

## Разработка базового расписания

### 1) Прототипы экранной форм

- Стартовое меню для начала работы в лаборатории

Главное окно, которое отображается при входе в приложение, служит стартовым экраном откуда можно начать саму лабораторную работу для студента (кнопка «Студент»), войти, как преподаватель для просмотра результатов выполнения работы (кнопка «Преподаватель») и выйти из приложения (кнопка «Выход»).

Прототип стартового меню приложения. Экран содержит три кнопки, расположенные вертикально в центре:

- Кнопка «Преподаватель» (с красным текстом)
- Кнопка «Студент»
- Кнопка «Выход» (с синим текстом)

- Меню для заполнения данных

Служит для регистрации студента, нужно ввести ФИО для того, чтобы начать лабораторную работу.

Введите ФИО

Зарегистрироваться

- Меня для выбора варианта

Нужно для того, чтобы студент мог выбрать свой вариант, который согласовывается перед этим с преподавателем.

1

2

3

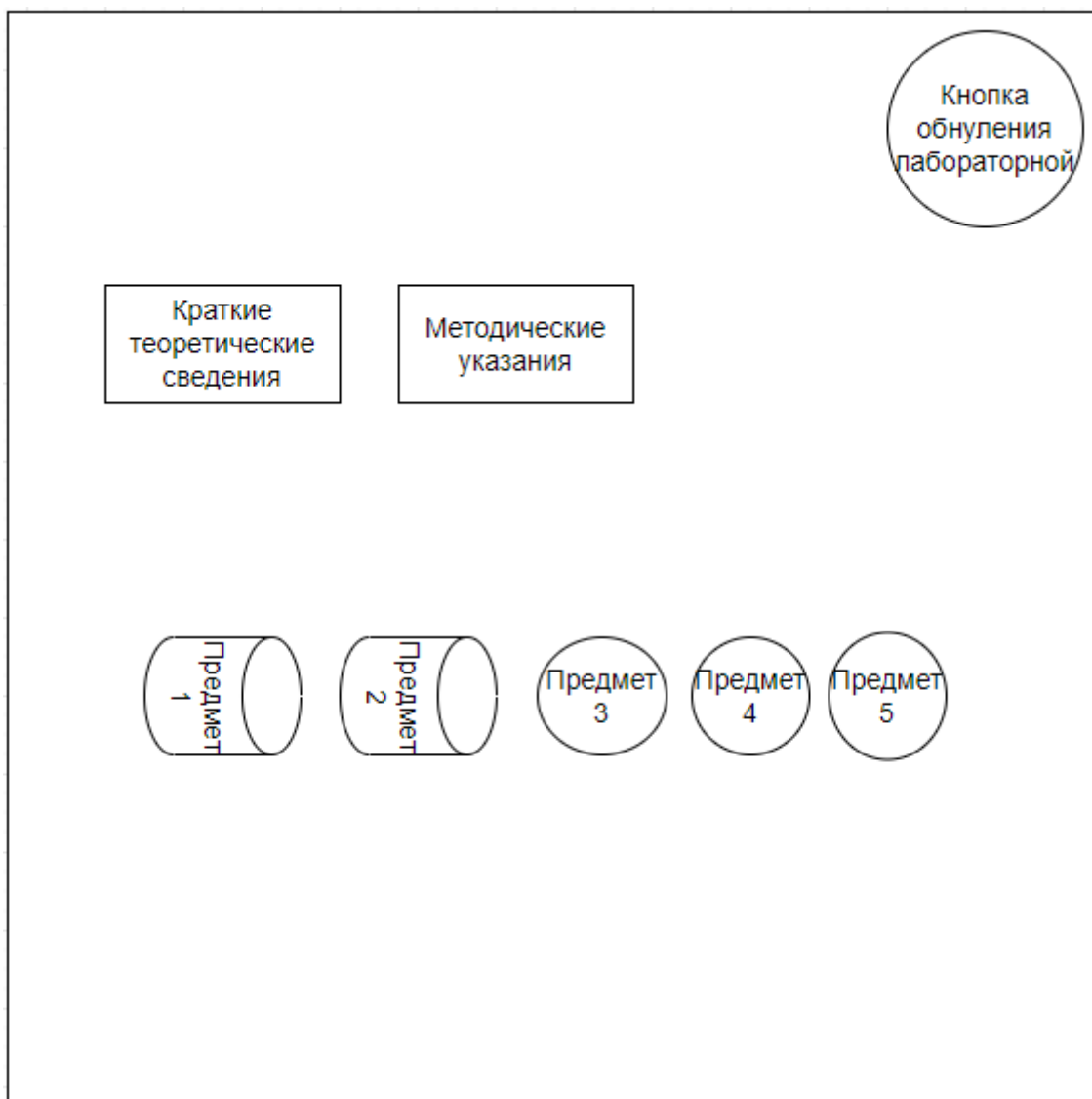
- Прототип экрана выполнения лабораторной работы

Сцена, которая запускается из меню выбора варианта, содержит интерактивное поле, на котором происходит весь основной этап выполнения работы, кнопки возвращения для предоставления кратких теоретических сведений, кнопка для вызова методических указаний, появление виртуального ассистента, который инструктирует студента по выполнению лабораторной, в этом разделе находятся кнопки: «Меню» для выхода в основное меню приложения, «Далее» для перехода к следующему шагу, «Назад» для перехода на шаг назад и «Пропустить» для того, чтобы можно было сразу перейти к выполнению. Так же есть кнопка обнуления выполнения лабораторной работы, при нажатии на нее мы начинаем все с начала.

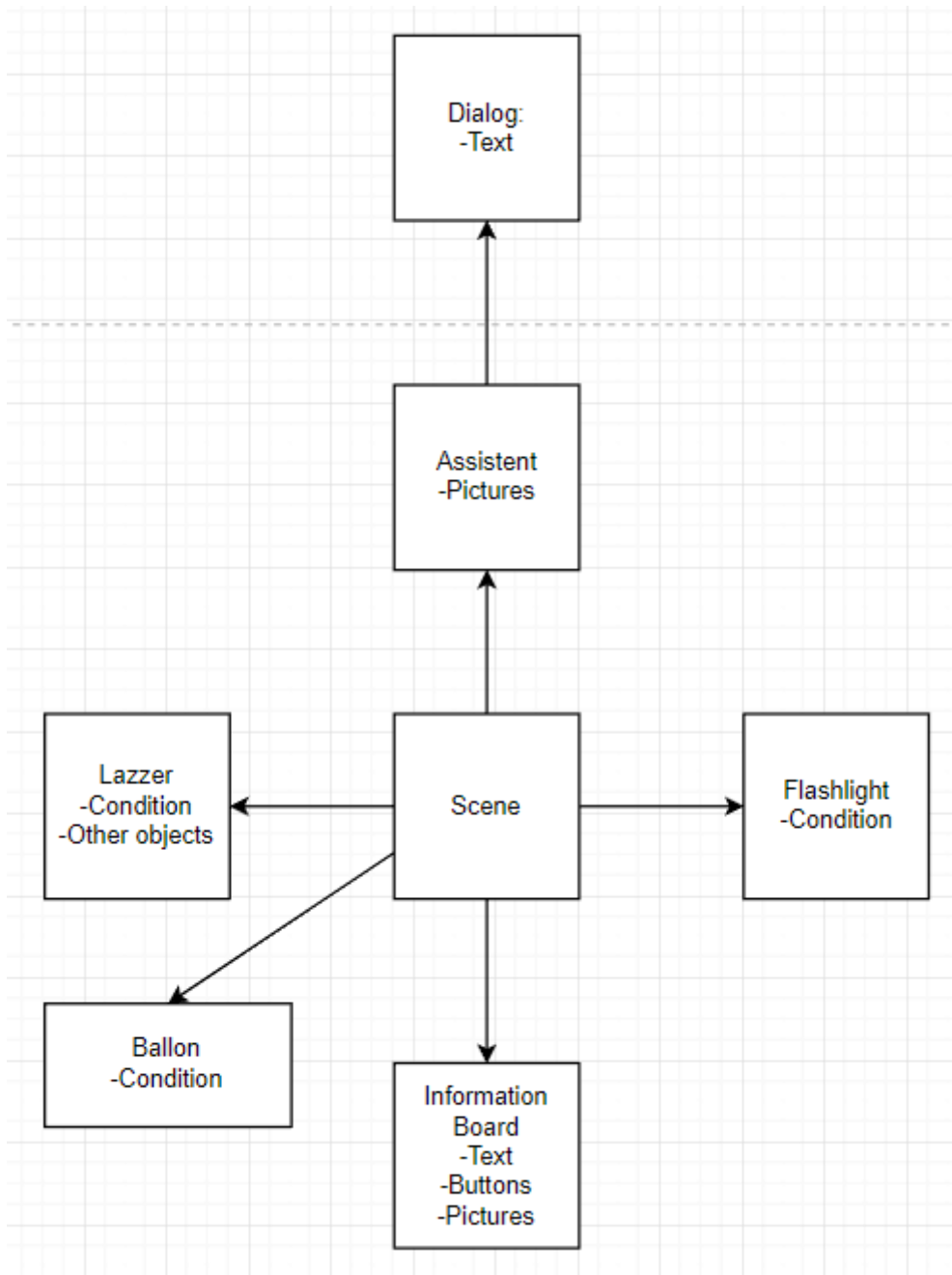


После того, как мы прошли инструктаж от виртуального помощника можно переходить к выполнению лабораторной работы. Для этого на столе предоставлены предметы и материалы, необходимые для правильной

реализации поставленной задачи. Предметы можно перетаскивать и соединять в единую конструкцию. Так же в любой момент можно обратиться к кратким теоретическим сведениям и методическим указаниям.



## 2) Диаграмма сущностей



## 3) Разработка api системы

- **DialogManager**

Функция предназначена для настройки ассистента – ознакомления с рабочим местом, вводной информации по работе и управлению.

Входные данные: текст ассистента

Выходные данные: определенному объекту изменяется булевый компонент SetActive.

- **Lazzer**

Функция, отвечающая за настройку лазера и обработку все возможных вариантов сборки.

Входные данные: координаты положения объектов.

Выходные данные: определенному объекту изменяется булевый компонент SetActive.

- **GoToMenu**

Функция возврата в главное меню.

Входные данные: активация элемента типа “button”.

Выходные данные: переключение сцены.

- **GoToScene**

Функция запуска лабораторной работы. Активирует сцену “New scene”.

Входные данные: ввод с клавиатуры.

Выходные данные: переключение сцены.

- **NewAnimFunc**

Функция, отвечающая за анимацию передвижения объектов.

Входные данные: координаты

Выходные данные: координаты

- **NewDragNDrop**

Функция, которая отвечает за свойство перемещения игроком игровых объектов. Она обрабатывает конкретный объект, который был взят игроком и при помощи встроенных ивентов Unity, присваивает ему свойства и отключает их (например: при взятии отключение своевольного перемещения, отключение физики, при отпуске объекта эти свойства обратно включаются)

Входные данные: экземпляр объекта типа GameObject, его компонент Rigidbody, классификация объекта (пользовательский тип данных TypeItem).

Выходные данные: выбранному объекту типа GameObject передаются новые значения их свойств.

- **Speak**

Функция, активирующая звуковое сопровождение.

Входные данные: ввод с клавиатуры, координаты

Выходные данные: звуковая дорожка

#### 4) 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. Разработка API приложения
    - 2.1.1. DialogManager;
    - 2.1.2. Fonarik;
    - 2.1.3. GoToMenu;
    - 2.1.4. GoToScene;
    - 2.1.5. Lazzer;
    - 2.1.6. MainMenu;
    - 2.1.7. NewAnimFunc;
    - 2.1.8. NewBehaviourScript;
    - 2.1.9. NewDragNDrop;
    - 2.1.10. ShowText;
    - 2.1.11. SimpleCameraController;
    - 2.1.12. Speak;
    - 2.1.13. TriggerPlace;
    - 2.1.14. UIController;
  - 2.2. Frontend-разработка
    - 2.2.1. Дизайн
      - 2.2.1.1. Разработка макетов окон
        - 2.2.1.1.1. Разработка макета главного экрана;
        - 2.2.1.1.2. Разработка макета меню заполнения данных;
        - 2.2.1.1.3. Разработка макета меню для выбора варианта;
        - 2.2.1.1.4. Разработка макета экрана выполнения лабораторной работы;
      - 2.2.1.2. Создание предметов
        - 2.2.1.2.1. Создание оптических линз;
        - 2.2.1.2.2. Создание стержней разной длины;
        - 2.2.1.2.3. Создание шарика;
        - 2.2.1.2.4. Создание корпусов для лазера;
        - 2.2.1.2.5. Создание фонарика;
        - 2.2.1.2.6. Создание лампы накаливания;

- 2.2.1.3. Создание виртуального окружения
  - 2.2.1.3.1.Создание стен;
  - 2.2.1.3.2.Создание доски с кнопками для перехода к кратким теоретическим сведениям и методическим указаниям;
  - 2.2.1.3.3.Создание доски с краткими теоретическими сведениями;
- 2.2.1.4. Взаимодействие с backend
  - 2.2.1.4.1.Определение модели привязки данных
  - 2.2.1.4.2.Настройка механизма обновления данных
  - 2.2.1.4.3.Определение динамического взаимодействия с данными;
- 2.2.1.5. Разработка общего стиль-кода приложения
  - 2.2.1.5.1.Создание логотипа приложения;
  - 2.2.1.5.2.Создание иконок элементов;
  - 2.2.1.5.3.Определение стиля и размера шрифтов;
- 3. Приемо-сдаточные испытания
  - 3.1. Подготовка и проведение демонстрации;
  - 3.2. Проведение испытаний;
- 4. Поддержка приложения
  - 4.1. Мониторинг работоспособности;
  - 4.2. Получение и обработка обратной связи;
  - 4.3. Улучшение работы приложения;
  - 4.4. Добавление новых функций;

##### 5) Оценка времени выполнения проекта по методу PERT

Работы	Количество	Оптимистичные трудозатраты за ед. в часах	Пессимистичные трудозатраты за ед. в часах	Наиболее вероятные трудозатраты за ед. в часах
Создание макетов	4	8.25	12	9.5
Создание методов API	21	1.5	4.7	3.1
Создание предметов	9	3.2	6.5	4.8



Посчитаем средние трудозатраты по каждой работе:

$$\text{Создание макетов} = \frac{33 + 48 + 38 \times 4}{6} = 38,8 \text{ чел.* час.}$$

$$\text{Создание методов API} = \frac{31 + 99 + 65 \times 4}{6} = 65 \text{ чел.* час.}$$

$$\text{Создание предметов} = \frac{29 + 59 + 43 \times 4}{6} = 43,3 \text{ чел.* час.}$$

Посчитаем среднеквадратичную оценку:

$$\text{Создание макетов} = \frac{48 - 33}{6} = 2,5 \text{ чел.* час.}$$

$$\text{Создание методов API} = \frac{99 - 31}{6} = 11,3 \text{ чел.* час.}$$

$$\text{Создание предметов} = \frac{59 - 29}{6} = 5 \text{ чел.* час.}$$

Посчитаем  $E_{\text{общ.}}$ :

$$E_{\text{общ.}} = 4 * 38,8 + 21 * 65 + 43,3 * 9 = 1909,9 \text{ чел.* час.}$$

Посчитаем  $СКО_{\text{общ.}}$ :

$$СКО_{\text{общ.}} = \sqrt{4 \times 2,5^2 + 21 \times 11,3^2 + 9 \times 5^2} = 54,1 \text{ чел.* час.}$$

Оценка суммарной трудоёмкости проекта с вероятностью 95%:

$$E_{95\%} = 1909,9 + 2 * 54,1 = 2018,1 \text{ чел.* час.}$$

6) Базовое расписание в виде диаграммы Ганта.

