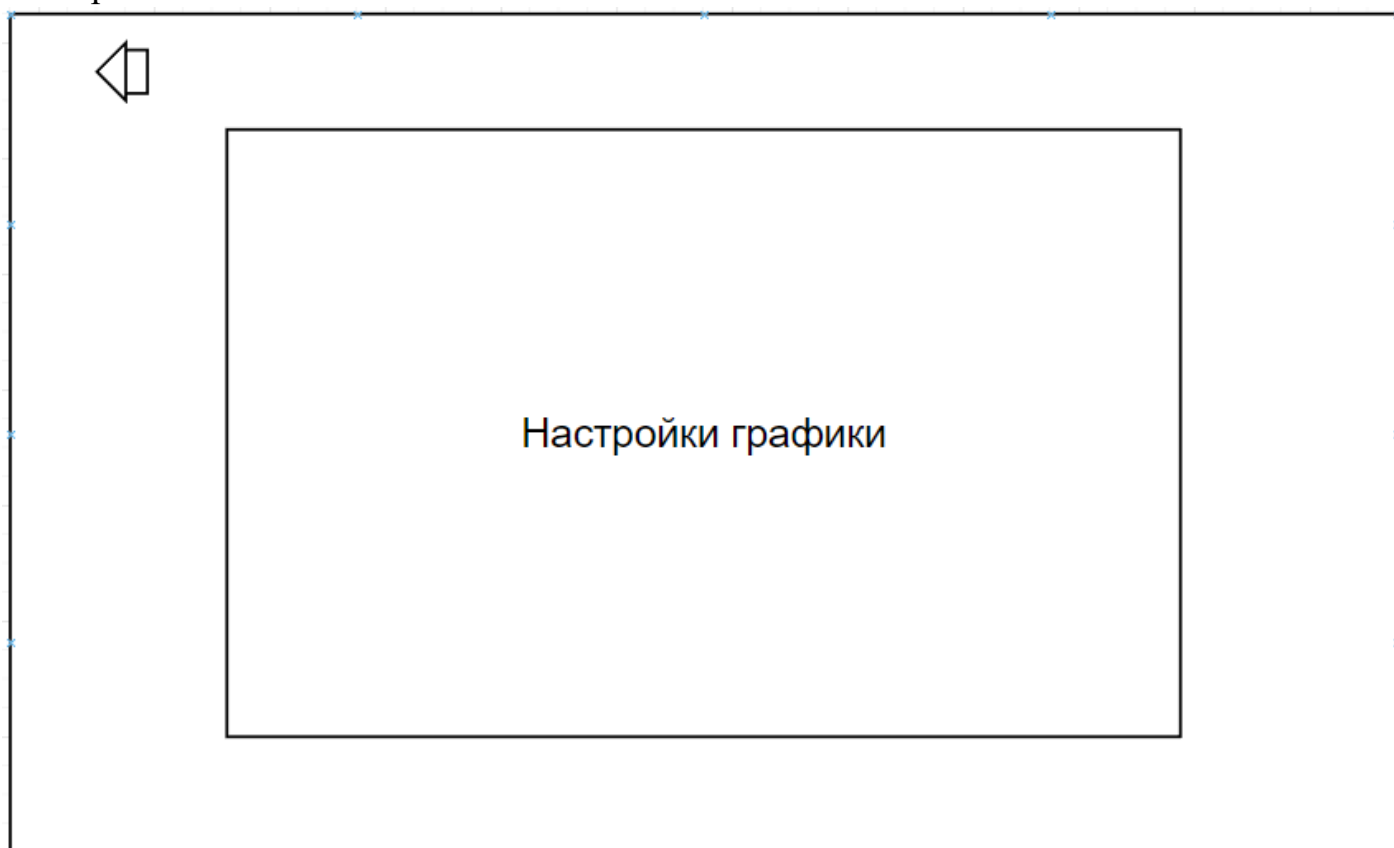
☐

4. Прототип экрана присоединится к курсу: экран состоит из поля для ввода идентификатора курса, окна с открытыми курсами, которые можно выбрать и присоединиться и кнопки присоединится при нажатии которой пользователь начинает проходить составленную программу.



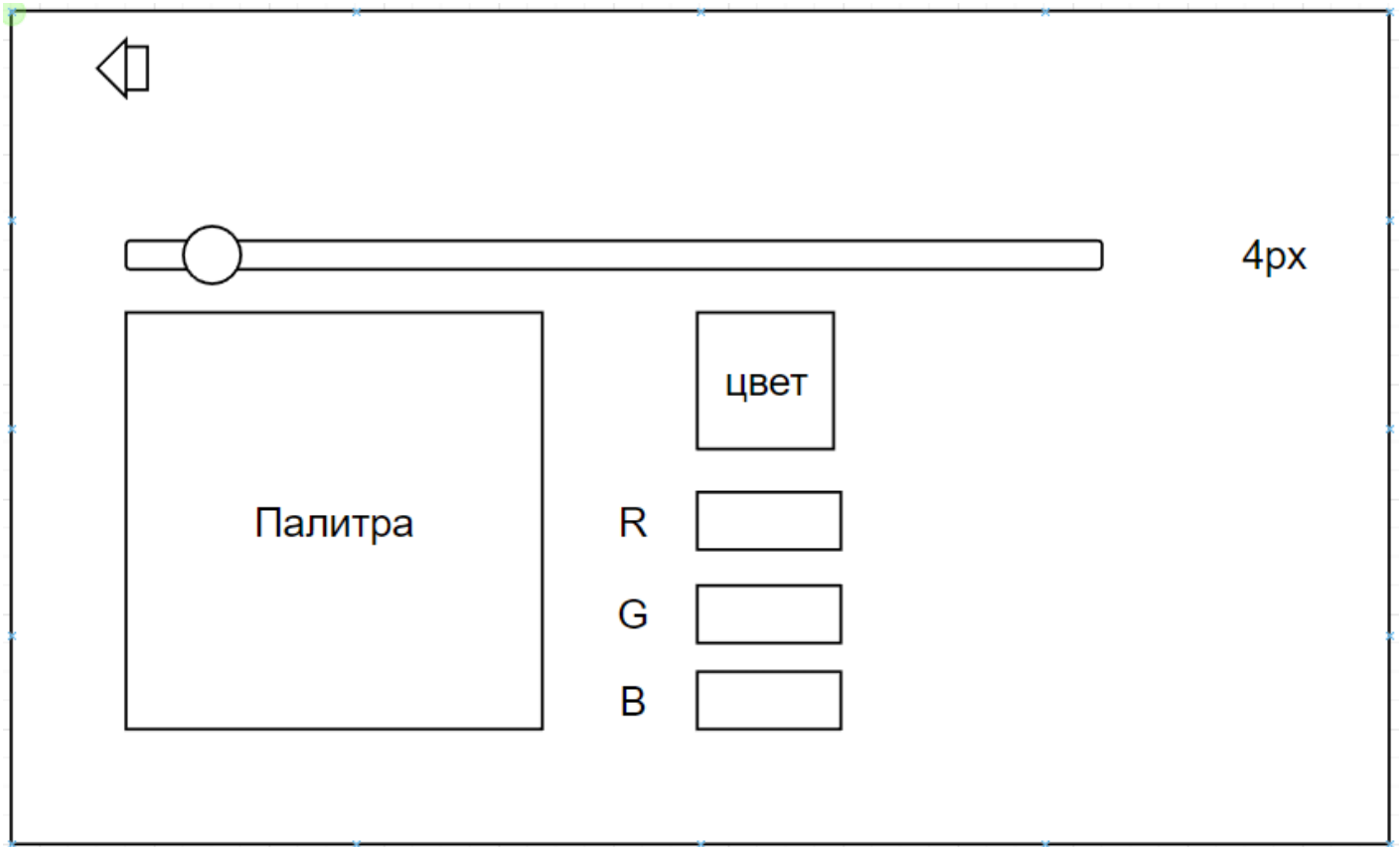
The wireframe shows a screen layout for joining a course. It features a back arrow icon in the top-left corner. Below it is a rectangular input field labeled "Введите UID курса". Underneath the input field is a large rectangular area labeled "Окно с открытыми курсами". At the bottom center of the screen is a rectangular button labeled "Присоединиться".

5. Прототип экрана настройки: экран настройки состоит из окна в котором представлены различные хар-ки связанные с разрешением и качеством картинки.

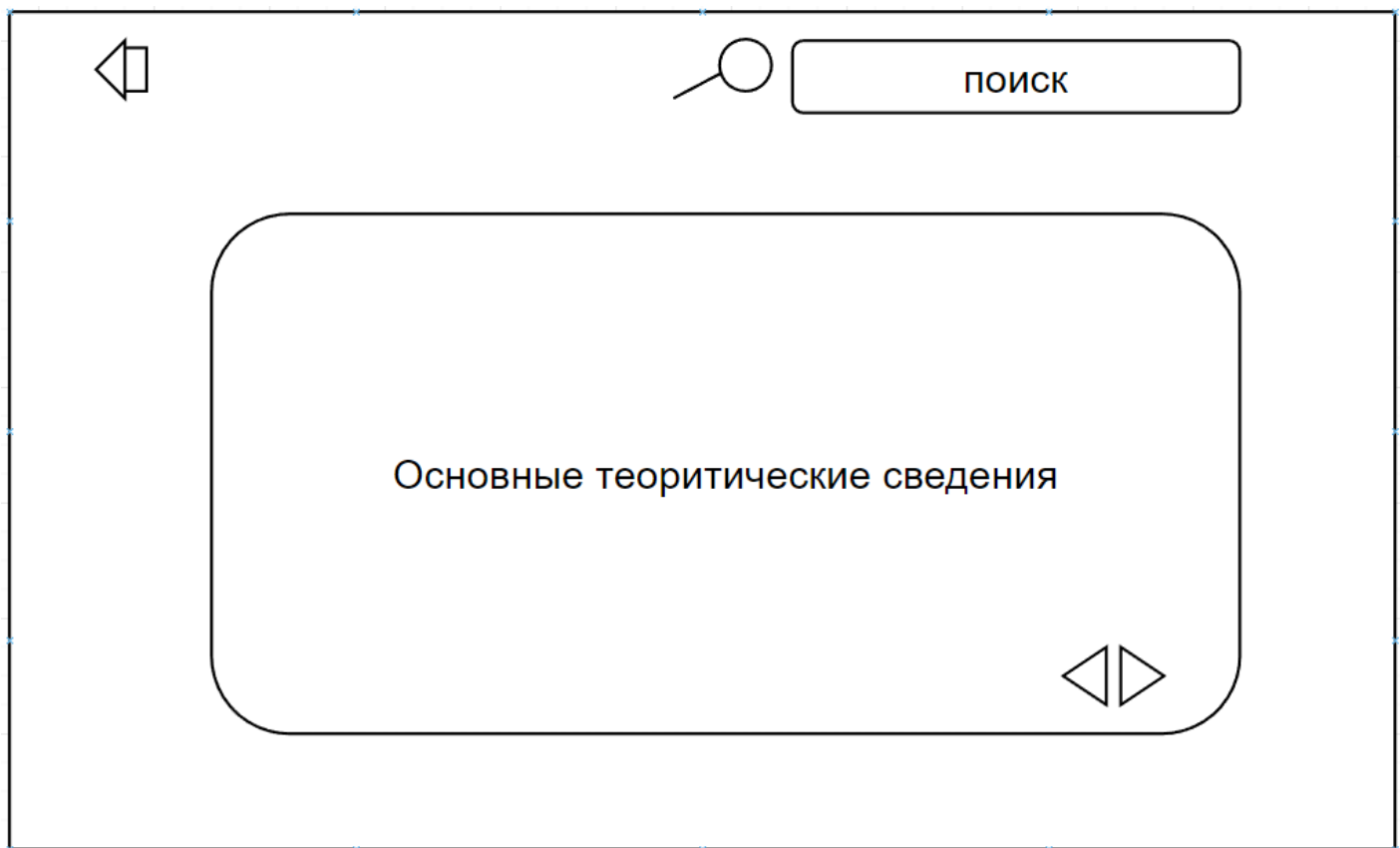


The wireframe shows a screen layout for graphics settings. It features a back arrow icon in the top-left corner. Below it is a large rectangular area labeled "Настройки графики".

6. Прототип окна с настройкой пера: экран состоит из двух основных настроек цвет и размер. Для изменения размера используется слайдер, для выбора цвета окно с палитрой. Также для удобства сделано окно для отображения выбранного пользователем цвета.



7. Прототип окна взаимодействия со справочным материалом: окно состоит из окна где написана теория по разделу, поля для быстрого поиска материала, и двух кнопок переключателей страниц.



8. Прототип окна для введения ответа по практическому материалу: окно состоит из Области с текстом задачи, поля для введения ответа и кнопкой для проверки введенного ответа

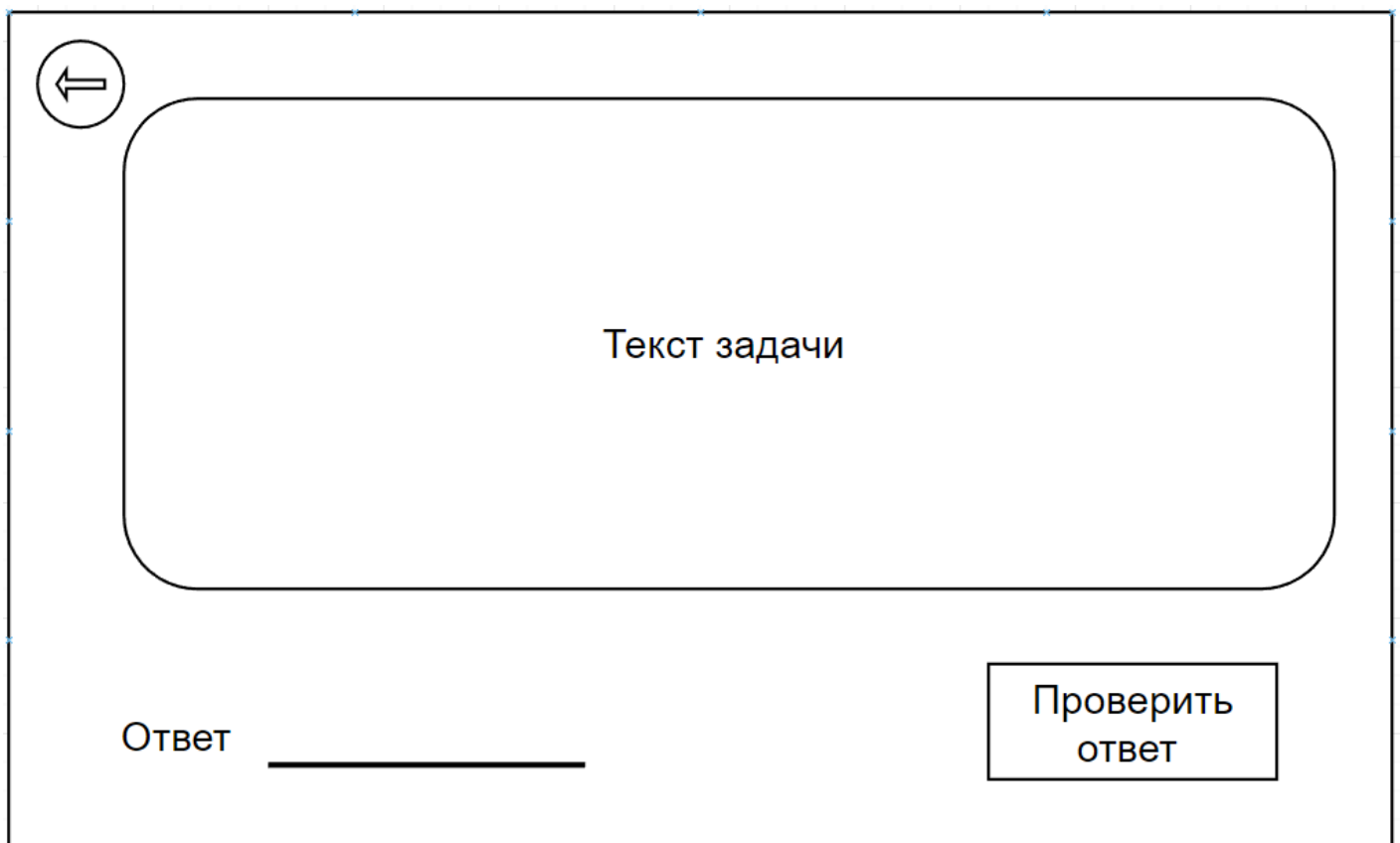
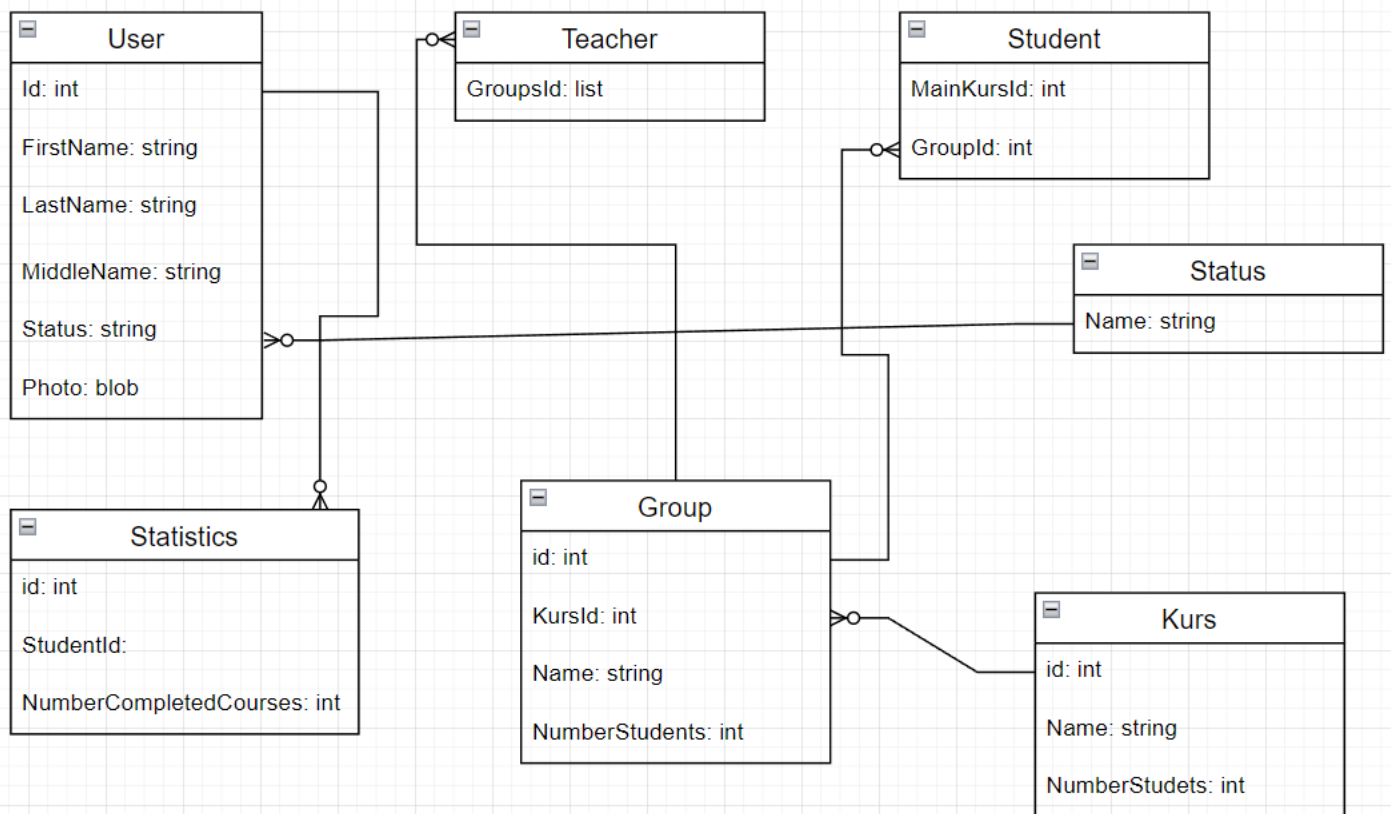


Диаграмма сущностей:



Разработка API системы:

1. LoadData

Функция предназначена для считывания информации из базы данных и представление её в виде списка объектов. Данная функция использует Entity Framework для сопоставления информации о сущности с объектами.

Входная информация: объект String содержащий в себе информацию для подключения к базе данных.

Выходная информация: список объектов List<T>.

2. SyncData

Функция предназначена для синхронизации информации из локальной базы данных программы с базой данных, хранимой на сервере. Осуществляется данный процесс с помощью репликации базы данных.

Входная информация: массив String содержащий в себе информацию для подключения к базе данных и информацию о выбираемых таблицах для репликации.

Выходная информация: enum-статус завершения операции.

3. UserAuthentication

Функция предназначена для аутентификации пользователя в приложении.

Входная информация: логин, пароль.

Выходная информация: enum-статус завершения операции.

4. UploadFile

Функция предназначена для выгрузки пользователем файла в локальную папку.

Входная информация: изображение со скриншота.

Выходная информация: png файл.

5. Rotate

Функция предназначена для поворота камеры пользователя.

Входная информация: нажатие клавиши мыши.

Выходная информация: вектор поворота.

6. Move

Функция предназначена для расчета
передвижения камеры

Входная информация: положение камеры в 3-х мерном пространстве

Выходная информация: позиция камеры с новыми координатами

7. ButtonActivator

Функция предназначена для активации кнопок нажатием на нее мышью.

Входная информация: номер элемента массива кнопок.

Выходная информация: true или false.

8 DrawCircle

Функция предназначена для рисования на доске кистью

Входные параметры: координаты x,y

Выходная информация: закрашивание определенного пикселя.

9 IEnumeratorText

Функция предназначена для последовательного вывода текста

Входная информация: имя файла.

Выходная информация: массив строк.

Иерархическая структура работ:

1. Разработка технического задания
 - 1.1. Сбор требований;
 - 1.2. Определение стадий и этапов разработки
 - 1.2.1. Определение стадий разработки;
 - 1.2.2. Определение сроков разработки;
 - 1.3. Общее описание
 - 1.3.1. Назначение продукта;
 - 1.3.2. Взаимодействие продукта;
 - 1.3.3. Допущения и ограничения продукта;
 - 1.3.4. Определение функций продукта;
2. Разработка приложения
 - 2.1. Backend-разработка
 - 2.1.1. Проектирование базы данных

- 2.1.1.1. Определение структуры базы данных;
- 2.1.1.2. Определение связей между сущностями;
- 2.1.1.3. Определение взаимодействия с базой данных;

2.1.2. Разработка API приложения

- 2.1.2.1. LoadData;
- 2.1.2.2. SyncData;
- 2.1.2.3. UserAuthentication;
- 2.1.2.4. UpLoadFile;
- 2.1.2.5. Rotate;
- 2.1.2.6. Move;
- 2.1.2.7. DrawCircle;
- 2.1.2.8. ButtonActivator;
- 2.1.2.9. IEnumeratorText.

2.1.3. Сетевое взаимодействие

- 2.1.3.1. Определение протокола взаимодействия;
- 2.1.3.2. Обеспечение защищённости соединения;
- 2.1.3.3. Определение местоположения хранения данных;

2.1.4. Взаимодействие с UI

- 2.1.4.1. Определение модели привязки данных;
- 2.1.4.2. Создание механизма обновления данных;

2.2. Frontend-разработка

2.2.1. Дизайн

2.2.1.1. Разработка макетов страниц

- 2.2.1.1.1. Разработка макета главного экрана;
- 2.2.1.1.2. Разработка макета меню настроек;
- 2.2.1.1.3. Разработка макета создания курса;
- 2.2.1.1.4. Разработка макета выбора курса;
- 2.2.1.1.5. Разработка макета окна со справочным материалом;
- 2.2.1.1.6. Разработка макета настроек рисования;
- 2.2.1.1.7. Разработка макета окна задачи;

2.2.1.2. Взаимодействие с backend

- 2.2.1.2.1. Определение модели привязки данных;
- 2.2.1.2.2. Настройка механизма обновления данных;
- 2.2.1.2.3. Определение динамического взаимодействия с данными;

2.2.1.3. Разработка общего стиль-кода приложения

- 2.2.1.3.1. Создание иконок элементов;
- 2.2.1.3.2. Определение стиля и размера шрифтов;

3. Приемо-сдаточные испытания

3.1. Подготовка и проведение демонстрации;

- 3.2. Проведение испытаний;
- 4. Размещение приложения
 - 4.1. Аренда сервера;
 - 4.2. Развёртывание приложения;
 - 4.3. Размещение в магазине приложений;
- 5. Поддержка приложения
 - 5.1. Мониторинг работоспособности;
 - 5.2. Получение и обработка обратной связи;
 - 5.3. Улучшение работы приложения
 - 5.4. Добавление новой функциональности;

Оценка времени выполнения проекта по методу PERT:

Работы	Количество	Оптимистичные затраты	Пессимистичные затраты	Наиболее вероятные затраты
Создание сущностей	7	2	6	3
Создание макетов	7	20	60	40
Создание методов API	9	10	70	20

Средние трудозатраты:

$$\text{Создание сущностей} = \frac{6 + 4 \times 3 + 2}{6} = 3.3 \text{ чел.* час.}$$

$$\text{Создание макетов} = \frac{60 + 4 \times 40 + 20}{6} = 40 \text{ чел.* час.}$$

$$\text{Создание методов API} = \frac{70 + 4 \times 20 + 10}{6} = 26,6 \text{ чел.* час.}$$

Среднеквадратичная оценка:

$$\text{Создание сущностей} = \frac{6 - 2}{6} = 0,6 \text{ чел.* час.}$$

$$\text{Создание макетов} = \frac{60 - 20}{6} = 6,6 \text{ чел.* час.}$$

$$\text{Создание методов API} = \frac{70 - 10}{6} = 10 \text{ чел.* час.}$$

$E_{\text{общ.}}$:

$$E_{\text{общ.}} = 7 * 3.3 + 7 * 40 + 9 * 26,6 = 542.5 \text{ чел.* час.}$$

СКО_{общ.}:

$$СКО_{\text{общ.}} = \sqrt{7 \times 0.6^2 + 7 \times 6.6^2 + 9 \times 10^2} = 34.7 \text{ чел. * час}$$

Оценка суммарной трудоёмкости проекта с вероятностью 95%:

$$E_{95\%} = 542.5 + 2 * 34.7 = 577.2 \text{ чел.* час.}$$

Базовое расписание в виде диаграммы Ганта:

Представлено в файле – «Базовое расписание.pdf»