**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**"ВЫСШАЯ ШКОЛА ЭКОНОМИКИ"**

Факультет компьютерных наук

Департамент программной инженерии

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| СОГЛАСОВАНО  Научный сотрудник МЛ ИССА факультета компьютерных наук Национального исследовательского университета «Высшая школа экономики», канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_О.В. Максименкова  "\_\_\_" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г. | |  | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  проф, канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  "\_\_\_" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. | |
| |  |  | | --- | --- | | Подп. и дата |  | | Инв. № дубл. |  | | Взам. Инв. № |  | | Подп. и дата |  | | Инв. № подл. |  | | **МОБИЛЬНАЯ 2D КОСМИЧЕСКАЯ АРКАДА  Пояснительная записка**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.04.01-01 81 01-1-ЛУ**  Исполнитель  Студент группы БПИ191  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Н.О. Казанцев /  "\_\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. | | | | |
|  | |  | | |

**УТВЕРЖДЕН**

**RU.17701729.04.01-01 81 01-1-ЛУ**

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл |  |

**МОБИЛЬНАЯ 2D КОСМИЧЕСКАЯ АРКАДА  
  
Пояснительная записка  
  
RU.17701729.04.01-01 81 01-1-ЛУ**

**Листов 40**

СОДЕРЖАНИЕ

[1. ВВЕДЕНИЕ 3](#_Toc40460816)

[1.1. Наименование программы 3](#_Toc40460817)

[1.2. Документы, на основании которых ведется разработка 3](#_Toc40460818)

[2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ 4](#_Toc40460819)

[2.1. Назначение программы 4](#_Toc40460820)

[2.1.1. Функциональное назначение 4](#_Toc40460821)

[2.1.2. Эксплуатационное назначение 4](#_Toc40460822)

[2.2. Краткая характеристика и области применения 4](#_Toc40460823)

[3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ 5](#_Toc40460824)

[3.1. Постановка задачи на разработку программы 5](#_Toc40460825)

[3.2. Описание алгоритма и функцинирования программы 5](#_Toc40460826)

[3.2.1. Описание алгоритма взаимодействия экранов программы 5](#_Toc40460827)

[3.2.2. Описание функционирования меню игры 5](#_Toc40460828)

[3.2.3. Описание игрового процесса 6](#_Toc40460829)

[3.2.4. Описание функционирования космического объекта, гравитационных взаимодейстий 7](#_Toc40460830)

[3.2.5. Описание функционирования выхода на орбиту меньшего тела 9](#_Toc40460831)

[3.2.6. Описание функционирования заднего игрового фона и алгоритма смещения текстуры 9](#_Toc40460832)

[3.3. Описание и обоснование выбора метода организации входных и выходных данных. 10](#_Toc40460833)

[3.4. Описание и обоснование выбора состава технических и программных средств 11](#_Toc40460834)

[3.4.1. Состав технических и программных средств 11](#_Toc40460835)

[3.4.2. Обоснование выбора технических и программных средств. 11](#_Toc40460836)

[4. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 13](#_Toc40460837)

[4.1. Предполагаемая потребность 13](#_Toc40460838)

[4.2. Ориентировочная экономическая эффективность 13](#_Toc40460839)

[4.3. Преимущества разработки по сравнению с отечественными или зарубежными аналогами 13](#_Toc40460840)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 14](#_Toc40460841)

[ПРИЛОЖЕНИЕ 1 15](#_Toc40460842)

[ПРИЛОЖЕНИЕ 2 16](#_Toc40460843)

[ПРИЛОЖЕНИЕ 3 18](#_Toc40460844)

1. ВВЕДЕНИЕ
   1. Наименование программы

Наименование программы «Мобильная 2D космическая аркада».

Наименование программы на английском языке – «Mobile 2D Space Arcade».

Наименование программы для пользователя – «Accretion».

* 1. Документы, на основании которых ведется разработка

Приказ декана факультета компьютерных наук И.В. Аржанцева "Об утверждении тем, руководителей курсовых работ студентов образовательной программы «Программная инженерия» факультета компьютерных наук" № 2.3-02/1112-04 от 11.12.2019

1. **НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ**
   1. Назначение программы
      1. Функциональное назначение

Приложение является игрой и несёт в себе развлекательную и образовательную функции. Игра является аркадой, вследствие чего главная задача - предоставить пользователю бесконечный геймплей, быстрое обучение, простой игровой процесс – всё, что характеризует этот жанр.

* + 1. Эксплуатационное назначение

Данная программа может использоваться для проведения свободного времени людьми разных возрастных категорий, будет полезна в учебных целях, как доступный и увлекательный способ объяснить человеку устройство космического пространства.

* 1. Краткая характеристика и области применения

“Мобильная 2D космическая аркада” предназначена для развлечения и возможности приятно провести время людьми разного возраста, а также способная в простой форме объяснить человеку как работает гравитация, рассказать о планетах, звездах и черных дырах, заинтересовать космической тематикой.

1. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ
   1. Постановка задачи на разработку программы

Разрабатываемое приложение должно соответствовать:

- описанным в техническом задании функциональным характеристикам («Мобильная 2D космическая аркада». Техническое задание);

- описанным в техническом задании требованиям к интерфейсу («Мобильная 2D космическая аркада». Техническое задание);

- описанным в техническом задании требованиям к надежности («Мобильная 2D космическая аркада». Техническое задание).

* 1. Описание алгоритма и функцинирования программы
     1. Описание алгоритма взаимодействия экранов программы

Для создания удобной навигации в игре, был создан экран главного меню, содержащий несколько разделов. Схема взаимодействия приведена на рис. 1

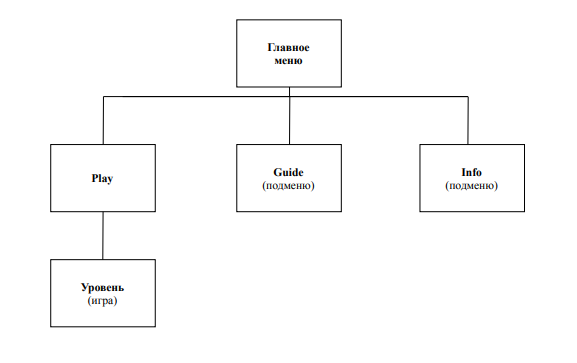


Рисунок 1 Взаимодействие экранов программы

* + 1. Описание функционирования меню игры

При запуске игры открывается главное меню (рис. 2) В нем пользователь видит следующие пункты:

1) Кнопку начала игры (при нажатии начнется загрузка основной сцены с игрой)

2) Обучение (в данном разделе можно узнать основные принципы игровых действий)

3) Информация об игре (здесь пользователь может узнать мотивацию автора при создании приложения, а также увидеть какие дополнения и пакеты использовались при разработке)



Рисунок 2 Главное меню

Если игрок уничтожается во время игры, то пользователю выводится соответствующее сообщение о произошедшем и предлагается на выбор один из пунктов: переиграть или выйти в главное меню (рис. 3)



Рисунок 3 Меню при смерти игрока

* + 1. Описание игрового процесса

Чтобы пройти игру игрок должен набрать достаточное количество игровой массы (характеристика тела, определяющая гравитационные свойства тела), чтобы ее хватило для трансформации в черную дыру, а после достигнуть некого предела. Так игра будет полностью пройдена

Во время игры пользователь должен набирать массу. Ее получение может происходить нескольми путями: мгновенным поглощением, в случае если другой объект слишком мал относительно игрока, или выводом на орбиту с последущим поглощением, если иное космическое тело незначительно меньше нашего.

Однако, игрок может потерять массу, ему будут мешать в этом другие космические тела. Такое может произойти, если случилось столкновение с большим по массе объектом. Тогда же вероятен и откат в прогрессе, т.е. обратная трансформация в меньшее космическое тело.

Чтобы воссоздать атмосферу космоса, вокруг пользователя генирируются случайные космические тела: астероиды, планеты (карликовые и обычные), малые звезды, звезды-гиганты, а также черные дыры.

* + 1. Описание функционирования космического объекта, гравитационных взаимодейстий

Космический объект – физическое тело с RigidBody2D компонентом (в нем хранится масса, коэффициент натяжения, угол, при котором на объект действует гравитация и т.п.)[8], два коллайдера – один отвечает за столкновения, а второй, имеющий радиус больший, чем радиус сферы объекта, отвечает за обработку объектов, влетающих в гравиационное поле тела (рис. 5).

При контакте двух объектов, рассчитывается вектор для каждого из них, но вот сила притяжения определяется относительно масс объектов по алгоритму, продемонстрированному на схеме ниже (рис.4). Поэтому больший по массе объект притягивает сильнее меньшего, имеет большую силу притяжения.

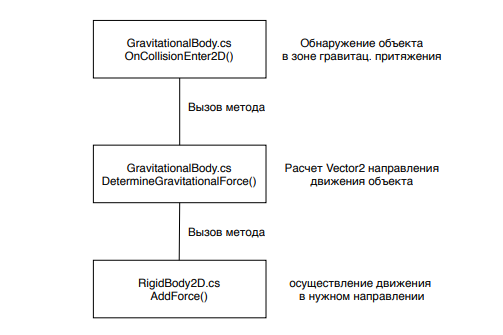


Рисунок 4 Схема расчета притяжений космических тел

Mesh Renderer [9] отвечает за визуальное отображение тела – астероид, планета, звезда, черная дыра.

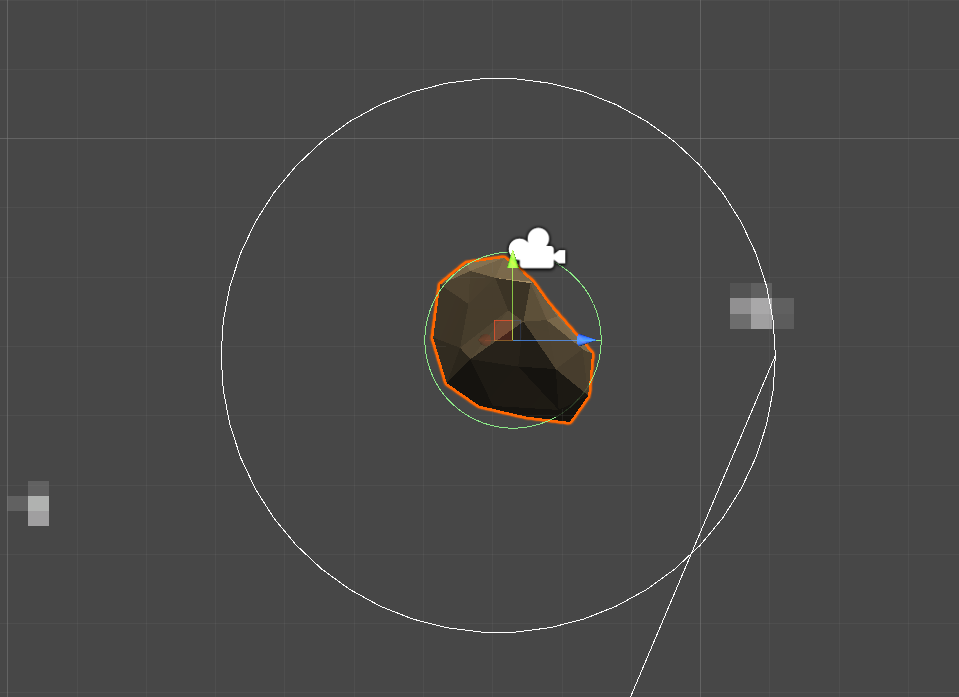


Рисунок 5 Внутренние и внешние коллайдеры

Если объект врезается в другой объект меньший по рангу на одну позицию (так, например, меньшим по рангу для планеты считается карликовая планета, для карликовой звезды - планета), то тело теряет массу, что в случае с игроком, отображается в панелях, показывающих очки массы и ранг игрока (рис. 6).

Если же космическое тело было слишком мало относительно другого (различие в рангах больше 1), то при столкновении меньший объект уничтожается, а оставшийся получает очки массы.

При равном ранге в случае столкновения оба объекта откатываются в прогрессе (при столкновении двух карликовых планет, обе превратятся в астероиды). При столкновении двух крупных объектов, после отката, они также чуть отталкиваются друг от друга.



Рисунок 6 Панель с информацией об игроке

* + 1. Описание функционирования выхода на орбиту меньшего тела

Если столкновение двух космических тел не произошло, а меньший объект находился в течение 5 секунд в зоне притяжения космического объекта, управляемого игроком, вращаясь вокруг него, то он автоматически выходит на орбиту. После этого тело получает команду стремиться совершать обороты вокруг более крупного объекта (рис. 7).

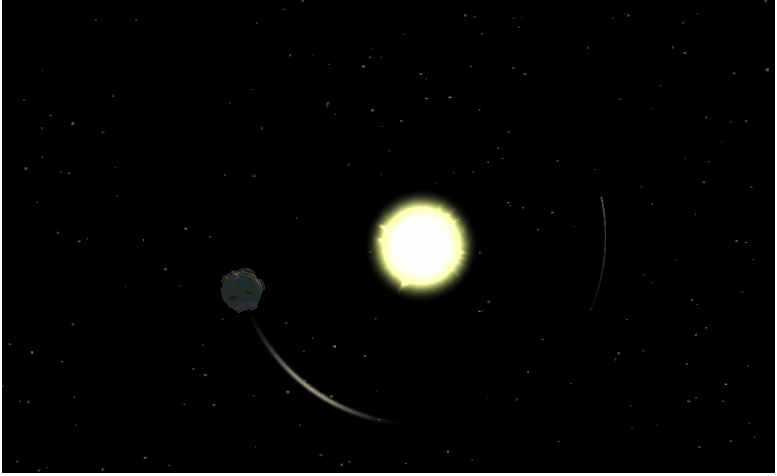


Рисунок 7 Планета на орбите у звезды

* + 1. Описание функционирования заднего игрового фона и алгоритма смещения текстуры

Игровой фон состоит из Quad’а[8] с наложенным на нем материалом. Если игрок перемещается в каком-то фиксированном направлении, то текстура материала смещается в обратном, тем самым получается бесконечная лента из изображения фона (рис. 8), которая создает эффект некончающегося игрового пространства, эффект движения у тела. [9]

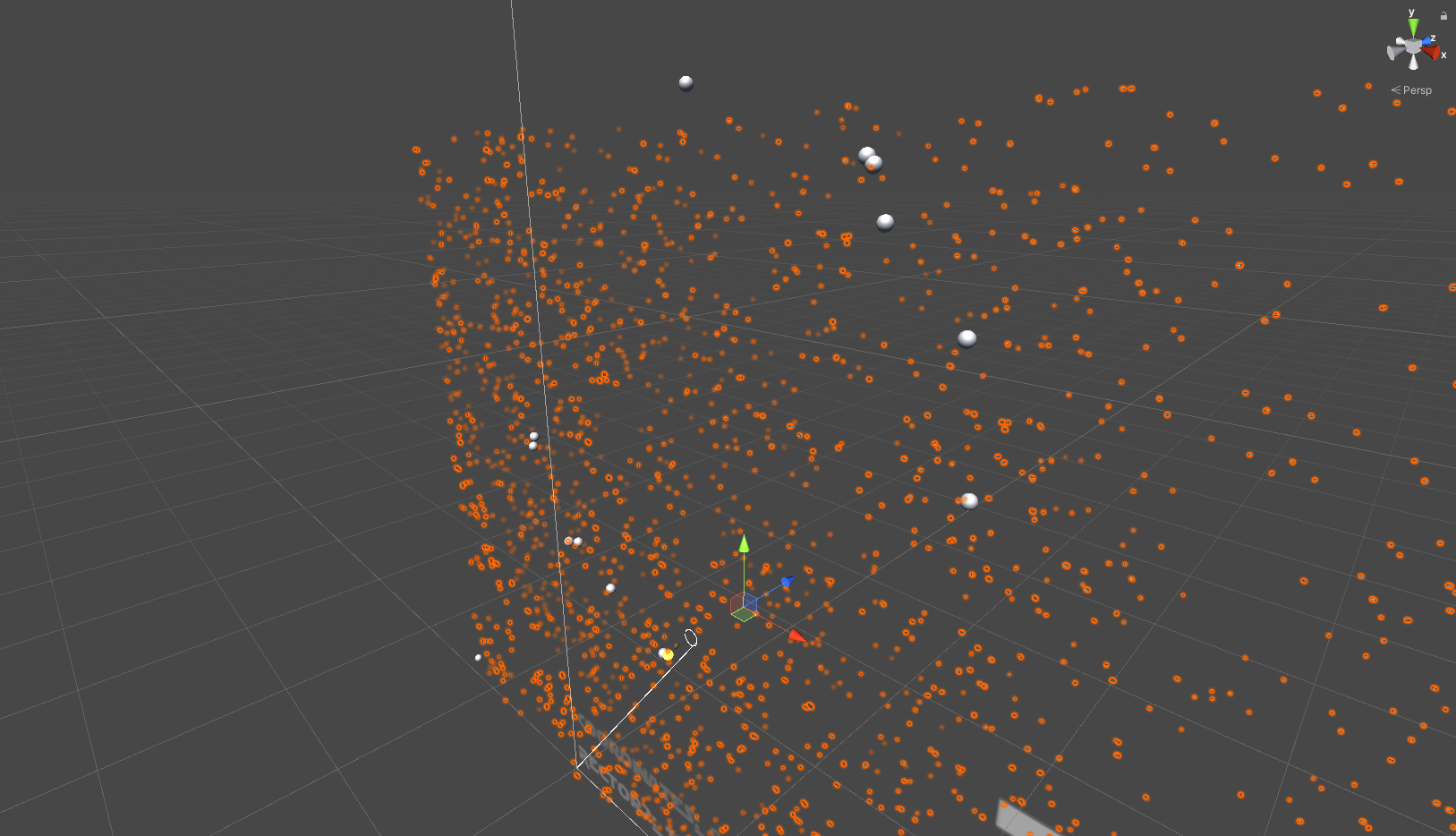


Рисунок 8 Материал, смещающийся от движения игрока

Алгоритм смещения текстуры:

1. Определяем вектор-направление игрока в данном кадр;
2. Определяем скорость движения космического тела игрока;
3. Смещаем текстуру на вектор, обратный вектору-направления, умноженному на скорость.
   1. Описание и обоснование выбора метода организации входных и выходных данных.

Входные данные в игре – действия игрока в приложении (нажатие на кнопки, передвижение космического тела джойстиком).

Выходные данные в программе – ответные отклики приложения на действия пользователя.

Механизм передвижения посредством джойстика был выбран в силу специфики платформы: наличие относительно малого, сенсорного экрана.

* 1. Описание и обоснование выбора состава технических и программных средств
     1. Состав технических и программных средств

Для корректной работы приложения необходим приведенный состав технических и программных средств.

Для компьтера:

1. операционная система Windows 7 (SP1+) или выше. Только 64-битные версия;
2. установленный Microsoft .NET Framework 4.0, который требует Windows Installer 3.1 или более позднюю версия;
3. Unity 2019.3.9f1 (64-bit) или выше;
4. Вычислительный процессор не ниже Intel Core i3 или подобный с тактовой частотой выше или равной 1.2 Ггц;
5. 512 ОЗУ или более;
6. Объем свободной памяти не менее 6 гб;
7. VGA-совместимый монитор и видеоадаптер;
8. Компьютерная мышь или тачпад.

Для смартфона:

1. Платформа Android 4.4 и выше.
2. Процессор с тактовой частотой не ниже 1 Ггц;
3. 300 мб оперативной памяти или более;
4. Объем свободной памяти не менее 90 мб;
   * 1. Обоснование выбора технических и программных средств.

При версии операционной системы младше прописанной, будет невозможно скомпилировать проект посредством Unity.

При версии Unity ниже будет также невозможно скомпилировать проект.

На смартфоне, который не соответствует минимальным параметрам, будет невозможно запустить проект в силу нехватки памяти, вычислительных мощностей или поддержки со стороны операционной системы.

Данный состав программных и технических средств обеспечит отображение объектов в хорошем качества, быструю обработку откликов на действия игрока, возможность работы без перебоев.

1. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ
   1. Предполагаемая потребность

“Мобильная 2D космическая аркада” - обучающая игра с моделированием гравитации и взаимодействия космических тел, будет востребована как способ развлечения, вариант отдыха, получения информации о вселенной. Игра может быть использована в образовательных учреждениях для демонстрации и объяснения основных концепций космоса.

* 1. Ориентировочная экономическая эффективность

В рамках данной работы расчёт экономической эффективности не предусмотрен.

* 1. Преимущества разработки по сравнению с отечественными или зарубежными аналогами

Основные преимущества данной игры:

1. Будет распространяться бесплатно;
2. Не будет требовать больших финансовых средств для разработки;
3. Будет иметь длительный срок службы;

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – ИПК Издательство стандартов, 2001.
2. ГОСТ 19.301-79 Программа и методика испытаний. //Единая система программной документации. – ИПК Издательство стандартов, 2001.
3. ГОСТ 19.401-78 Текст программы. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.505-79 Руководство оператора. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.404-79 Пояснительная записка. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.602-78 Правила дублирования, учета и хранения программных документов, выполненных печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. Англоязычная документация Unity [Электронный ресурс]// URL: https://docs.unity3d.com/Manual/index.html (Дата обращения: 11.04.2020, режим доступа: свободный).
9. Penny de Byl, Hoslistic Mobile Game Development with unity. 1 edition, Burlington: Routledge, 6(17) – 6(19), Nov 2014.

ПРИЛОЖЕНИЕ 1

**ТЕРМИНОЛОГИЯ**

Ниже приведен список используемых в тексте терминов, рекомендованных для ознакомления.

**Главное меню –** экран, показываемый человеку при начальном запуске игры.

**Игровое меню** – экран, который может открыть пользователь, находясь на игровой сцене.

**Разделы меню –** экраны, на которые пользователь может попасть из главного меню.

**Масса –** главнаяфизическаяхарактеристика космического тела. Благодаря ее увеличению, игрок движется в прогрессе на пути к полному прохождению игры.

**Скорость –** степень быстроты движения космического тела в пространстве.

**Астероид –** один из космических объектов, характеризующийся отсутствием притяжения, малой массой. Внешний вид – булыжник случайного цвета.

**Планета -** один из космических объектов, характеризующийся слабой силой притяжения, средней массой, способностью выводить объекты к себе на орбиту. Внешний вид – похожая на Землю сфера с различным цветовым оформлением.

**Звезда -** один из космических объектов, характеризующийся сильной степенью притяжения, огромной массой, способностью выводить объекты к себе на орбиту и поглощать планеты. Внешний вид – шар с анимацией термоядерного реакции, высвобождения энергии и сильного свечения.

**Черная дыра –** финальный космический объект, характеризующийся огромной силой притяжения, самой большой массой, способностью поглощать любые объекты. Внешний вид – черный шар с анимацией искажения, преломления света.

**Коллайдер –** компонент для обработки соприкосновений и взаимодействий объектов в программе Unity3D

**NPC –** неигровой персонаж, космический объект, генерирующийся вокруг игрока, обладающий гравитацией и некоторой массой.

ПРИЛОЖЕНИЕ 2

Таблица 1 Описание и функциональное назначение классов

|  |  |
| --- | --- |
| Классы | Назначение |
| ApplicationSet | Класс для регулирования настроек графики |
| AudioManager | Класс,ответственный за музыку в игре |
| CameraScript | Класс для плавного перемещения камеры вслед за объектом |
| Constants | Статический класс, хранящий в себе все игровые константы |
| Events | Контроллер всех событий в игре, их вызов |
| GameDeath | Контроллер конца игр, взаимодействие с уровнем |
| GravitationalBody | Контроллер любого космического объекта, хранение характеристик тела, просчет гравитационных взаимодействий, трансформаций в другие объекты. |
| LineRendererEx | Класс, ответственный за графическую отрисовку орбит, поля гравитационного притяжения |
| MenuCamera | Контроллер движения камеры в меню |
| MenuManager | Класс меню для запуска новой игры |
| Movement | Контроллер игрока. Обработка сенсорного и кнопочного управления |
| OrbitCatcher | Класс обнаружения тел в поле гравитации и проверки условий их выхода на орбиту |
| OrbitManager | Класс для графического отображения события выхода тела на орбиту |
| PrefabsStorage | Класс для работы с моделями астероидов, планет, звезд, черных дыр |
| ProgressBar | Класс для графического отображения прогресс-бара в интерфейсе |
| ScollBackground | Класс, ответственный за бесконечное прокручивание заднего фона игры, создания эффекта движения в космосе |
| Spawner | Класс, ответственный за псевдо-случайную генерацию космических объектов вокруг главного игрока |
| TextTyper | Класс, выводящий на экран сообщения |
| TrashCollector | Класс для работы с мусором. Удаление из памяти далеких космических объектов |
| UIController | Класс для контроля и измнения UI в игре |

ПРИЛОЖЕНИЕ 3

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ МЕТОДОВ И СВОЙСТВ**

Таблица 2 Описание методов, свойств класса ApplicationSet.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Методы | | | | |
| Имя | Модификатор  доступа | Тип | Аргументы | Назначение |
| Awake | private | void | - | Применяет настройки приложения |
| Свойства | | | | |
| Имя | Модификатор доступа | Тип | Доступ | Назначение |
| VSyncCount | private | int | set | Отключает или устанавливает Vsync настройку с некоторым параметром |
| TargetFrameRate | private | int | set | Выставляет максимальную частоту кадров для экрана |

Таблица 3 Описание методов и полей класса AudioManager.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| audioSource | private | void | Компонент, позволяющий работать с музыкой в Unity | |
| music | public | AudioClip | Звуковая дорожка для проигрывания | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Awake | private | void | - | Говорит приложению не прерывать скрипт при открытии новой сцены |
| PlayMusic | public | void | - | Включает музыку |
| StopMusic | public | void | - | Выключает музыку |

Таблица 4 Описание методов и полей класса CameraScript.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| FollowSpeed | public | float | Скорость слежения камеры за объектом | |
| Target | public | Transform | Объект слежения | |
| targetZPosition | public | int | Желаемая координата Z камеры в трехмерном пространстве | |
| Zoom1 | public | float | Начальный зум камеры | |
| Zoom2 | public | float | Желаемый зум камеры | |
| duration | public | float | Интервал плавного зума в секундах | |
| elapsed | private | float | 3 интервальная переменная для осуществления плавного зума | |
| transition | private | bool | Разрешено ли изменять зум | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Start | private | void | - | Устанавливает стандартную координату камеры при инициализации объекта |
| FixedUpdate | private | void | - | Перемещает вслед за объектом слежения. Осущесвляет плавный зум |
| SmoothChangeOrthographicSize | public | void | Float max | Осуществляет плавный зум |

Таблица 5 Описание полей класса Constants.cs

|  |  |  |  |
| --- | --- | --- | --- |
| Поля | | | |
| Имя | Модификатор  доступа | Тип | Назначение |
| vSyncCount | public | Int | Коэфициент технологии Vsync |
| targetFrameRate | public | Int | Максимальная частота обновления экрана в кадрах |
| GravityPower | public | Float | Сила гравитационного притяжения |
| HierarchyDict | public | Dictionary<string,int> | Иерархия значимости космических объектов |
| GrowCoefDict | public | Dictionary<string, int> | Коэфициент роста в зависимости от типа объекта |
| PlayerNextObject | public | String | Строковое представление следующего объекта при трансформации |
| AsteroidStartingMass | public | Float | Стартовая масса астероида |
| AsteroidCriticalMass | public | Int | Критическая масса астероида |
| AsteroidScale | public | Int | Коэфициент увеличения астероида |
| AsteroidMainCameraDistance | Public | Int | Близость камеры к астероиду |
| DwarfPlanetsStartingMass | Public | Int | Стартовая масса карликовой планеты |
| DwarfPlanetCriticalMass | Public | Int | Критическая масса карликовой планеты |
| DwarfPlanetScale | Public | Float | Коэфициент увеличения карликовой планеты |
| ColliderRadius\_DwarfPlanet | Public | Float | Радиус коллайдера карликовой планеты |
| PlanetMainCameraDistance | Public | int | Близость камеры к карликовой планеты |
| PlanetsStartingMass | Public | Int | Стартовая масса планеты |
| PlanetCriticalMass | Public | Int | Критическая масса планеты |
| PlanetScale | Public | Float | Радиус коллайдера планеты |
| ColliderRadius\_Planet | Public | Float | Радиус коллайдера планеты |
| DwarfStarStartingMass | Public | Int | Стартовая масса карликовой звезды |
| DwarfStarCriticalMass | Public | Int | Критическая масса карликовой звезды |
| DwarfStarScale | Public | Int | Величина карликовой звезды |
| ColliderRadius\_DwarfStar | Public | Float | Радиус коллайдера карликовой звезды |
| StarStartingMass | Public | Float | Стартовая масса звезды |
| StarCriticalMass | Public | Int | Критическая масса звезды |
| StarScale | Public | Float | Величина звезды |
| ColliderRadius\_Star | Public | Float | Радиус коллайдера звезды |
| GiantStarStartingMass | Public | Int | Стартовая масса звезды-гиганта |
| GiantStarCriticalMass | Public | Int | Критическая масса звезды-гиганта |
| GiantStarScale | public | Float | Величина звезды-гиганта |
| ColliderRadius\_GiantStar | Public | Float | Радиус коллайдера звезды-гиганта |
| NeutronStarStartingMass | Public | int | Стартовая масса нейронной звезды |
| NeutronStarCriticalMass | Public | Int | Критическая масса нейронной звезды |
| NeutronStarScale | Public | Float | Величина нейтронной звезды |
| ColliderRadius\_NeutronStar | Public | Float | Радиус коллайдера нейтронной звезды |
| BlackHoleStartingMass | Public | Int | Стартовая масса черной дыры |
| BlackHoleCriticalMass | Public | Int | Критическая масса черной дыры |
| BlackHoleScale | Public | Float | Величина черной дыры |
| ColliderRadius\_BlackHole | Public | Float | Радиус коллайдера черной дыры |
| MaxGravitationalDistance | Public | Int | Максимальное расстояние для притяжения объектов |
| LowerSecondsGenPlanet | Public | Int | Минимум в секундах между генерациями объектов |
| UpperSecondsGenPlanet | Public | Int | Максимум в секундах между генерациями объектов |
| DistanceToGenerateObjects | Public | Int | Дистанция от игрока для генерации других космических тел |
| TrailDisapearTime | Public | Float | Время перед исчезновением следа от небесных тел |
| maxSpeed | Public | Float | Максимальная скорость игрока |
| minSpeed | Public | Float | Минимальная скорость игрока |
| ColorfulMaterialFolder | Public | String | Путь до цветных материалов |
| StarsMeterialsFolder | Public | String | Путь до материалов звезд |
| PlayerIsMoving | Public | bool | Двигается ли игрок |
| HierarchyMinMass | Public | Dictionary<String,int> | Словарь минимальных масс тел |
| HierarchyMaxMass | Public | Dictionary<String,int> | Словарь максимальных масс тел |
| LegalToSpawn | Public | Dictionary<String,String[]> | Словарь, где каждому типу космического объекта игрока соответствует словарь из разрешенных объектов для генерации |

Таблица 6 Описание полей и методов класса Events.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| NotifyMovement | public | EventHandler | Событие движения игрока | |
| lastTimeMovingEventFetched | private | System.Datetime | Последнее время, когда событие было вызвано | |
| rnd | private | Random | Генератор рандомных чисел | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| InvokeNotifyMovement | public | void | - | Уведомление всех подписчиков события для начала генерации космических объектов |
| StartSpawning | public | void | - | Вызывает функцию генерации космических объектов из другого класса |

Таблица 7

Описание полей и методов класса GameDeath.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| panel | public | GameObject | Панель для отображения всех кнопок начала игры и выхода в меню | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| ShowPanel | public | void | - | Отобразить панель на экране |
| Game\_Start | public | void | - | Загрузить сцену игры |
| MainMenu | public | void | - | Загрузить главное меню |

Таблица 8 Описание методов, свойств и полей класса GravitationalBody.cs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Поля | | | | | |
| Имя | Модификатор  доступа | Тип | | Назначение | |
| prefabs | private | PrefabStorage | | Ссылка на объект-хранилище префабов | |
| name | public | string | | Имя объекта | |
| maxDistance | public | float | | Максимальная дистанция для гравитационного влияния на другие объекты | |
| startingMass | private | float | | Переменная свойства StartingMass. Хранит массу объекта | |
| imaginaryMass | private | float | | Переменная свойства ImaginaryMass. Хранит максимальную массу объекта | |
| initialVelocity | public | Vector2 | | Вектор скорости космического тела | |
| rb | private | RigidBody2D | | Ссылка на RigidBody космического тела | |
| attractableBodies | private | List<RigidBody2D> | | Ссылки на объекты, которые попадают в поле гравитационного взаимодействия тела в данную секунду | |
| OrbitBodies | public | List<GameObject> | | Ссылки на объекты, которые находятся на орбите | |
| rnd | private | System.Random | | Генератор случайных чисел | |
| onOrbit | public | bool | | Находится ли тело на орбите | |
| target | public | Transform | | Тело, на чьей орбите мы находимся, если такое есть | |
| lastTimeDownGrade | private | DateTime | | Время последнего отката типа космического обхекта игрока | |
| FirstAsteroidToBeHit | private | bool | | Был ли поглощен первый астероид. Флаг для инциализации обучения | |
| FirstTrIntoDwarfPlanet | private | bool | | Было ли осуществено первое превращение в карликовую планету. Флаг для инциализации обучения | |
| Свойства | | | | | |
| Имя | Модификатор доступа | Тип | | Доступ | Назначение |
| StartingMass | public | float | | get;set; | Хранит массу объекта |
| ImaginaryMass | public | float | | get;set; | Хранит максимальную массу объекта |
| Методы | | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение | |
| Start | private | void | - | Осуществляет начальную настройку всего космического тела | |
| SetupMapPoint | private | void | - | Настройка иконки объекта для отображения на мини-карте | |
| SetupRigidbody2D | private | void | - | Настройка Rigidbody компонента у объекта | |
| SetupColliders | private | void | - | Настройка коллайдеров у объекта | |
| SetupTrailRenderer | private | void | - | Настройка следа объекта, который он оставляет | |
| SetupMeshRenderer | private | void | - | Настройка модели объекта | |
| FixedUpdate | private | void | - | Функция, вызывающаяся каждый кадр, просчитывает физику притяжения объектов, вызов функций для трансформаций | |
| OnMouseDown | private | void | - | Функция, вызывающаяся при клике на игрока, поглощает все объекты на орбите, восполняя массу | |
| onDestroy | private | void | - | Функция, вызывающаяся при смерти игрока и выводящая панель меню на экран | |
| onCollisionEnter2D | private | void | Collision2D coll | Вызывается при ивенте OnCollisionEnter2D и просчитывает увеличение/уменьшение и вызов трансофрмации при удовлетворяющих условиях | |
| CalculateMassGrow | public | int | - | Высчитывание массы при трансформации в больший объект | |
| CalculateMassGrow | public | int | String nm | Высчитывание массы при трансформации в больший объект по рангу | |
| ShareObjectData | public | string | String option | Возвращает информацию об игроке в удобном для отображения в UI виде | |
| GetMovementVector | public | Vector3 | - | Функция получения вектора направления игрока | |
| Upgrade\_Retransform\_Object | public | void | - | Функция, которая отвечает за старт трансформации объекта в больший | |
| Downgrade\_Retransform\_Object | public | Void | - | Функция, которая отвечает за старт трансформации объекта в меньший | |
| Retansform\_asteroid | Public | Void |  | Функция трансформирует объект в астероид | |
| Retransform\_DwarfPlanet | Public | Void | - | Функция трансформирует объект в карликовую планету | |
| Retransform\_Planet | Public | Void | - | Функция трансформирует объект в планету | |
| Retransform\_DwarfStar | Public | Void | - | Функция трансформирует объект в карликовую звезду | |
| Retransform\_Star | Public | Void | - | Функция трансформирует объект в звезду | |
| Retransform\_GiantStar | Public | Void | - | Функция трансформирует объект в звезду-гиганта | |
| Retransform\_NeutronStar | Public | Void | - | Функция трансформирует объект в нейтронную звезду | |
| Retransform\_BlackHole | public | Void | - | Функция трансформирует объект в черную дыру | |

Таблица 9 Описание методов и полей класса LineRendererEx.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| vertexCount | public | int | Количество углов у фигуры. Много углов – похоже на окрудность | |
| lineWidth | public | float | Ширина рисуемой линии | |
| radius | public | float | Расстояние от центра правильной фигуры до любого узла | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| SetupCircle | private | void | - | Проводит первоначальную настройку объекта для отрисовки фигуры |

Таблица 10 Описание полей и методов класса MenuCamera.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| Targetposition | public | Vector3 | Желаемая координата, которую должна занять камера | |
| Camera\_move\_enabled | public | bool | Разрешено ли камере менять свою координату | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Update | private | void | - | Каждый кадр на немного перемещает камеру к Targetposition если движение разрешено (Camera\_move\_enabled) |

Таблица 11 Описание полей и методов класса MenuManager.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| startGameButton | public | Button | Ссылка на ui элемент кнопки, ответственной за начало игры при клике | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| StartGame | private | void | - | Срабатывает при клике на кнопку и загружает сцену главной игры |

Таблица 12 Описание методов и полей класса Movement.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| speed | private | int | Скорость персонажа | |
| speedLimit | private | int | Максимальная скорость игрока | |
| isMoving | public | bool | Двигается ли игрок в данный момент | |
| rb | private | Rigidbody2D | Ссылка на компонент RigidBody игрока | |
| variableJoystick | public | FloatingJoystick | Сенсорный джойстик для управления | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Start | private | void | - | Получает ссылку на компонент RigidBody при инициализации объекта |
| Update | private | void | - | Обрабатывает сенсорные нажатия и нажатия клавиш клавиатуры, перемещая персонажа в правильном направлении |

Таблица 13 Описание методов и полей класса OrbitCatcher.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| isPlayerColliding | private | bool | Есть ли в зоне притяжения этого комического объекта другой? | |
| timerCountDown | public | float | Через столько секунд объект, находившийся в зоне притяжения, может выйти на орбиту | |
| player | private | GravitationalBody | Ссылка на привязанный скрипт GravitationalBody к игроку | |
| FirstOrbitObject | private | bool | Флаг для определения когда был выведен на орбиту первый объект. Нужно для обучения | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| OnTriggerEnter2D | private | void | Collider2D other | Сигнализирует когда объект вошел нашу гравитационную зону |
| OnTriggerStay2D | private | void | Collider2D other | Сигнализирует каждый кадр если объект продолжает находиться в нашем поле гравитации. В случае если таймер отсчитал положенное для выхода на орбиту время – выводит на орбиту |
| OnTriggerExit2D | private | void | Collider2D other | Сигнализирует когда объект покинул нашу зону гравитации |

Таблица 14 Описание полей и методов класса OrbitManager.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| OrbitRenderer | private | LineRenderer | Ссылка на объект отрисовки окружности орбиты | |
| player | private | GravitationalBody | Ссылка на GravitationalBody скрипт игрока | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| OnTriggerEnter2D | private | void | Collider2D other | Сигнализирует когда объект вошел нашу гравитационную зону для начала отрисовки следа за ним |
| OnTriggerStay2D | private | void | Collider2D other | Сигнализирует каждый кадр если объект продолжает находиться в нашем поле гравитации |
| OnTriggerExit2D | private | void | Collider2D other | Сигнализирует когда объект покинул нашу зону гравитации, чтобы перестать рисовать за ним след |

Таблица 15 Описание полей класса PrefabsStorage.cs

|  |  |  |  |
| --- | --- | --- | --- |
| Поля | | | |
| Имя | Модификатор  доступа | Тип | Назначение |
| asteroids | public | GameObject[] | Хранилище моделей астероидов |
| asteroidsMaterials | public | Material[] | Хранилище материалов астероидов |
| planets | public | GameObject[] | Хранилище моделей планет |
| stars | public | GameObject[] | Хранилище моделей звезд |
| BlackHole | public | GameObject[] | Хранилище моделей черной дыры |

Таблица 16 Описание методов и полей класса ProgressBar.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| barDisplay | public | float | Текущий прогресс | |
| pos | public | Vector2 | Позиция бара на экране | |
| size | public | Vector2 | Размер бара на экране | |
| emptyTex | public | Texture2D | Текстура фона прогресс бара | |
| fullTex | public | Texture2D | Текстура заполнения прогресс бара | |
| player | private | GameObject | Ссылка на объект игрока | |
| playersgravity | private | GravitationalBody | Ссылка на GravitationalBody скрипт игрока | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| OnGUI | private | void | - | Отрисовывает бар на экране |
| Start | private | void | - | Получает ссылку на объект игрока и его привязанный скрипт |
| FixedUpdate | private | void | - | Пересчитывает заполненность бара |

Таблица 17 Описание полей и методов класса ScrollBackground.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| speed | public | float | Скорость прокручивания фона | |
| M\_Material | private | Material | Хранит изображение фона в виде материала | |
| pl | private | GameObject | Ссылка на объект игрока | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Start | private | void | - | Получает ссылку на объект игрока и материал фона |
| FixedUpdate | private | void | - | Прокручивает каждый кадр фон в противоположном от движения игрока направлении, тем самым создавая эффект движения |

Таблица 18 Описание методов и полей класса Spawner.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| genType | public | string | Тип объекта генерации | |
| randomHash | private | int | Сид для рандома | |
| rnd | private | System.Random | Генератор случайных чисел | |
| lastTimeGenerated | private | System.DateTime | Время последней генерации | |
| prefabs | private | PrefabsStorage | Ссылка на хранилище космических объектов | |
| player | private | GameObject | Ссылка на объект игрока | |
| asteroidsCount | private | int | Подсчет кол-ва астероидов для генерации в начале игры | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| OnGUI | private | void | - | Отрисовывает бар на экране |
| Start | private | void | - | Получает ссылки на объекты игрока, хранилище |
| FixedUpdate | private | void | - | Генерирует астероиды вокруг игрока |
| GetRandomNumber | private | int | Int max, int min, double probabilityPower | Генерирует случайный id космического объекта. Чем больше число, тем меньше шанс его генерации |
| Spawn | public | void | - | Решает какой тип объекта сгенерировать и генерирует его |
| SpawnAsteroid | private | void | Vector3 vc | Инициализирует объект астероида и помещает его на сцене |
| SpawnDwarfPlanet | private | void | Vector3 vc | Инициализирует объект карликовой планеты и помещает его на сцене |
| SpawnPlanet | private | void | Vector3 vc | Инициализирует объект планеты и помещает его на сцене |
| SpawnDwarfStar | private | void | Vector3 vc | Инициализирует объект карликовой звезды и помещает его на сцене |
| SpawnStar | private | void | Vector3 vc | Инициализирует объект звезды и помещает его на сцене |
| SpawnGiantStar | private | void | Vector3 vc | Инициализирует объект звезды-гиганта и помещает его на сцене |
| SpawnNeutronStar | private | void | Vector3 vc | Инициализирует объект нейтронной звезды и помещает его на сцене |
| SpawnBlackHole | private | void | Vector3 vc | Инициализирует объект черной дыры и помещает его на сцене |

Таблица 19 Описание полей и методов класса TextTyper.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| uiText | public | Text | UI элемент текста для отображения | |
| firstText | public | string | Первая строчка вывода | |
| secondText | public | string | Вторая строчка вывода | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Start | private | IEnumerator | - | Выводит на экран начальное сообщение обучения |
| TypeText | public | TypeText | String text | При вызове и передаче параметра-строки выводит ее на экран с анимацией |

Таблица 20 Описание методов класса TrashCollector.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| OnCollisionEnter2D | private | void | - | Удаляет объект, который соприкоснулся с границей мира (находился слишком далеко от игрока) |

Таблица 21 Описание методов и полей класса UIController.cs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поля | | | | |
| Имя | Модификатор  доступа | Тип | Назначение | |
| planetName | public | Text | UI элемент текста для отображения типа объекта игрока | |
| Mass | public | Text | UI элемент текста для отображения массы | |
| Vector | public | Text | UI элемент текста для отображения вектора-направления | |
| Coords | public | Text | UI элемент текста для отображения координатов игрока | |
| nextObject | public | Text | UI элемент текста для отображения следующего объекта игрока | |
| player | public | Gameobject | Ссылка на объект игрока | |
| Методы | | | | |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Start | private | void | - | Выводит начальный текст всех характеристик на экран |
| FixedUpdate | private | void | - | Обновляет текст всех характеристик на экране |

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подп. | Дата |
| Измененных | замененных | новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |