**The Circle World  
 (Круглый Мир)**

**Технический дизайн-документ**

СОДЕРЖАНИЕ

[1 План разработки 3](#_Toc455216452)

[1.1 Подготовка персонажа 3](#_Toc455216453)

[1.2 Подготовка основных элементов 3](#_Toc455216454)

[1.3 Создание первого уровня 3](#_Toc455216455)

[2 Структура проекта 4](#_Toc455216456)

[3 Персонаж 5](#_Toc455216457)

[3.1 Стандартный геймплей 5](#_Toc455216458)

[3.1.1 Содержимое модели: 5](#_Toc455216459)

[3.1.2 Составляющие объекта: 5](#_Toc455216460)

[3.1.3 Основные анимации объекта: 5](#_Toc455216461)

[3.1.4 Управление персонажем: 6](#_Toc455216462)

[4 Среда 7](#_Toc455216463)

[4.1 Основные элементы 7](#_Toc455216464)

[4.2 SkyBox 7](#_Toc455216465)

[4.3 Fog 7](#_Toc455216466)

[5 Камера 7](#_Toc455216467)

[6 Препятствия 7](#_Toc455216468)

[7 Смерти и Respawn-ы 7](#_Toc455216469)

[8 GUI 7](#_Toc455216470)

[9 Сборка под Android 7](#_Toc455216471)

# План разработки

Во время разработки проекта требуется придерживаться примерно следующего плана:

## Подготовка персонажа

1. Создание фигурки персонажа
2. Простейшее движение персонажа
3. Анимации персонажа
4. Полное движение персонажа

## Подготовка основных элементов

1. Камеры и скрипт движения за персонажем
2. Создание базовых элементов – Ground, GiveDamage
3. SkyBox
4. Respawn
5. Subtitler

## Создание первого уровня

1. Полоса с базовыми препятствиями
2. Настройка камеры и запуск уровня
3. Набросок сцены и стартовой кат-сцены
4. Тестовый билд под android
5. Полное планирование уровня
6. Текстурирование всех препятствий
7. Расставление всех препятствий
8. Тексурирование базовых Кругляшей и Ведущего
9. Расставление всех визуальных составляющих
10. Реализация кат-сцены
11. Полный билд под андроид

# Структура проекта

* Prefabs
  + Player – все вариации Колобка
  + Characters – все персонажи использующиеся в кат-сценах и обстановке
  + Dangers – все опасности
  + Environment – земля, стены и т.д.
* External – содержит все сторонние пакеты, использующиеся в проекте
  + QuickCutsceneCreator
  + ClassicSkybox
  + StandartAssets
* Scripts:
  + CharacterControl
  + CameraToPlayer
  + Enemies
    - GiveDamageToPlayer
  + Managers
    - LevelManager
    - GameManager
    - InputManager
    - MobileInput
    - PcInput
* Animations – здесь абсолютно для каждого персонажа создается отдельная папка, в которую помещаются, как аниматор, так и все анимации.
* Textures
  + Characters
  + Environment
* Materials

# Персонаж

В игре присутствует 3 типа геймплея, для каждого используется своя модель персонажа.

## Стандартный геймплей

### Содержимое модели:

* Body (сфера с цветным шейдером без текстуры)
  + Face (сфера чуть большего радиуса, чем Body с текстурой лица, шейдер – LegacyShaders/Transparent/Cutout/Soft
* Left Hand и другие конечности – эллипсоиды с текстурой конечности, расположенные в необходимых местах.

### Составляющие объекта:

* Box collider – занимает все тело и конечности
* Rigit Body
* PlayerControllerMain
* Animator

### Основные анимации объекта:

* Step – объект в движении – анимация шага – в геймплее игрок всегда находится в движении, поэтому это основное положение, анимация представлена движением конечностей и тела
* Down – объект в пригнутом состоянии, катится, как шар, анимация статична – конечности убраны, тело прижато к полу, коллайдер сжат. Прокрутка тела осуществляется скриптом, при этом при возвращении из этого положения в другое – скрипт докручивает тело до стандартного положения.
* Jump – объект в прыжке, анимация статична – конечности выставлены, как при прыжке.
* Dead – объект отлетает назад и падает на спину, непосредственно отлет назад регулируется скриптом.
* Дополнительные положения используются только в кат-сценах и рассматриваются для них

Четыре анимации являются связанными полностью с помощью триггеров, доп. анимации связываются по необходимости

### Управление персонажем:

* Скрипт - PlayerControllerMain
* Персонаж всегда сам двигается вперед – в направлении оси Y, скорость выносится в инспектор
* Существует два способа движения по бокам, выбираются в инспекторе
  + Непрерывное движение – вводится скорость, управляется акселлерометром
  + Дискретное движение – управляется свайпами – происходит переход на соседнюю дорожку, в инспектор выносится скорость перехода и количество дорожек.
* Присест осуществляется свайпом вниз, при этом начинается прокрутка тела, запускается анимация, но параметры движения не меняются
* Прыжок осуществляется свайпом вверх, происходит по параболе, сила импульса – в инспекторе.
* Прыжок и присест возможны друг из друга, если осуществляется присест из прыжка, телу сообщается импульс вниз, если наоборот, то происходит обычный прыжок.
* Имеется метод Kill, который проигрывает анимацию смерти и запускает Respawn.

# Среда

## Основные элементы

## SkyBox

## Fog

# Камера

Для перевода камеры в новый ракурс во время геймплея используются кат-сцены. После проигрывания камера остается в конечном положении и можно продолжать геймплей так.

# Препятствия

# Смерти и Respawn-ы

# GUI

## Главное меню

Обычное меню реализуется с помощью внешнего пакета PauseController.

## Пауза

Пауза реализуется с помощью внешнего пакета PauseController.

# Сборка под Android

## Общие данные

Игра по-умолчанию предназначена для Android

Функционирует в ландшафтной ориентации

## Управление

Управление осуществляется с помощью свайпов и акселлерометра.