

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Образовательная программа бакалавриата «Программная инженерия»

**СОГЛАСОВАНО**  
Научный руководитель,  
младший научный сотрудник  
МНУЛ ИССА ФКН, к. т. н.

**УТВЕРЖДАЮ**  
Академический руководитель  
образовательной программы  
«Программная инженерия»  
профессор департамента программной  
инженерии, канд. техн. наук

\_\_\_\_\_ О. В. Максименкова  
«\_\_» \_\_\_\_\_ 2020 г.

\_\_\_\_\_ В.В. Шилов  
«\_\_» \_\_\_\_\_ 2020 г.

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

**Браузерная мультиплеерная игра "Minigames"**

**Пояснительная записка**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.01-01 81 01-1-ЛУ**

Исполнитель  
студент группы БПИ 196

\_\_\_\_\_ /М.С. Шестаков /  
«\_\_» \_\_\_\_\_ 2020 г.

**Москва 2020**

УТВЕРЖДЕН  
RU.17701729.04.01-01 81 01-1-ЛУ

**Браузерная мультиплеерная игра "Minigames"**

**Пояснительная записка**

**RU.17701729.04.01-01 81 01-1**

**Листов 83**

<i>Подп. и дата</i>	
<i>Инв. № дубл.</i>	
<i>Взам. инв. №</i>	
<i>Подп. и дата</i>	
<i>Инв. № подл</i>	

**Москва 2020**

**RU.17701729.04.01-01 81 01-1****СОДЕРЖАНИЕ**

## Оглавление

1.	ВВЕДЕНИЕ .....	4
1.1	Наименование программы.....	4
1.2	Документы, на основании которых ведется разработка программы .....	4
2.	НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ .....	5
2.1	Назначение программы.....	5
2.1.1	Функциональное назначение.....	5
2.1.2	Эксплуатационное назначение .....	5
2.2	Краткая характеристика области применения .....	5
3.	ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ .....	6
3.1	Постановка задачи на разработку программы .....	6
3.2	Описание алгоритма и функционирования программы.....	6
3.2.1	Описание игрового процесса .....	6
3.2.2	Описание функционирования интерфейса в игре.....	6
3.2.3	Описания функционирования передвижения персонажа .....	9
3.2.4	Описание функционирования системы действий персонажа .....	9
3.2.5	Описание функционирования системы здоровья персонажа .....	9
3.2.6	Описание функционирования системы оружия .....	9
3.2.7	Описание функционирования выстрелов .....	10
3.2.8	Описание работы системы игровых режимов (миниигр).....	10
3.2.9	Описание работы системы начисления очков и выбора победителя .....	10
3.2.10	Описание функционирования подвижной платформы .....	10
3.2.11	Описание функционирования поиска матча.....	10
3.2.12	Описание функционирования системы команд и взаимодействия с сервером .....	10
3.2.13	Описание работы JSON-запросов.....	12
3.3	Описание и обоснование выбора метода организации входных и выходных данных .....	13
3.3.1	Описание метода организации входных и выходных данных .....	13
3.3.2	Обоснование метода организации входных и выходных данных.....	13
3.4	Описание и обоснование выбора состава технических и программных средств .....	13
4.	ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ .....	15

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

4.1	Предполагаемая потребность .....	15
4.2	Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами.....	15
ПРИЛОЖЕНИЕ 1 ТЕРМИНОЛОГИЯ .....		16
ПРИЛОЖЕНИЕ 2 .....		17
ПРИЛОЖЕНИЕ 3 .....		82
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ .....		82
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ .....		83

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1****1. ВВЕДЕНИЕ****1.1 Наименование программы**

Наименование программы – "Миниигры"

Наименование программы на английском языке – "Minigames"

**1.2 Документы, на основании которых ведется разработка программы**

Разработка программы ведётся на основании приказа 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 11.12.2020.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1****2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ****2.1 Назначение программы****2.1.1 Функциональное назначение**

Программа является игрой и предназначена для развлечения. Содержит несколько мини-игр, поэтому игровой процесс является разнообразным. Программа запускается в браузере, поэтому ее не нужно скачивать и устанавливать. Может быть использована в любой ОС с современным браузером и доступом в интернет. Программа поддерживает мультиплеер, поэтому в нее можно играть с друзьями.

Развивает умение действовать в новой ситуации и скорость реакции.

**2.1.2 Эксплуатационное назначение**

Программа разделена на 2 части — клиент и сервер.

Клиентская часть может эксплуатироваться на любом ПК с наличием доступа в интернет и современным браузером. Для работы клиентской части необходим сервер.

Серверная часть может быть запущена на ПК с установленной Windows 7 и выше / Ubuntu 16.04 или другой совместимой с ней ОС семейства Linux, и наличием доступа в интернет.

Конечные потребители — любители казуальных игр, без какой-либо специальной подготовки.

**2.2 Краткая характеристика области применения**

Программа представляет собой игру, предназначена для развлечения и повышения настроения.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### 3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

#### 3.1 Постановка задачи на разработку программы

В программе должны быть реализованы следующие функции:

- 1) Выбор никнейма для персонажа
- 2) Поиск случайного матча
- 3) Управление персонажем
- 4) Подсчёт очков в игре
- 5) Показ промежуточных и финальных результатов игры
- 6) Возможность отправить сообщение во внутриигровой чат
- 7) Показ информации от текущем режиме (задача и оставшееся время)

#### 3.2 Описание алгоритма и функционирования программы

##### 3.2.1 Описание игрового процесса

Игрок управляет персонажем от третьего лица. Игроку нужно набрать больше финальных очков, чем другие, чтобы победить. Финальные очки выдаются в конце каждой миниигры в зависимости от занятого игроком места (чем выше место, тем больше он получит финальных очков). Чтобы занять более высокое место в миниигре пользователь должен собрать как можно больше игровых очков, которые выдаются за выполнение игровых задач. Игровые задачи определяются миниигрой.

##### 3.2.2 Описание функционирование интерфейса в игре

Стартовое меню, которое показывается после загрузки игры (рис. 1):



Рисунок 1. Стартовое меню

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

В нём можно ввести никнейм и начать игру.

После нажатия кнопки играть показывается меню ожидания матча (рис. 2):



Рисунок 2. Меню ожидания матча

В нём пользователь может посмотреть информацию о матче.

Во время игры, игроку показывается следующий интерфейс (рис. 3):

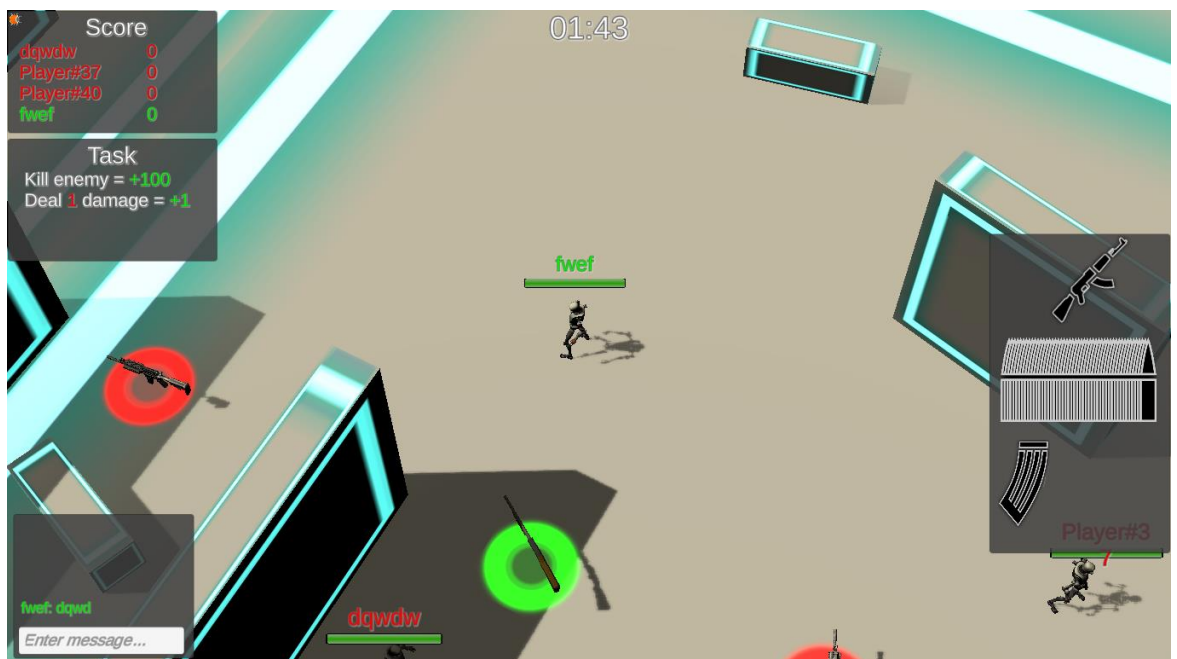


Рисунок 3. Интерфейс во время игры

В нём показывается текущее количество очков у каждого игрока, задача, предметы в зависимости от текущей миниигры. Так же с помощью данного интерфейса пользователь может отправить сообщение в чат.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# RU.17701729.04.01-01 81 01-1

Между минииграми игроку показывается меню с промежуточными результатами игры (рис. 4):



Рисунок 4. Меню с промежуточными результатами

В нём показывается оставшееся количество миниигр, время до следующей миниигры, а также текущая таблица результатов.

В конце игры пользователю будет показано следующее меню с финальными результатами (рис. 5):



Рисунок 5. Меню с промежуточными результатами

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

В нём показывается победитель и таблица результатов. Так же в данном меню есть кнопка “Exit”, которая позволяет выйти из текущей игры и попасть в стартовое меню.

### 3.2.3 Описания функционирования передвижения персонажа

За передвижение персонажа отвечает компонента MotionController. У неё есть свойства TargetDirection и TargetRotation, которые указывают направление, в котором должен двигаться персонаж и направление, в котором развёрнут прицел персонажа. MotionController через специальный коллайдер, расположенный у ног персонажа определяет, стоит ли он на земле. Далее, если персонаж стоит на земле, то MotionController устанавливает его вектор скорости в направлении TargetDirection. Если скорость персонажа меньше определённого значения, то вместо прямой установки вектора скорости на максимальное значение, контроллер прикладывает усилие к центру персонажа, имитируя отталкивание ногами от земли. Это необходимо для корректной работы физической системы при столкновении с другими объектами.

### 3.2.4 Описание функционирования системы действий персонажа

Для управления действиями у каждого персонажа есть компонента ActionController. Она умеет хранить текущее действие, выполнять его и сменять на другое.

### 3.2.5 Описание функционирования системы здоровья персонажа

Для управления здоровьем персонажа используется компонента HPController. Она хранит текущее количество здоровья, максимальное количество здоровья и отображает это на специальной полосе рядом с персонажем. Так же через эту компоненту происходит нанесение урона. Если здоровье персонажа становится равным нулю, то HPController отправляет событие о смерти персонажа.

### 3.2.6 Описание функционирования системы оружия

В игре есть три вида оружия: пистолет, автомат и дробовик. У каждого оружия есть патроны и магазины, работа с которыми организована через класс ReloadingGun. На один выстрел из оружия тратится один патрон. Когда патроны кончаются, персонажу требуется перезарядить оружие. На перезарядку тратится один магазин. Когда у оружия кончаются патроны и магазины, оно автоматически исчезает из инвентаря персонажа. Количество патронов и магазинов зависит от типа оружия и может быть задана через интерфейс инспектора в Unity Editor.

Выстрелы из оружия происходят через систему действий. Для этого есть два вида действий: ShootPistolAction для стрельбы по одному выстрелу за раз (используется в пистолете и дробовике) и ShootSemiautoAction для стрельбы до тех пор, пока мышка удерживается (используется в автомате).

Каждое оружие может находиться как в инвентаре у персонажа, так и лежать на земле. Когда персонаж подходит к оружию, лежащему на земле, оно автоматически попадает к нему в инвентарь. В случае, если у игрока уже было какое-то оружие, оно заменяется подобранным.

Ключевым отличием между видами оружия является механика стрельбы: пистолет производит один выстрел, дробовик стреляет дробью, автомат выпускает много выстрелов подряд. Так же у оружия может быть разброс. У дробовика он наибольший, у автомата средний, у пистолета полностью отсутствует.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

### 3.2.7 Описание функционирования выстрелов

Выстрелы из оружия происходят через специальный класс ShootSystem. При выстреле с помощью физического движка Unity происходит поиск места, куда попала пуля. В случае попадания в персонажа по сети передаётся команда DrawTargetedTracerCommand, которая рисует след от пули между персонажем, который стрелял и персонажем в которого попали. В случае, если игрок промахнулся трасер будет нарисован между ним и точкой с помощью команды DrawPositionTracerCommand. Данная методика позволяет скорректировать отклонения в синхронизации, связанные с задержкой передачи данных. Так же в ShootSystem реализован функционал для создания разброса, который работает с помощью распределения Гаусса.

### 3.2.8 Описание работы системы игровых режимов (миниигр)

В игре есть несколько игровых режимов. Каждый из них обладает своими правилами игры. На данный момент разработано два режима:

#### 1) Шутер

- На карте в случайных местах появляется оружие
- Цель игрока – нанести как можно больше урона по другим персонажам

#### 2) Сбор монеток

- На карте в случайных местах появляется оружие
- Цель игрока – собрать как можно больше монеток

### 3.2.9 Описание работы системы начисления очков и выбора победителя

За определённые действия в каждой миниигре игрок будут получать очки. После окончания миниигры и подсчёта результатов, игроки сортируются по количеству очков. Далее, согласно полученному месту, каждый игрок получает определённое количество финальных очков. Тот, кто собрал больше всех финальных очков за все миниигры (суммарно) побеждает.

### 3.2.10 Описание функционирования подвижной платформы

В начале каждой игры выбирается клиент, который будет управлять каждой подвижной платформой. Платформа движется между с двумя заранее установленными позициями. Когда персонаж встаёт на платформу, он начинает перемещаться вместе с ней.

### 3.2.11 Описание функционирования поиска матча

Когда игрок вводит ник и нажимает кнопку “Play”, начинается поиск матча. Для этого на сервер отправляется запрос о всех текущих свободных матчах. Если сервер нашёл свободный матч, то игрок присоединяется к нему. Иначе клиент отправляет к серверу запрос на создание нового матча. После того, как в матче набирается необходимое количество игроков начинается игра.

### 3.2.12 Описание функционирования системы команд и взаимодействия с сервером

Взаимодействие с сервером происходит посредством специального бинарного протокола.

Существует несколько типов сообщений, которые клиент может отправить к серверу:

#### 1) Обычное сообщение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

- Параметры: номер комнаты (4 байта), флаги (1 байт), тело сообщения
  - Тип сообщения = 1
  - Кодируется следующим образом: Тип сообщения (1 байт) + Комната (4 байта) + Флаги (1 байт) + Тело сообщения (любое кол-во байт)
  - После получения такого сообщения сервер разошлёт его всем другим клиентам в комнате
- 2) Сообщение с уникальным кодом
- Параметры: номер комнаты (4 байта), флаги (1 байт), уникальный код (8 байт), тело сообщения
  - Тип сообщения = 2
  - Кодируется следующим образом: Тип сообщения (1 байт) + Комната (4 байта) + Флаги (1 байт) + Уникальный код (8 байт) + Тело сообщения (любое кол-во байт)
  - После получения такого сообщения сервер разошлёт его всем другим клиентам в комнате, если ранее сообщения с таким уникальным кодом в эту комнату не поступало. В противном случае сервер проигнорирует данное сообщение.
- 3) Запрос на пересылку сообщений
- Параметры: номер комнаты (4 байта), флаги (1 байт), номер сообщения с которого нужно начать пересылку (4 байта), номер сообщения до которого нужно переслать сообщения (4 байта)
  - Тип сообщения = 3
  - Кодируется следующим образом: Тип сообщения (1 байт) + Комната (4 байта) + Флаги (1 байт) + Номер сообщения с которого нужно начать пересылку (4 байта) + Номер сообщения до которого нужно переслать сообщения (4 байта)
  - После получения такого сообщения сервер перешлёт клиенту все сообщения, лежащие между заданными номерами
- 4) Запрос на присоединение к комнате
- Параметры: номер комнаты (4 байта), флаги (1 байт)
  - Тип сообщения = 4
  - Кодируется следующим образом: Тип сообщения (1 байт) + Комната (4 байта) + Флаги (1 байт)
  - После получения такого сообщения сервер должен добавить игрока в заданную комнату. Если комнаты с таким номер не существует, сервер должен её предварительно создать
- 5) Запрос на покидание комнаты
- Параметры: номер комнаты (4 байта), флаги (1 байт)
  - Тип сообщения = 5

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

- Кодировается следующим образом: Тип сообщения (1 байт) + Комната (4 байта) + Флаги (1 байт)

- После получения такого сообщения сервер должен удалить игрока из заданной комнаты

**6) JSON-запрос**

- Параметры: номер комнаты (4 байта), флаги (1 байт), JSON-строка

- Тип сообщения = 6

- Кодировается следующим образом: Тип сообщения (1 байт) + Комната (4 байта) + Флаги (1 байт) + JSON строка, закодированная в UTF-8

- Обработка такого сообщения сервером зависит от внутренних параметров сообщения

Сервер отправляет клиенту сообщение следующего вида:

- Порядковый номер сообщения (4 байта), Номер комнаты, в которое направлено сообщение (4 байта), тело сообщения
- В случае, если сервер отправляет клиенту JSON-ответ, то тело сообщения будет иметь следующий вид: число 254 (1 байт), JSON-строка, закодированная в формате UTF-8
- В случае, если сервер хочет отправить пустое сообщение, тело сообщения будет иметь вид: число 255 (1 байт), тело сообщения

Так же у каждого сообщения от клиента к серверу есть специальные флаги:

- Если у переменной с флагами установлен первый бит в 1, значит отправляемое сообщение важное и сервер должен сохранить его у себя (чтобы иметь потом возможность заново его отправить другим клиентам в случае потери). Данный флаг применим только для 1-го и 2-го типов сообщений.
- Если у переменной с флагами установлен второй бит в 1, значит клиент хочет, чтобы сервер переотправил только сообщения, помеченный важными. Данный флаг применим только для 3-го типа сообщений.

Клиенты обмениваются между собой специальными командами (через сообщения к серверу), которые кодируются следующим образом: Код команды (1 байт) + Тело команды (несколько байт). Тело команды представляет собой поля команды, закодированные в порядке их объявления в классе команды. Создание кода для сериализации и десериализации команд происходит посредством специального скрипта на питоне – codegen.py.

Во время поиска матча игрок присоединяется к специальной комнате сервера. Так же существует комната для матча, в которой обрабатываются сообщения, связанные с глобальными событиями в матче (например, выбор следующего игрового режима). В комнате для текущего игрового режима происходит отправка и получение сообщений, связанных с текущей миниигрой.

**3.2.13 Описание работы JSON-запросов**

Для более удобного общения с сервером в программе реализованы JSON-запросы. Каждый запрос кодируется в виде JSON-объекта, в котором обязательно должны быть поля `_id` – номер запроса и `_type` – тип запроса. Существует несколько типов запросов: получить список свободных матчей, создать, присоединиться к матчу, начать матч.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

### 3.3 Описание и обоснование выбора метода организации входных и выходных данных

#### 3.3.1 Описание метода организации входных и выходных данных

Входные данные представляют собой нажатия клавиш клавиатуры и движение мыши. Выходными данными программы является изображение. Формат изображения определяется размером окна в браузере и характеристиками монитора. Выходные данные серверной части представляют собой логи событий.

#### 3.3.2 Обоснование метода организации входных и выходных данных

Метод организации входных и выходных данных клиентской части выбран исходя из простоты и удобства в использовании для конечного пользователя. Метод организации входных и выходных данных серверной части был выбран для удобства отладки программных ошибок.

### 3.4 Описание и обоснование выбора состава технических и программных средств

Для надёжной и бесперебойной работы программы требуется следующий состав технических и программных средств:

#### Для серверной части:

- 1) Центральный процессор: частота не ниже 1.2 ГГц, поддержка набора инструкций SSE2;
- 2) Операционная система: Windows 7 и выше, Ubuntu 16.04 или другая совместимая с ней ОС семейства Linux;
- 3) Оперативная память: не менее 256 МБ свободной ОЗУ;
- 4) Свободное место: не менее 1.5 ГБ на жестком диске;
- 5) Node JS последней стабильной версии
- 6) Интернет соединение: скорость не ниже 8 мбит / сек

Поскольку сервер работает на платформе Node JS, то ему нужно данное программное обеспечение и совместимая с ним система.

Заданное количество свободного места необходимо для хранения исполняемого кода сервера, логов и модулей Node JS.

Серверу необходимо интернет соединение с заданной скоростью для обмена данными с пользователями без задержек.

Заданное количество ОЗУ необходимо для корректной работы Node JS и хранения данных о текущих матчах.

#### Для клиентской части:

- 1) Центральный процессор: 64 битный с частотой не ниже 2.1 ГГц;
- 2) Графическая карта: совместимая с технологией WebGL;
- 3) Оперативная память: не менее 512 МБ свободной ОЗУ;
- 4) Свободное место: достаточное количество свободного места на жестком диске для корректной работы с браузером;

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

- 5) Периферийные устройства: Клавиатура, мышь;
- 6) Интернет соединение: скорость не ниже 4 мбит / сек;
- 7) Современный браузер с поддержкой технологий WebGL, HTML 5, WASM

Требования к браузеру и процессору выбраны исходя из ограничений, указанных на сайте Unity к WebGL сборкам.

Программе необходимо заданное количество ОЗУ для хранения текстур и исполняемого файла игры, поскольку браузер сохраняет все файлы в ней.

Требование к скорости интернет соединению выбрано для возможности обмена данных с другими игроками и сервером без задержек.

Клавиатура и мышь необходимы исходя из требований к входным и выходным данным.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

### 4. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

#### 4.1 Предполагаемая потребность

В мире много любителей казуальных игр. Короткая игровая сессия позволяет не тратить на игру много времени и заходить несколько раз в течение дня. Игру не надо скачивать, она запускается на любой ОС. Это увеличивает потенциальную аудиторию. Целевая аудитория – люди всех возрастов, любители казуальных игр.

#### 4.2 Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

Данная игра бесплатная. Отличительная особенность – несколько разнообразных мини-игр, которые выбираются случайно в каждой игровой сессии. Браузерных игр с такой механикой обнаружить не удалось.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# RU.17701729.04.01-01 81 01-1

## ПРИЛОЖЕНИЕ 1

### ТЕРМИНОЛОГИЯ

**Мини-игра** – небольшая игра, в которой игроку нужно быстрее своих соперников понять цель игры и выполнить ее. Длительность такой мин-игры не превышает 1-2 минуты.

**Логи** – файлы с отчетами о событиях, возникших в ходе работе программы. События указываются в хронологическом порядке.

**Мультиплеер** – режим компьютерной игры, при котором в нее играет несколько человек.

**Никнейм (игровой ник)** – псевдоним пользователя, показывающийся другим игрокам.

**Казуальная игра** – компьютерная игра с простыми правилами и несложным управлением, ориентированная на большую аудиторию.

**Игровая механика** – набор правил и способов, реализующий определённым образом некоторую часть интерактивного взаимодействия игрока и игры.

**Игровой движок** – программное обеспечение, используемое для разработки компьютерных игр.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

## ПРИЛОЖЕНИЕ 2

Классы	Назначение
AIController	Класс для игрока, управляемого искусственным интеллектом
AutoRotateToCamera	Компонента для автоматического поворота интерфейса к камере
BulletTrailRenderer	Компонента для отрисовки следа от пули
Character.ActionController	Компонента для обработки действий, которые может делать игрок
Character.Actions.IAction	Интерфейс действия пользователя
Character.Actions.PushAction	Действие персонажа, при котором он толкает предметы перед собой с определенной силой
Character.Actions.ShootAction<T>	Действие персонажа, при котором он стреляет по врагам
Character.Actions.ShootPistolAction	Действие для стрельбы из пистолета (делает один выстрел за клик)
Character.Actions.ShootSemiautoAction	Действие для стрельбы из автомата (непрерывно производит выстрелы, пока действие выполняется)
Character.CameraFollower	Компонента для следования камеры за игроком
Character.CharacterAnimator	Класс для анимации персонажа
Character.CharacterController	Базовый класс для компоненты, управляющей персонажем
Character.Guns.BombGun	Класс для гранатомета
Character.Guns.BombGunController	Класс для гранатомета, расположенного в виде игрового объекта на поле
Character.Guns.GunController<T>	Класс для оружия, которое расположено на игровом поле и которое можно подобрать
Character.Guns.GunState	Перечисление с возможными состояниями оружия
Character.Guns.IGun	Интерфейс оружия
Character.Guns.Pistol	Класс для пистолета
Character.Guns.PistolController	Класс для пистолета, расположенного в виде игрового объекта на поле
Character.Guns.ReloadGun	Базовый класс для оружия, которое может перезаряжаться
Character.Guns.SemiautoController	Класс для автомата, расположенного в виде игрового объекта на поле
Character.Guns.SemiautoGun	Класс для автомата

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Character.Guns.ShootSystem	Класс с функциями для стрельбы
Character.Guns.ShotGun	Класс для дробовика
Character.Guns.ShotgunController	Класс для дробовика, расположенного в виде игрового объекта на поле
Character.HP.DamageSource	Класс для работы с источниками урона
Character.HP.HPChange	Структура, представляющая собой изменение здоровья игрового объекта
Character.HP.HPController	Класс для компоненты здоровья игрового объекта
Character.MotionController	Компонента для передвижения персонажа по игровому полю
Character.PlayerAnimationState	Структура, хранящая состояние анимации персонажа
Client	Класс с функциями для работы с игровым полем
Coin	Класс для монетки
CommandsSystem.Commands.AddOrChangeInstance	Команда, сообщающая о изменении одного из Instance в текущей игре
CommandsSystem.Commands.AddPlayerToGame	Добавляет игрока в игру
CommandsSystem.Commands.ApplyForceCommand	Команда, сообщающая, что к игровому объекту нужно применить силу
CommandsSystem.Commands.ChangeHPCommand	Команда, сообщающая что нужно изменить здоровье объекта
CommandsSystem.Commands.ChangePlayerProperty	Класс для хранения и синхронизации состояния игрока
CommandsSystem.Commands.ChangePlayerScore	Команда для изменения очков игрока
CommandsSystem.Commands.CreateChatMessageCommand	Команда для отправки сообщения в чат
CommandsSystem.Commands.DrawPositionTracerCommand	Команда рисования следа от пули между персонажем и координатой
CommandsSystem.Commands.DrawTargetedTracerCommand	Команда рисования следа от пули между персонажами
CommandsSystem.Commands.ExplodeBombCommand	Команда для взрыва указанной бомбы
CommandsSystem.Commands.PickCoinCommand	Команда, сообщающая, что игрок подобрал монетку
CommandsSystem.Commands.PickUpGunCommand	Команда, сообщающая, что персонаж подобрал оружие
CommandsSystem.Commands.PlayerProperty	Состояние персонажа, которое можно синхронизировать по сети
CommandsSystem.Commands.PlayerPushCommand	Команда сообщающая, что для данного игрока нужно показать анимацию толкания

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

CommandsSystem.Commands.SetAfterShowResultsCommand	Команда, сообщающая что GameManager должен перестать показывать результаты игры
CommandsSystem.Commands.SetGameMode	Команда, сообщающая, что нужно запускать заданный игровой режим
CommandsSystem.Commands.SetPlatformStateCommand	Команда для синхронизации состояния подвижной платформы
CommandsSystem.Commands.SpawnParabolaFlyingCommand	Команда для создания объекта с компонентой ParabolaFlyingObject
CommandsSystem.Commands.SpawnPlayerCommand	Команда для создания на игровом поле персонажа
CommandsSystem.Commands.SpawnPrefabCommand	Команда для создания объекта на игровом поле
CommandsSystem.Commands.StartGameCommand	Команда начать игру
CommandsSystem.Commands.TakeOwnCommand	Команда для смены владельца у объекта
CommandsSystem.CommandsSystem	Класс для сериализации и десериализации команд
CommandsSystem.ICommand	Интерфейс для команды
DefaultNamespace.AutoHideOnStart	Компонента, автоматически отключающая объект при старте
Editor.ClientEditor	Класс для показа отладочной информации в инспекторе Unity
Events.EventsHandler	Класс для обработки событий
Events.EventsManager	Класс для управления обработкой событиями
Game.Game.MatchesManager.STATE	Перечисление возможных состояний менеджера матчей
Game.MatchInfo	Класс для информации о матче
Game.MatchesManager	Класс для управления и присоединения к игровым матчам
GameMechanics.Bomb	Класс для бомбы
GameMechanics.BombOnCollisionExploder	Компонента для взрыва бомбы при столкновении с объектом
GameMechanics.BombTriggerHPEExploder	Компонента для взрыва бомбы при столкновении с объектом, имеющим здоровье
GameMechanics.MovingPlatform	Класс для подвижной платформы
GameMechanics.ParabolaFlyingObject	Компонента для объекта, летящего по траектории параболы
GameMode.GameManager	Класс для управления игрой
GameMode.GameMode.GameManager.STATE	Перечисление возможных состояний игры
GameMode.GameMode.PickCoinsGameMode.STATE	Перечисление состояний режима
GameMode.GameMode.ShooterGameMode.STATE	Перечисление состояния игрового режима
GameMode.GameModeFunctions	Класс с функциями для игровых режимов
GameMode.IGameMode	Интерфейс для игрового режима

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## RU.17701729.04.01-01 81 01-1

GameMode.Instance	Класс для хранения и синхронизации информации о клиенте
GameMode.InstanceManager	Класс для управления instance
GameMode.PickCoinsGameMode	Класс для игрового режима со сбором монет
GameMode.Player	Класс для игрока
GameMode.PlayerStorage	Компонента для хранения ссылки игрока внутри персонажа, которым он управляет
GameMode.PlayersManager	Класс для хранения информации об игроках
GameMode.ShooterGameMode	Класс для игрового режима в формате шутера
HandAnimationBlender	Класс для анимации рук персонажа
IOwnedEventHandler	Интерфейс для обработки событий, когда у объекта меняется владелец
Interpolation.Interpolation.UnmanagedGameObject<T>.Data	Данные для синхронизации объекта
Interpolation.InterpolationFunctions	Класс с функциями для интерполяции
Interpolation.ManagedGameObject<T>	Компонента для игрового объекта, управляемого из текущего клиента
Interpolation.Managers.PlayerManagedGameObject	Компонента для персонажа, управляемого из текущего клиента
Interpolation.Managers.PlayerUnmanagedGameObject	Компонента для персонажа, управляемого из другого клиента
Interpolation.Properties.GameObjectProperty<T>	Базовый класс для состояния, которое можно синхронизировать по сети
Interpolation.Properties.IGameObjectProperty	Интерфейс для состояния, которое можно синхронизировать по сети
Interpolation.UnmanagedGameObject<T>	Компонента для объекта, управляемого из другого клиента
JsonRequest.Request	Класс для JSON запроса к серверу
JsonRequest.RequestType	Тип JSON запроса к серверу
JsonRequest.RequestsManager	Класс для управления JSON запросами
JsonRequest.Response	Класс, представляющий ответ сервера на JSON запрос
MainUIController	Компонента, управляющая главным интерфейсом в игре
MessageFlags	Флаги для отправки сообщений на сервер
MessageType	Перечисления со видами сообщений, которые можно отправить на сервер

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Networking.ClientCommandsRoom	Класс комнаты для общения с сервером
Networking.CommandsHandler	Класс для обработки команд с сервера
Networking.UniqCodes	Перчисление с кодами для уникальных команд
ObjectID:	Класс для хранения объектов по ID
ObjectID:.ObjectData	Класс для хранения объекта с данными
PlayerController	Компонента для персонажа, управляемого человеком
QueueExtension	Класс с дополнительными функциями для очереди
Triangle	Класс для треугольника
TrianglePolygon	Класс, представляющий выпуклый многоугольник, составленный из треугольников
UI.DebugUI	Компонента отладочного интерфейса
UI.MultiImagePanel	Класс для панели интерфейса с одинаковыми изображениями
UI.StartUIController	Компонента для управления стартовым интерфейсом в игре
Util2.AutoDisableRendererOnStart	Компонента, автоматически отключающая Renderer у объекта при старте
Util2.AutoID	Компонента, автоматически присваивающая объекту ID
Util2.AutoMatchJoiner	Компонента для автоматического поиска и входа в матч
Util2.RotatingItem	Компонента для вращающегося игрового объекта
Util2.gUtil	Класс с дополнительными методами
Vector2Extension	Класс с дополнительными функциями для Vector2
WebGLEditorScript	Класс для установки настроек сборки в webgl
WebSocketHandler	Класс для работы с WebSocket
sClient	Главный класс, управляющий игрой, выбором матча
sClient.STATE	Состояние

Таблица 3.1

Описание полей, методов и свойств класса AIController

Методы класса AIController				
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### RU.17701729.04.01-01 81 01-1

Имя	Модификатор доступа	Тип	Аргументы	Назначение
Update	private	void		Изменяет параметры игрока на основе ИИ. Автоматически вызывается Unity каждый кадр

Таблица 3.2

Описание полей, методов и свойств класса AutoRotateToCamera

Методы класса AutoRotateToCamera				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
LateUpdate	private	void		Поворачивает объект к камере. Автоматически вызывается Unity каждый кадр

Таблица 3.3

Описание полей, методов и свойств класса BulletTrailRenderер

Поля класса BulletTrailRenderер				
Имя	Модификатор доступа	Тип	Назначение	
main	private	bool	Является ли данная компонента главной, от которой все копируются	
speed	public	float	Скорость	
v1	public	Vector3	Первая позиция	
v2	public	Vector3	Вторая позиция	
Методы класса BulletTrailRenderер				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
MoveFromTo	public	void	Vector3 v1, Vector3 v2	Создаёт след от пули
_moveTo	private	void	Vector3 v2	Устанавливает цель, к которой должен следовать данный след

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Update	private	void		Передвигает след от пули
--------	---------	------	--	--------------------------

Таблица 3.4

Описание полей, методов и свойств класса Character.ActionController

Поля класса Character.ActionController				
Имя	Модификатор доступа	Тип	Назначение	
animator	private	CharacterAnimator	Ссылка на CharacterAnimator игрока	
currentAction	private	IAction	Текущее действие	
Target	public	Vector3	Координата, в которую направлен прицел игрока	
_actionDoing	private	bool	Переменная для хранения DoAction	
Методы класса Character.ActionController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменны
StopCurrent	private	void		Прекращает выполнение текущего действия
SetAction<T>	public	void	System.Action<T> setup	Изменяет действие персонажа
SetNothing	public	void		Изменяет действие перонажа на пустое

Таблица 3.5

Описание полей, методов и свойств интерфейса Character.Actions.IAction

Методы интерфейса Character.Actions.IAction				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnStartDoing	private	void		Функция, которая вызывается, когда игрок начинает делать действие
OnStopDoing	private	void		Функция, которая вызывается,

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# **RU.17701729.04.01-01 81 01-1**

				когда игрок прекращает делать действие
--	--	--	--	--

Таблица 3.6

Описание полей, методов и свойств класса Character.Actions.PushAction

Поля класса Character.Actions.PushAction				
Имя	Модификатор доступа	Тип	Назначение	
pushCollider	public	GameObject	Объект, внутри которого нужно толкать предметы	
force	public	float	Сила с которой нужно толкать предметы	
animator	private	CharacterAnimator	Ссылка на CharacterAnimator у игрока	
Методы класса Character.Actions.PushAction				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
OnStartDoing	public	void		Начинает выполнять данное действие
pushEnd	public	void		Толкает предметы перед собой. Автоматически вызывается Unity в конце анимации рук персонажа
OnStopDoing	public	void		Заканчивает выполнять данное действие

Таблица 3.7

Описание полей, методов и свойств класса Character.Actions.ShootAction<T>

Поля класса Character.Actions.ShootAction<T>			
Имя	Модификатор доступа	Тип	Назначение
animator	private	CharacterAnimator	Ссылка на CharacterAnimator у игрока
gun	public	T	Оружие, из которого игрок стреляет

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Методы класса Character.Actions.ShootAction<T>				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnEnable	private	void		Обрабатывает подбор оружия игроком. Вызывается при включении скрипта
OnDisable	private	void		Обрабатывает потерю оружия игроком. Вызывается при выключении скрипта
Start	private	void		Инициализирует переменные
OnStartDoing	public	void		Функция, которая вызывается, когда игрок начинает делать действие
OnStopDoing	public	void		Функция, которая вызывается, когда игрок прекращает делать действие

Таблица 3.8

Описание полей, методов и свойств класса Character.Actions.ShootPistolAction

Поля класса Character.Actions.ShootPistolAction				
Имя	Модификатор доступа	Тип	Назначение	
needShoot	private	float	Время, когда была отдана последняя команда стрелять. Нужно для возможности отдать команду стрелять, если до перезарядки орудия осталось меньше 150 мсек	
Методы класса Character.Actions.ShootPistolAction				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnStartDoing	public	void		Отдает команду стрелять

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

OnStopDoing	public	void		Вызывается при прекращении действия
LateUpdate	private	void		Обновляет состояние оружия. Если была отдана команда стрелять и оружие заряжено -- стреляет

Таблица 3.9

Описание полей, методов и свойств класса Character.Actions.ShootSemiautoAction

Поля класса Character.Actions.ShootSemiautoAction				
Имя	Модификатор доступа	Тип	Назначение	
needShoot	private	bool	Переменная показывающая, нужно ли производить выстрел	
Методы класса Character.Actions.ShootSemiautoAction				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnStartDoing	public	void		Начинает стрельбу из оружия
OnStopDoing	public	void		Прекращает стрельбу из оружия
LateUpdate	private	void		Обновляет состояние оружия. Если была отдана команда стрелять и оружие заряжено -- стреляет

Таблица 3.10

Описание полей, методов и свойств класса Character.CameraFollower

Поля класса Character.CameraFollower				
Имя	Модификатор доступа	Тип	Назначение	
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

character	public	GameObject	Игрок, за которым должна следовать камера	
yLevel	public	float	Высота относительно игрока, на которой должна располагаться камера	
Методы класса Character.CameraFollower				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
LateUpdate	private	void		Перемещает камеру в позицию над игроком. Автоматически вызывается Unity каждый кадр

Таблица 3.11

Описание полей, методов и свойств класса Character.CharacterAnimator

Поля класса Character.CharacterAnimator				
Имя	Модификатор доступа	Тип	Назначение	
animator	private	Animator	Ссылка на Animator у персонажа	
handAnimationBlender	private	HandAnimationBlender	Ссылка на HandAnimationBlender у персонажа	
Idle	private	int	Константа с хэшем от idle	
Speed	private	int	Константа с хэшем от speed	
Push	private	int	Константа с хэшем от push	
RotationSpeed	private	int	Константа с хэшем от rotationSpeed	
_animationState	private	PlayerAnimationState	Переменная для хранения состояния анимации персонажа	
Методы класса Character.CharacterAnimator				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
SetIdle	public	void	bool idle	Управляет анимацией стояния
SetSpeed	public	void	float speed	Передаёт в аниматор скорость персонажа

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

SetPush	public	void		Показывает анимацию толкания
SetRotationSpeed	public	void	float rotationSpeed	Передает в аниматор скорость поворота игрока

Таблица 3.12

Описание полей, методов и свойств класса Character.CharacterController

Поля класса Character.CharacterController				
Имя	Модификатор доступа	Тип	Назначение	
target	public	GameObject	Объект, которым управляет данный контроллер	
motionController	protected	MotionController	Ссылка на motionController у управляемого персонажа	
actionController	protected	ActionController	Ссылка на actionController у управляемого персонажа	
Методы класса Character.CharacterController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	protected	void		Инициализирует переменные

Таблица 3.13

Описание полей, методов и свойств класса Character.Guns.BombGun

Поля класса Character.Guns.BombGun				
Имя	Модификатор доступа	Тип	Назначение	
_fullReloadTime	public	float	Время перезарядки магазина	
_reloadTime	public	float	Время перезарядки между выстрелами	
_bulletsInMagazine	public	int	Количество выстрелов в одном боекомплекте	
GetFullReloadTime()	public	float	Возвращает время перезарядки магазина	
GetReloadTime()	public	float	Возвращает время перезарядки между выстрелами	
GetBulletsInMagazine()	public	int	Возвращает количество выстрелов в одном боекомплекте	
Методы класса Character.Guns.BombGun				
Имя	Модификатор доступа	Тип	Аргументы	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### RU.17701729.04.01-01 81 01-1

DoShoot	protected	void		Производит выстрел из гранатомета
Run	public	void		Создает гранатомет на игровом поле

Таблица 3.14

Описание полей, методов и свойств класса Character.Guns.GunController<T>

Поля класса Character.Guns.GunController<T>				
Имя	Модификатор доступа	Тип	Назначение	
gun	public	T		
picked	private	float	Время, когда была последний раз проведена попытка подобрать оружие. Нужно для предотвращения спама командами подобрать оружие	
Методы класса Character.Guns.GunController<T>				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnTriggerEnter	private	void	Collider other	Автоматически вызывается Unity при столкновении с другими объектами

Таблица 3.15

Описание полей, методов и свойств интерфейса Character.Guns.IGun

Свойства интерфейса Character.Guns.IGun					
Имя	Модификатор доступа	Тип	Назначение		
state	private get; private set	GunState	Состояние оружия		
Методы интерфейса Character.Guns.IGun					
Имя	Модификатор доступа	Тип	Аргументы	Назначение	
OnPickedUp	private	void	GameObject player	Обработчик. Вызывается при подборе игроком оружия	
OnDropped	private	void		Обработчик. Вызывается	
Изм.		Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1					
Инв. № подл.		Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

				при потере игроком оружия
--	--	--	--	---------------------------------

Таблица 3.16

Описание полей, методов и свойств класса Character.Guns.Pistol

Поля класса Character.Guns.Pistol				
Имя	Модификатор доступа	Тип	Назначение	
_fullReloadTime	public	float	Время перезарядки магазина	
_reloadTime	public	float	Время перезарядки между выстрелами	
_bulletsInMagazine	public	int	Количество выстрелов в одном боекомплекте	
damage	public	float	Урон от выстрела	
GetFullReloadTime()	public	float	Возвращает время перезарядки магазина	
GetReloadTime()	public	float	Возвращает время перезарядки между выстрелами	
GetBulletsInMagazine()	public	int	Возвращает количество выстрелов в одном боекомплекте	
Методы класса Character.Guns.Pistol				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
DoShoot	protected	void		Производит выстрел из пистолета
Run	public	void		Создает пистолет на игровом поле

Таблица 3.17

Описание полей, методов и свойств класса Character.Guns.ReloadGun

Поля класса Character.Guns.ReloadGun			
Имя	Модификатор доступа	Тип	Назначение
_state	public	int	Состояние оружия в виде целого числа (нужно для сериализации)
position	public	Vector3	Позиция на игровом поле, на котором должно появиться оружие
id	public	int	id оружия
_bulletsCount	public	int	Переменная для хранения bulletsCount

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

_magazinesCount	public	int	Переенная для хранения magazinesCount	
player	protected	MotionController	Ссылка на MotionController игрока, который держит оружие	
IsEmpty()	public	bool	Проверяет, остались ли патроны или магазины в оружии	
needTime	private	float	Время, которое осталось до перезарядки патрона или магазина	
Методы класса Character.Guns.ReloadGun				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
GetFullReloadTime	public	float		Возвращает время перезарядки магазина
GetReloadTime	public	float		Возвращает время перезарядки между выстрелами
GetBulletsInMagazine	public	int		Возвращает количество выстрелов в одном боекомплекте
ReloadingGun	public	конструктор		Конструктор оружия, которое может перезаряжаться. Инициализирует переменные
OnPickedUp	public	void	GameObject player	Обрабатывает событие, когда игрок подбирает оружие
Spawn	private	void		Создает на игровом поле данное оружие
OnDropped	public	void		Обрабатывает выбрасывание игроком оружия
Update	public	void	float dt	Обновляет состояние оружия. Перезаряжает патроны в оружии
DoShoot	protected	void		Выпускает пулю из оружия

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# **RU.17701729.04.01-01 81 01-1**

Shoot	public	void		Производит выстрел из оружия
SetReloadBullet	private	void		Устанавливает состояние на перезарядку патрона
SetReloadMagazine	public	void		Устанавливает состояние на перезарядку магазина

Таблица 3.18

Описание полей, методов и свойств класса Character.Guns.SemiautoGun

Поля класса Character.Guns.SemiautoGun				
Имя	Модификатор доступа	Тип	Назначение	
_fullReloadTime	public	float	Время перезарядки магазина	
_reloadTime	public	float	Время перезарядки магазина	
_bulletsInMagazine	public	int	Количество выстрелов в одном боекомплекте	
damage	public	float	Урон от выстрела	
accuracy	public	float	Точность (вычисляется по Гауссу)	
GetFullReloadTime()	public	float	Возвращает время перезарядки магазина	
GetReloadTime()	public	float	Возвращает время перезарядки между выстрелами	
GetBulletsInMagazine()	public	int	Возвращает количество выстрелов в одном боекомплекте	
Методы класса Character.Guns.SemiautoGun				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
DoShoot	protected	void		Производит выстрел из автомата
Run	public	void		Создает автомат на игровом поле

Таблица 3.19

Описание полей, методов и свойств класса Character.Guns.ShootSystem

Поля класса Character.Guns.ShootSystem			
Имя	Модификатор доступа	Тип	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

_raycastHit	private	RaycastHit	Переменная для хранения результата от попадания пули	
Методы класса Character.Guns.ShootSystem				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
GetGunPosition	public	Vector3	Vector3 characterPosition	Находит раположение оружия у игрока
DrawTracer	public	void	Vector3 start, Vector3 stop	Рисует трассер от выстрела
SimpleRaycast	public	bool	Transform transform, Quaternion rotation, Vector3 directionDelta, out RaycastHit raycastRes, out ICommand command	Выпускает пулю из заданной позиции и проверяет в какой объект она попала Отрисовывает выстрел.
ShootWithDamage	public	bool	GameObject gameObject, Quaternion rotation, Vector3 directionDelta, float damage	Производит выстрел с уроном
RandomDelta	public	Vector3	double sigma	Создает случайное отклонение на основе распределения Гаусса
ShootWithBomb	public	void	GameObject gameObject, Vector3 target, string bombPrefab	Производит выстрел бомбой

Таблица 3.20

Описание полей, методов и свойств класса Character.Guns.ShotGun

Поля класса Character.Guns.ShotGun			
Имя	Модификатор доступа	Тип	Назначение
_fullReloadTime	public	float	Время перезарядки магазина
_reloadTime	public	float	Время перезарядки магазина
_bulletsInMagazine	public	int	Количество выстрелов в одном боекомплекте

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

damage	public	float	Урон от выстрела	
shootsCount	public	int	Количество дробы, вылетающее за один выстрел	
accuracy	public	float	Точность (вычисляется по Гауссу)	
GetFullReloadTime()	public	float	Возвращает время перезарядки магазина	
GetReloadTime()	public	float	Возвращает время перезарядки между выстрелами	
GetBulletsInMagazine()	public	int	Возвращает количество выстрелов в одном боекомплекте	
Методы класса Character.Guns.ShotGun				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
DoShoot	protected	void		Производит выстрел из дробовика
Run	public	void		Создает дробовик на игровом поле

Таблица 3.21

Описание полей, методов и свойств класса Character.HP.DamageSource

Методы класса Character.HP.DamageSource				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
None	public	int		Возвращает числовое значение, соответствующее отсутствию источнику урона
InstaKill	public	int		Возвращает числовое значение, соответствующее бесконечно сильному источнику урона
Player	public	int	int id	Возвращает числовое значение, соответствующее игроку, который является

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

				источником урона
Player	public	int	GameObject gameObject	Возвращает числовое значение, соответствующее источнику урона от игрока
Bomb	public	int		Возвращает числовое значение, соответствующее источнику урона от бомбы
GetSourceGO	public	GameObject	int damageSource	Получает объект, от которого был получен урон

Таблица 3.22

Описание полей, методов и свойств структуры Character.HP.HPChange

Поля структуры Character.HP.HPChange				
Имя	Модификатор доступа	Тип	Назначение	
delta	public	float	Дельта, на которую изменилось здоровье	
source	public	int	Источник изменения здоровья	
Методы структуры Character.HP.HPChange				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
HPChange	public	конструктор	float delta, int source	Конструктор изменения здоровья

Таблица 3.23

Описание полей, методов и свойств класса Character.HP.HPController

Поля класса Character.HP.HPController				
Имя	Модификатор доступа	Тип	Назначение	
MaxHP	public	float	Максимальное количество здоровья у объекта	
HPAnimationSpeed	public	float	Скорость анимирования изменения здоровья у объекта	
hpImage	public	Image	Изображение, на котором рисуется полоска со здоровьем	
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

dead	private	bool	Переменная, хранящая был ли убит данный объект	
currentHp	public	float	Текущее количество здоровья	
_hpOnBar	private	float	Переменная для хранения hpOnBar	
Методы класса Character.HP.HPController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
Update	private	void		Анимирует полосу здоровья. Автоматически вызывается Unity каждый кадр
TakeDamage	public	float	float damage, int source, bool autoSendChange	Наносит урон по здоровью данного объекта
_applyHpChange	public	void	HPChange hpChange	Применяет изменения здоровья
ApplyHPChange	public	void	GameObject target, HPChange hpChange	Применяет изменение здоровья к объекту

Таблица 3.24

Описание полей, методов и свойств класса Character.MotionController

Поля класса Character.MotionController			
Имя	Модификатор доступа	Тип	Назначение
rigidbody	private	Rigidbody	Ссылка на Rigidbody персонажа
capsuleCollider	private	CapsuleCollider	Ссылка на CapsuleCollider персонажа
animator	private	CharacterAnimator	Ссылка на CharacterAnimator персонажа
moveForce	public	float	Сила, которая изменяет скорость персонажа
speed	public	float	Максимальная скорость персонажа
rotationSpeed	public	float	Скорость поворота персонажа
groundCollisions	private	List<GameObject>	
isGrounded	private	bool	Находится ли персонаж на земле
Свойства класса Character.MotionController			

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Имя	Модификатор доступа	Тип	Назначение	
TargetDirection	public get; public set	Vector3	Направление, в котором должен двигаться персонаж	
TargetRotation	public get; public set	Vector3	Направление, в котором смотрит персонаж	
Методы класса Character.MotionController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
DeGround	private	void	GameObject gameObject	Провряет, что персонаж всё ещё стоит на земле.
OnTriggerStay	private	void	Collider other	Сообщает, что персонаж всё ещё стоит на данном коллайдере. Автоматически вызывается Unity
OnTriggerExit	private	void	Collider other	Сообщает, что персонаж больше не стоит на данном коллайдере . Автоматически вызывается Unity
FixedUpdate	private	void		Производит передвижение персонажа. Автоматически вызывается Unity при каждой обработке физики
OnDrawGizmos	private	void		Отрисовывает отладочную информацию о состоянии игрока. Автоматически вызывается средой Unity

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Таблица 3.25

Описание полей, методов и свойств структуры Character.PlayerAnimationState

Поля структуры Character.PlayerAnimationState			
Имя	Модификатор доступа	Тип	Назначение
idle	public	bool	Должна ли сейчас быть анимация стояния на месте
speed	public	float	Скорость персонажа
rotationSpeed	public	float	Скорость поворота персонажа

Таблица 3.26

Описание полей, методов и свойств класса Client

Поля класса Client				
Имя	Модификатор доступа	Тип	Назначение	
bulletTrailRenderer	public	BulletTrailRenderer	Рендерер следов от пуль	
spawnBorder	public	GameObject	Объект, хранящий границу внутри которой можно создавать объекты	
spawnPolygon	public	TrianglePolygon	Граница игрового поля	
mainPlayerObj	public	GameObject	Главный персонаж	
cameraObj	public	GameObject	Камера	
GameObject>	private	Dictionary<string,	Словарь префабов, которые можно создавать на игровом поле	
prefabsList	public	List<GameObject>	Список префабов, которые можно создавать на игровом поле	
SpawnPrefab(string	public	GameObject	Создаёт объект на игровом поле	
Свойства класса Client				
Имя	Модификатор доступа	Тип	Назначение	
client	private get; private set	Client	Статическая ссылка на Client (синглтон)	
Методы класса Client				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Awake	private	void		Инициализирует переменные
RemoveObject	public	void	GameObject gameObject	Удаляет объект с игрового поля
SpawnObject	public	GameObject	SpawnPrefabCommand command	Создаёт объект на игровом поле

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Таблица 3.27

Описание полей, методов и свойств класса Coin

Поля класса Coin				
Имя	Модификатор доступа	Тип	Назначение	
picked	private	float	Время, когда последний раз была отправлена команда подобрать монетку на сервер	
Методы класса Coin				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnTriggerEnter	private	void	Collider other	Подбирает монетку при столкновении с игроком. Автоматически вызывается Unity

Таблица 3.28

Описание полей, методов и свойств класса CommandsSystem.Commands.AddOrChangeInstance

Поля класса CommandsSystem.Commands.AddOrChangeInstance				
Имя	Модификатор доступа	Тип	Назначение	
instance	public	Instance	Изменившийся Instance	
Методы класса CommandsSystem.Commands.AddOrChangeInstance				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Применяет изменения

Таблица 3.29

Описание полей, методов и свойств класса CommandsSystem.Commands.AddPlayerToGame

Поля класса CommandsSystem.Commands.AddPlayerToGame				
Имя	Модификатор доступа	Тип	Назначение	
player	public	Player	Добавляемый игрок	
Методы класса CommandsSystem.Commands.AddPlayerToGame				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Добавляет игрока в игру

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# **RU.17701729.04.01-01 81 01-1**

Таблица 3.30

Описание полей, методов и свойств класса `CommandsSystem.Commands.ApplyForceCommand`

Поля класса CommandsSystem.Commands.ApplyForceCommand				
Имя	Модификатор доступа	Тип	Назначение	
objectId	public	int	Id игрового объекта, к которому нужно применить силу	
force	public	Vector3	Сила	
Методы класса CommandsSystem.Commands.ApplyForceCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
ApplyForceCommand	public	конструктор	GameObject gameObject, Vector3 force	Конструктор команды
this	private	конструктор	ObjectID.GetID	
Run	public	void		Применяет силу, если объект обрабатывается на данном клиенте

Таблица 3.31

Описание полей, методов и свойств класса `CommandsSystem.Commands.ChangeHPCommand`

Поля класса CommandsSystem.Commands.ChangeHPCommand				
Имя	Модификатор доступа	Тип	Назначение	
id	public	int	id объекта, у которого нужно изменить здоровье	
HpChange	public	HPChange	Изменение здоровья	
Методы класса CommandsSystem.Commands.ChangeHPCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Применяет изменение здоровья

Таблица 3.32

Описание полей, методов и свойств класса `CommandsSystem.Commands.ChangePlayerProperty`

Поля класса <code>CommandsSystem.Commands.ChangePlayerProperty</code>				
Имя	Модификатор доступа	Тип	Назначение	
<code>property</code>	<code>public</code>	<code>PlayerProperty</code>	Состояние игрока	
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

deltaTime	public	float	Время, которое прошло с последнего изменения состояния	
Методы класса CommandsSystem.Commands.ChangePlayerProperty				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Применяет изменение состояния

Таблица 3.33

Описание полей, методов и свойств класса CommandsSystem.Commands.ChangePlayerScore

Поля класса CommandsSystem.Commands.ChangePlayerScore				
Имя	Модификатор доступа	Тип	Назначение	
player	public	int	Игрок, у которого нужно изменить очки	
Score	public	int	Новое количество очков	
Методы класса CommandsSystem.Commands.ChangePlayerScore				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Применяет изменения

Таблица 3.34

Описание полей, методов и свойств класса

CommandsSystem.Commands.CreateChatMessageCommand

Поля класса CommandsSystem.Commands.CreateChatMessageCommand				
Имя	Модификатор доступа	Тип	Назначение	
playerid	public	int	Игрок, отправивший сообщение	
message	public	string	Сообщение	
Методы класса CommandsSystem.Commands.CreateChatMessageCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Отображает сообщение в чате

Таблица 3.35

Описание полей, методов и свойств класса

CommandsSystem.Commands.DrawPositionTracerCommand

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Поля класса CommandsSystem.Commands.DrawPositionTracerCommand				
Имя	Модификатор доступа	Тип	Назначение	
player	public	int	id персонажа, выпустивший пулю	
target	public	Vector3	Координата, в которую попала пуля	
Методы класса CommandsSystem.Commands.DrawPositionTracerCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Выполняет команду

Таблица 3.36

Описание полей, методов и свойств класса

CommandsSystem.Commands.DrawTargetedTracerCommand

Поля класса CommandsSystem.Commands.DrawTargetedTracerCommand				
Имя	Модификатор доступа	Тип	Назначение	
player	public	int	Id персонажа, выпустившего пулю	
target	public	int	Id персонажа, в которого попала пуля	
HpChange	public	HPChange	Изменение здоровья персонажа, в которого попала пуля	
Методы класса CommandsSystem.Commands.DrawTargetedTracerCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Выполняет команду

Таблица 3.37

Описание полей, методов и свойств класса CommandsSystem.Commands.ExplodeBombCommand

Поля класса CommandsSystem.Commands.ExplodeBombCommand				
Имя	Модификатор доступа	Тип	Назначение	
bombId	public	int	id бомбы, которую нужно взорвать	
Методы класса CommandsSystem.Commands.ExplodeBombCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Взрывает указанную бомбу

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Таблица 3.38

Описание полей, методов и свойств класса `CommandsSystem.Commands.PickCoinCommand`

Поля класса CommandsSystem.Commands.PickCoinCommand				
Имя	Модификатор доступа	Тип	Назначение	
player	public	int	Id игрока, подобравшего монетку	
coin	public	int	Id подобранной монетки	
Методы класса CommandsSystem.Commands.PickCoinCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Подбирает монетку

Таблица 3.39

Описание полей, методов и свойств класса `CommandsSystem.Commands.PickUpGunCommand`

Поля класса CommandsSystem.Commands.PickUpGunCommand				
Имя	Модификатор доступа	Тип	Назначение	
player	public	int	ID персонажа	
gun	public	int	ID оружия	
Методы класса CommandsSystem.Commands.PickUpGunCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Подбирает оружие

Таблица 3.40

Описание полей, методов и свойств класса `CommandsSystem.Commands.PlayerProperty`

Поля класса CommandsSystem.Commands.PlayerProperty				
Имя	Модификатор доступа	Тип	Назначение	
id	public	int	ID персонажа	
position	public	Vector3	Позиция персонажа	
rotation	public	Quaternion	Поворот персонажа	
animationState	public	PlayerAnimationState	Состояние анимации персонажа	
characterAnimator	private	CharacterAnimator	Ссылка на CharacterAnimator персонажа	
Методы класса CommandsSystem.Commands.PlayerProperty				
Имя	Модификатор доступа	Тип	Аргументы	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

CopyFrom	public	void	PlayerProperty state	Копирует состояние из другого
FromGameObject	public	void	GameObject gameObject	Получает состояние из объекта
ApplyToObject	public	void	GameObject gameObject	Применяет состояние к объекту
CreateChangedCommand	public	ICommand	float deltaTime	Создаёт команду для отправки состояния другим клиентам
Interpolate	public	void	PlayerProperty lastLastState, PlayerProperty lastState, PlayerProperty nextState, float coef	Интерполирует состояние между другими

Таблица 3.41

Описание полей, методов и свойств класса `CommandsSystem.Commands.PlayerPushCommand`

Поля класса CommandsSystem.Commands.PlayerPushCommand				
Имя	Модификатор доступа	Тип	Назначение	
playerId	public	int	id игрока, для которого нужно показать анимацию	
Методы класса CommandsSystem.Commands.PlayerPushCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Показывает анимацию толкания для соответствующего игрока

Таблица 3.42

Описание полей, методов и свойств класса  
`CommandsSystem.Commands.SetAfterShowResultsCommand`

Поля класса <code>CommandsSystem.Commands.SetAfterShowResultsCommand</code>				
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Имя	Модификатор доступа	Тип	Назначение	
–	public	int	Переменная для корректной работы сериализации	
Методы класса CommandsSystem.Commands.SetAfterShowResultsCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Изменяет состояние GameManager на соответствующее команде

Таблица 3.43

Описание полей, методов и свойств класса **CommandsSystem.Commands.SetGameMode**

Поля класса CommandsSystem.Commands.SetGameMode				
Имя	Модификатор доступа	Тип	Назначение	
gamemodeCode	public	int	Код игрового режима	
roomId	public	int	Номер комнаты, в которой будет проводиться данный игровой режим	
currentGameNum	public	int	Номер игры по порядку	
Методы класса CommandsSystem.Commands.SetGameMode				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Загружает заданный игровой режим

Таблица 3.44

Описание полей, методов и свойств класса **CommandsSystem.Commands.SetPlatformStateCommand**

Поля класса CommandsSystem.Commands.SetPlatformStateCommand				
Имя	Модификатор доступа	Тип	Назначение	
id	public	int	Id подвижной платформы	
direction	public	int	Направление, в котором должна двигаться платформа сейчас	
Методы класса CommandsSystem.Commands.SetPlatformStateCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Синхронизирует состояние

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

				платформы с заданным
--	--	--	--	----------------------

Таблица 3.45

Описание полей, методов и свойств класса

CommandsSystem.Commands.SpawnParabolaFlyingCommand

Поля класса CommandsSystem.Commands.SpawnParabolaFlyingCommand				
Имя	Модификатор доступа	Тип	Назначение	
command	public	SpawnPrefabCommand	Команда для создания объекта	
medium	public	Vector3	Среднее положение объекта	
target	public	Vector3	Конечное положение объекта	
totalTime	public	float	Время полёта объекта	
Методы класса CommandsSystem.Commands.SpawnParabolaFlyingCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Создает на игровом поле объект с хаданными параметрами

Таблица 3.46

Описание полей, методов и свойств класса CommandsSystem.Commands.SpawnPlayerCommand

Поля класса CommandsSystem.Commands.SpawnPlayerCommand				
Имя	Модификатор доступа	Тип	Назначение	
command	public	SpawnPrefabCommand	Базовая команда для создания объекта	
playerId	public	int	Id игрока, который будет управлять данным персонажем	
Методы класса CommandsSystem.Commands.SpawnPlayerCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Создает персонажа с заданными параметрами

Таблица 3.47

Описание полей, методов и свойств класса CommandsSystem.Commands.SpawnPrefabCommand

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Поля класса CommandsSystem.Commands.SpawnPrefabCommand				
Имя	Модификатор доступа	Тип	Назначение	
prefabName	public	string	Название префаба, в которой нужно создать	
position	public	Vector3	Позиция, в которой нужно создать объект	
rotation	public	Quaternion	Поворот, на который должен быть развернут объект	
id	public	int	Id объекта	
owner	public	int	Владелец объекта	
creator	public	int	Игрок, создавший объект (если объект создан не игроком следует указать -1)	
random	private	System.Random	Игрок, создавший объект (если объект создан не игроком следует указать -1)	
Методы класса CommandsSystem.Commands.SpawnPrefabCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Создает объект с заданными параметрами

Таблица 3.48

Описание полей, методов и свойств класса CommandsSystem.Commands.StartGameCommand

Поля класса CommandsSystem.Commands.StartGameCommand				
Имя	Модификатор доступа	Тип	Назначение	
_	public	int		
Методы класса CommandsSystem.Commands.StartGameCommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Начинает игру

Таблица 3.49

Описание полей, методов и свойств класса CommandsSystem.Commands.TakeOwnCommand

Поля класса CommandsSystem.Commands.TakeOwnCommand			
Имя	Модификатор доступа	Тип	Назначение
objectId	public	int	Id объекта, у которого меняется владелец
owner	public	int	Id нового владельца объекта
Методы класса CommandsSystem.Commands.TakeOwnCommand			

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# RU.17701729.04.01-01 81 01-1

Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	public	void		Применяет изменения владельца

Таблица 3.50

Описание полей, методов и свойств класса CommandsSystem.CommandsSystem

Поля класса CommandsSystem.CommandsSystem				
Имя	Модификатор доступа	Тип	Назначение	
_stream	private	MemoryStream	MemoryStream для внутреннего использования (нужен для уменьшения нагрузки на сборщик мусора)	
_writer	private	BinaryWriter	BinaryWriter для внутреннего использования (нужен для уменьшения нагрузки на сборщик мусора)	
Методы класса CommandsSystem.CommandsSystem				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
EncodeCommand<T>	private	void	T command, Stream stream	Кодирует заданную команду в бинарный вид и записывает в stream. Данный метод использует кодогенерацию
ResetWriteStreams	private	void		Очищает потоки для записи команд от данных
EncodeSimpleCommand<T>	public	byte[]	T command, int room, MessageFlags flags	Кодирует обычную команду
EncodeUniqCommand<T>	public	byte[]	T command, int room, int code1, int code2, MessageFlags flags	Кодирует уникальную команду

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

EncodeAskMessage	public	byte[]	int room, int firstIndex, int lastIndex, MessageFlags flags	Кодирует команду запроса сообщений с сервера
EncodeJoinGameRoomMessage	public	byte[]	int room, MessageFlags flags	Кодирует команду присоединения к игровой комнате
EncodeLeaveGameRoomMessage	public	byte[]	int room, MessageFlags flags	Кодирует команду покидания игровой комнаты
EncodeJsonMessage	public	byte[]	string json, int room, MessageFlags flags	Кодирует JSON-сообщение на сервер
DecodeCommand	public	ICommand	byte[] array, out int num, out int room	Декодирует команду с сервера В данном методе используется кодогенерация

Таблица 3.51

Описание полей, методов и свойств интерфейса CommandsSystem.ICommand

Методы интерфейса CommandsSystem.ICommand				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Run	private	void		Выполняет команду

Таблица 3.52

Описание полей, методов и свойств класса DefaultNamespace.AutoHideOnStart

Методы класса DefaultNamespace.AutoHideOnStart				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	public	void		

Таблица 3.53

Описание полей, методов и свойств класса Editor.ClientEditor

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Методы класса Editor.ClientEditor				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnInspectorGUI	public	void		Выводит отладочную информацию в инспектор Unity

Таблица 3.54

Описание полей, методов и свойств класса Events.EventsManager

Поля класса Events.EventsManager				
Имя	Модификатор доступа	Тип	Назначение	
handler	public	EventsHandler	Текущий обработчик событий	
Методы класса Events.EventsManager				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Awake	private	void		Инициализирует переменные. Автоматически вызывается Unity
Start	private	void		Устанавливает обработчики событий

Таблица 3.55

Описание полей, методов и свойств класса Game.MatchInfo

Поля класса Game.MatchInfo				
Имя	Модификатор доступа	Тип	Назначение	
roomId	public	int	id комнаты, в которой проводится матч	
name	public	string	Имя матча	
maxPlayersCount	public	int	Максимальное количество игроков в матче	
players	public	List<string>	Список игроков в матче	
state	public	int	Текущее состояние матча. 0 означает, что матч ещё не начался. 1 -- начался.	
Методы класса Game.MatchInfo				
Имя	Модификатор доступа	Тип	Аргументы	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

MatchInfo	public	конструктор	string name, int roomid, int maxPlayersCount, List<string> players, int state	Конструктор информации о матче
ToJson	public	JsonValue		Записывает информацию о матче в формате JSON
FromJson	public	MatchInfo	JsonValue json	Создаёт информацию о матче из JSON-объекта

Таблица 3.56

Описание полей, методов и свойств класса Game.MatchesManager

Поля класса Game.MatchesManager				
Имя	Модификатор доступа	Тип	Назначение	
_state	private	STATE	Переменная для хранения состояния менеджера матчей	
currentMatch	public	MatchInfo	Информация о текущем матче	
Методы класса Game.MatchesManager				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
CreateMatch	private	void	MatchInfo matchInfo	Создаёт матч с заданными параметрами
JoinMatch	private	void	int matchid	Присоединяется к заданному матчу
GetMatchesList	private	void		Получает список матчей и автоматически присоединяется к одному из возможных
SendStartGame	public	void		Посылает сообщение о старте матча
SetMatchIsPlaying	public	void		Устанавливает локальное состояние матча на PLAYING

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

HandleJsonMatchChanged	public	void	JsonValue json	Обработывает JSON-сообщение, что текущий матч изменился
Update	public	void		Обновляет состояние менеджера матчей
Reset	public	void		

Таблица 3.57

Описание полей, методов и свойств класса GameMechanics.Bomb

Поля класса GameMechanics.Bomb				
Имя	Модификатор доступа	Тип	Назначение	
damage	public	float	Урон от попадания	
explosionForce	public	float	Сила взрыва	
area	public	SphereCollider	Сфера, внутри которой взрывается бобма	
noDamageToCreator	public	bool	Должна ли бомба наносить урон игроку, который её создал (false, если должна)	
radius	public	float	Радиус взрыва	
Методы класса GameMechanics.Bomb				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	public	void		Инициализирует переменные
Explode	public	void		Синхронно взрывает бомбу
RealExplode	public	void		Взрывает бомбу локально

Таблица 3.58

Описание полей, методов и свойств класса GameMechanics.BombOnCollisionExploder

Методы класса GameMechanics.BombOnCollisionExploder				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	public	void		Настраивает физику, чтобы бомба не сталкивалась с

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# RU.17701729.04.01-01 81 01-1

				персонажем, создавшем её
OnCollisionEnter	public	void	Collision other	Взрывает бомбу. Автоматически вызывается Unity при столкновении с другим объектом

Таблица 3.59

Описание полей, методов и свойств класса GameMechanics.BombTriggerHPExploder

Методы класса GameMechanics.BombTriggerHPExploder				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnTriggerEnter	private	void	Collider other	Взрывает бомбу, если объект имеет здоровье. Автоматически вызывается Unity при столкновении

Таблица 3.60

Описание полей, методов и свойств класса GameMechanics.MovingPlatform

Поля класса GameMechanics.MovingPlatform					
Имя	Модификатор доступа	Тип	Назначение		
nextTransform	public	Transform	Позиция, между которой должна перемещаться платформа		
lastPosition	private	Vector3	Предыдущая позиция платформы		
nextPosition	private	Vector3	Следующая позиция		
speed	public	float	Скорость платформы		
stayTime	public	float	Время, на которое платформа останавливается		
state	public	int	Состояние платформы		
int	private	const	Состояние платформы, при котором она движется		
int	private	const	Состояние платформы, при котором она стоит на месте		
int	private	const	Состояние плафтформы, при котором она ожидает, когда ей дадут команду передвигаться		
Изм.		Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1					
Инв. № подл.		Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

direction	public	int	Направление, в котором движется платформа	
int	private	const	Направление от предыдущего к следующему	
int	private	const	Направление от следующего к предыдущему	
id	private	int	ID платформы	
currentStayingTime	private	float	Время, которое платформе осталось стоять на месте	
Методы класса GameMechanics.MovingPlatform				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
OnCollisionEnter	private	void	Collision other	Обработывает событие, когда какой-либо объект попадает на платформу. Автоматически вызывается Unity при столкновении
OnCollisionExit	private	void	Collision other	Обработывает событие, когда объект уходит с платформы. Автоматически вызывается Unity
SetMoveState	public	void	int direction	Переключает плафторму в режим движения
Update	public	void		Обновляет состояние платформы и перемещает её
HandleOwnTaken	public	void	int owner	Обработывает событие, когда у платформы появляется новый владелец

Таблица 3.61

Описание полей, методов и свойств класса GameMechanics.ParabolaFlyingObject

Поля класса GameMechanics.ParabolaFlyingObject

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# RU.17701729.04.01-01 81 01-1

Имя	Модификатор доступа	Тип	Назначение	
start	public	Vector3	Стартовая позиция	
medium	public	Vector3	Средняя позиция	
stop	public	Vector3	Конечная позиция	
totalTime	public	float	Время полёта	
startTime	public	float	Время, когда начался полёт	
Методы класса GameMechanics.ParabolaFlyingObject				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	public	void		Инициализирует переменные
OnCollisionEnter	private	void	Collision other	Автоматически прекращает движение при столкновении объектом
Update	public	void		Перемещает объект. Автоматически вызывается Unity каждый кадр

Таблица 3.62

Описание полей, методов и свойств класса GameMode.GameManager

Поля класса GameMode.GameManager				
Имя	Модификатор доступа	Тип	Назначение	
int	private	const	Количество игр, которое нужно сыграть	
sceneReloaded	public	bool	Переменная, хранящая была ли загружена новая сцена на предыдущем кадре	
gamesCount	public	int	Количество сыгранных игр	
_state	private	STATE	Переменная для хранения состояния игры	
gameMode	public	IGameMode	Текущий игровой режим	
timeEnd	public	float	Время, когда должен закончиться игровой режим	
showResultsWaitTime	private	float	Время, которое осталось до конца показа результатов	
availableGameModes	private	List<int>	Игровые режимы, которые можно запустить	
Методы класса GameMode.GameManager				
Имя	Модификатор доступа	Тип	Аргументы	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



**RU.17701729.04.01-01 81 01-1**

SetGameMode	public	void	int gamemodeCode, int roomId, int currentGameNum	Изменяет игровой режим
SetAfterShowResults	public	void		Перестает показывать результаты игры
Reset	public	void		Сбрасывает состояние переменных
ChooseGameMode	private	int		Выбирает один из возможных режимов
Update	public	void		Обновляет состояние игры

Таблица 3.63

Описание полей, методов и свойств класса GameMode.GameModeFunctions

Поля класса GameMode.GameModeFunctions				
Имя	Модификатор доступа	Тип	Назначение	
_raycastHitInfo	private	RaycastHit	RaycastHit для внутреннего использования (нужен, чтобы уменьшить нагрузку на сборщик мусора)	
Методы класса GameMode.GameModeFunctions				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SpawnPlayer	public	void	int playerId	Создает персонажа в случайном месте
SpawnPlayers	public	void		Создаёт персонажа для каждого из игроков
FindPlaceForSpawn	public	Vector3	float height, float radius	Ищет место для создания объекта заданного размера

Таблица 3.64

Описание полей, методов и свойств интерфейса GameMode.IGameMode

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Свойства интерфейса GameMode.IGameMode				
Имя	Модификатор доступа	Тип	Назначение	
TimeLength	private get; private set	float	Время в секундах, которое длится игровой режим	
Методы интерфейса GameMode.IGameMode				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Update	private	bool		Обновляет состояние игрового режима
Stop	private	bool		Завершает выполнение игрового режима

Таблица 3.65

Описание полей, методов и свойств класса GameMode.Instance

Поля класса GameMode.Instance				
Имя	Модификатор доступа	Тип	Назначение	
id	public	int	id данного instance	
name	public	string	Отображаемое имя у данного instance	
currentLoadedGamemodeNum	public	int	Текущий загруженный игровой режим	
Методы класса GameMode.Instance				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Instance	public	конструктор		Конструктор для сериализации
Instance	public	конструктор	int id	Конструктор
Send	public	void		Отправляет информацию об instance другим игрокам

Таблица 3.66

Описание полей, методов и свойств класса GameMode.InstanceManager

Поля класса GameMode.InstanceManager
--------------------------------------

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Имя	Модификатор доступа	Тип	Назначение	
instances	public	List<Instance>	Список instance в текущей игре	
currentInstance	public	Instance	Текущий instance	
ID	public	int	Id текущего instance	
Методы класса GameMode.InstanceManager				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Init	public	void		Инициализирует переменные
Reset	public	void		Сбрасывает состояние переменных

Таблица 3.67

Описание полей, методов и свойств класса GameMode.PickCoinsGameMode

Поля класса GameMode.PickCoinsGameMode				
Имя	Модификатор доступа	Тип	Назначение	
state	private	STATE	Текущее состояние режима	
coinsCount	public	int	Количество созданных монет	
TimeLength	public	float	Время в секундах, которое длится игровой режим	
Методы класса GameMode.PickCoinsGameMode				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SpawnRandomCoin	public	void	int num	Создает монету в случайном месте
Update	public	bool		Обновляет состояние игрового режима
Stop	public	bool		Завершает выполнение игрового режима

Таблица 3.68

Описание полей, методов и свойств класса GameMode.Player

Поля класса GameMode.Player			
Имя	Модификатор доступа	Тип	Назначение
id	public	int	ID игрока

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

score	public	int	Количество очков у игрока	
name	public	string	Имя игрока	
owner	public	int	Владелец игрока (id instance, который управляет игроком)	
controllerType	public	int	Кем управляется игрок. 0 -- человеком, 1 -- компьютером	
totalScore	public	int	Суммарное количество очков за все игровые режимы	
placeInLastGame	public	int	Место, занятое в последней игре	
Методы класса GameMode.Player				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Player	public	конструктор		Конструктор для сериализации
Player	public	конструктор	int id, int owner, int controllerType	Конструктор игрока
ToString	public	string		Записывает информацию об игроке в строку

Таблица 3.69

Описание полей, методов и свойств класса GameMode.PlayerStorage

Поля класса GameMode.PlayerStorage			
Имя	Модификатор доступа	Тип	Назначение
_player	private	Player	Переменная для хранения ссылки на игрока
namePanel	public	TextMeshProUGUI	Панель, на которой должно отображаться имя игрока

Таблица 3.70

Описание полей, методов и свойств класса GameMode.PlayersManager

Поля класса GameMode.PlayersManager				
Имя	Модификатор доступа	Тип	Назначение	
players	public	List<Player>	Список игроков в текущем матче	
mainPlayer	public	Player	Главный игрок, которым управляет человек	
playersCount	public	int	Количество игроков в игре	
Методы класса GameMode.PlayersManager				
Имя	Модификатор доступа	Тип	Аргументы	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

GetPlayerById	public	Player	int id	Возвращает игрока с заданным ID
AddScoreToPlayer	public	void	GameObject player, int score	Начисляет очки игроку
AddScoreToPlayer	public	void	Player player, int score	Начисляет очки игроку
IsMainPlayer	public	bool	Player player	Проверяет, является ли данный игрок главным
Reset	public	void		Сбрасывает значения переменных

Таблица 3.71

Описание полей, методов и свойств класса GameMode.ShooterGameMode

Поля класса GameMode.ShooterGameMode				
Имя	Модификатор доступа	Тип	Назначение	
state	private	STATE	Текущее состояние игрового режима	
spawnedGunsCount	private	int	Количество созданного на игровом поле оружия	
timeToSpawnNextGun	private	float	Время, через которое нужно создать следующее оружие	
TimeLength	public	float	Время в секундах, которое длится игровой режим	
Методы класса GameMode.ShooterGameMode				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SpawnRandomGun	private	void	int num	Создаёт случайное оружие на игровом поле
Update	public	bool		Обновляет состояние игрового режима
Stop	public	bool		Завершает выполнение игрового режима

Таблица 3.72

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Описание полей, методов и свойств класса HandAnimationBlender

Поля класса HandAnimationBlender				
Имя	Модификатор доступа	Тип	Назначение	
blendSpeed	public	float	Скорость переключения между анимациями	
blendStart	private	bool	Началось ли производить переключение между анимациями	
blendStop	private	bool	Началось ли производить обратное переключение между анимациями	
blendCoeff	private	float	Текущий коэффициент смешивания слоев анимации	
layerIndex	private	int	Индекс слоя с анимациями рук	
animator	private	Animator	Ссылка на Animator персонажа	
Методы класса HandAnimationBlender				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
Update	private	void		Обновляет анимацию рук (если нужно). Автоматически вызывается Unity каждый кадр
StartHandAnimation	public	void		Начинает анимацию рук. Автоматически вызывается Unity при запуске анимации
StopHandAnimation	public	void		Прекращает анимацию рук. Автоматически вызывается Unity по окончанию анимации

Таблица 3.73

Описание полей, методов и свойств интерфейса IOwnedEventHandler

Методы интерфейса IOwnedEventHandler
--------------------------------------

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Имя	Модификатор доступа	Тип	Аргументы	Назначение
HandleOwnTaken	private	void	int owner	Обработчик события, когда у объекта меняется владелец

Таблица 3.74

Описание полей, методов и свойств класса Interpolation.InterpolationFunctions

Методы класса Interpolation.InterpolationFunctions				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
CubicHermiteSpline	public	float	float start, float stop, float m0, float m1, float t	Вычисляет сплайн Эрмита
CubicHermiteSpline3	public	float	float p0, float p1, float p2, float dt01, float dt12, float t	Вычисляет сплайн Эрмита для трёх точек
Lerp3Points	public	Vector3	Vector3 p0, Vector3 p1, Vector3 p2, float dt01, float dt12, float t	Интерполирует вектор
InterpolatePosition	public	Vector3	Vector3 lastPosition, Vector3 lastPosition, Vector3 nextPosition, float coef	Интерполирует позицию между точкам
InterpolateRotation	public	Quaternion	Quaternion lastRotation, Quaternion nextRotation, float coef	Интерполирует поворот
InterpolatePlayerAnimationState	public	PlayerAnimationState	PlayerAnimationState last, PlayerAnimationState next, float coef	Интерполирует анимацию персонажа
InterpolateBool	public	bool	bool last, bool next, float coef	Интерполирует логическую переменную
InterpolateFloat	public	float	float last, float next, float coef	Интерполирует число с
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

				плавающей запятой
BezierCurve	public	float	float p0, float p1, float p2, float t	Получает значение в кривой Безье
BezierCurve	public	Vector3	Vector3 p0, Vector3 p1, Vector3 p2, float t	Получает координаты вектора в кривой Безье

Таблица 3.75

Описание полей, методов и свойств класса Interpolation.ManagedGameObject<T>

Поля класса Interpolation.ManagedGameObject<T>				
Имя	Модификатор доступа	Тип	Назначение	
lastSendState	private	float	Время, когда в последний раз был синхронизирован объект	
property	public	T	Свойство объекта	
updateTime	protected	float	Период синхронизации состояния	
Методы класса Interpolation.ManagedGameObject<T>				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	public	void		Инициализирует переменные
Update	private	void		Отправляет состояние объекта, если нужно. Автоматически вызывается Unity каждый кадр

Таблица 3.76

Описание полей, методов и свойств класса Interpolation.Managers.PlayerUnmanagedGameObject

Методы класса Interpolation.Managers.PlayerUnmanagedGameObject				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
pushEnd	public	void		Обработывает событие конца анимации. Нужен, чтобы unity не кидал warning, что событие не

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



**RU.17701729.04.01-01 81 01-1**

				было обработано
--	--	--	--	--------------------

Таблица 3.77

Описание полей, методов и свойств класса Interpolation.Properties.GameObjectProperty<T>

Методы класса Interpolation.Properties.GameObjectProperty<T>				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
CopyFrom	public	void	T state	Копирует состояние из другого
CopyFrom	public	void	IGameObjectProperty state	Копирует состояние из другого
Interpolate	public	void	T lastLastState, T lastState, T nextState, float coef	Интерполирует состояние между другими
Interpolate	public	void	IGameObjectProperty lastLastState, IGameObjectProperty lastState, IGameObjectProperty nextState, float coef	Интерполирует состояние между другими
CreateChangedCommand	public	ICommand	float deltaTime	Создаёт команду для отправки состояния другим клиентам
FromGameObject	public	void	GameObject gameObject	Получает состояние из объекта
ApplyToObject	public	void	GameObject gameObject	Применяет состояние к объекту

Таблица 3.78

Описание полей, методов и свойств интерфейса Interpolation.Properties.IGameObjectProperty

Методы интерфейса Interpolation.Properties.IGameObjectProperty				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
CopyFrom	private	void	IGameObjectProperty state	Копирует состояние из другого

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

FromGameObject	private	void	GameObject gameObject	Получает состояние из объекта
ApplyToObject	private	void	GameObject gameObject	Применяет состояние к объекту
Interpolate	private	void	IGameObjectProperty lastLastState, IGameObjectProperty lastState, IGameObjectProperty nextState, float coef	Интерполирует состояние между другими
CreateChangedCommand	private	ICommand	float deltaTime	Создаёт команду для отправки состояния другим клиентам

Таблица 3.79

Описание полей, методов и свойств класса Interpolation.UnmanagedGameObject<T>

Поля класса Interpolation.UnmanagedGameObject<T>				
Имя	Модификатор доступа	Тип	Назначение	
timePerFrame	private	float	Период синхронизации состояния	
lastlastState,	private	Data	Состояния объекта	
state	private	IGameObjectProperty	Текущее состояние объекта	
nextNextState	private	Data	Состояние объекта после следующего	
lastMessageTime	private	float	Время, когда было получено последнее обновление состояния	
P0P1InterpolationCoef	private	float	Коэффициент между предпредыдущим и предыдущим состоянием	
P1P2InterpolationCoef	private	float	Коэффициент между предыдущим и следующим состоянием	
interpolationTime	private	float	Текущее время интерполяции	
Методы класса Interpolation.UnmanagedGameObject<T>				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Init	private	void		Инициализирует переменные
SwitchToNextState	private	void		Переключается на следующее состояние

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

SetStateAnimated	public	void	T State, float deltaSinceLast	Устанавливает состояние объекта с плаыным переходом
Interpolate	private	void	float coef	Интерполирует и применяет состояние
Animate	private	void	float delta	Анимировать состояние
Update	private	void		Обновляет состояние. Автоматически вызывается Unity каждый кадр

Таблица 3.80

Описание полей, методов и свойств класса JsonRequest.Request

Поля класса JsonRequest.Request				
Имя	Модификатор доступа	Тип	Назначение	
room	public	int	Комната для запроса	
id	public	int	ID запроса	
json	public	string	Запрос в JSON формате	
isCompleted	public	bool	Выполнен ли запрос	
callback	public	Action<JsonValue>	Действие, которое нужно сделать после получения ответа	
timeoutTime	public	float	Время в которое запрос будет признан невыполненным	
timeout	public	float	Время, через которое запрос будет признан невыполненным	
retries	public	int	Количество попыток повторить запрос	
room,	public	Request(ClientCommandsRoom	Конструктор JSON запроса	
Методы класса JsonRequest.Request				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
this	private	конструктор	room.roomID, type, json, callback, timeout, retries	Конструктор JSON запроса
Update	public	void		Обновляет состояние запроса

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

GotResponse	public	void	Response response	Обрабатывает получение ответа на запрос
-------------	--------	------	----------------------	--

Таблица 3.81

Описание полей, методов и свойств класса JsonRequest.RequestsManager

Поля класса JsonRequest.RequestsManager				
Имя	Модификатор доступа	Тип	Назначение	
Request>	public	SortedDictionary<int,	Словарь с запросами	
Методы класса JsonRequest.RequestsManager				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Update	public	void		Обновляет состояния запросов
Send	public	void	Request request	Отправляет запрос на сервер

Таблица 3.82

Описание полей, методов и свойств класса JsonRequest.Response

Поля класса JsonRequest.Response				
Имя	Модификатор доступа	Тип	Назначение	
_id	public	int	ID запроса / ответа	
json	public	LightJson.JsonValue	Ответ в JSON формате	
Методы класса JsonRequest.Response				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Response	private	конструктор	int _id, LightJson.JsonValue json	Конструктор ответа на запрос
Deserialize	public	Response	byte[] arr	Десериализует ответ на запрос из бинарного сообщения
Run	public	void		Обрабатывает ответ с сервера
ToString	public	string		Выводит ответ с сервера в строку

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Таблица 3.83

Описание полей, методов и свойств класса MainUIController

Поля класса MainUIController			
Имя	Модификатор доступа	Тип	Назначение
_mainui	private	MainUIController	Переменная для хранения ссылки на MainUIController
spawned	private	bool	Был ли создан главный интерфейс
gunImage	public	Image	Элемент интерфейса с картинкой текущего оружия
pistolSprite, bulletsPanel	public	Sprite	Картинки оружия
	public	MultiImagePanel	Элемент интерфейса, показывающий оставшееся количество патронов
magazinesPanel	public	MultiImagePanel	Элемент интерфейса, показывающий оставшееся количество магазинов
gunsPanel	public	GameObject	Панель интерфейса, показывающая информацию об оружии
scoreText	public	TextMeshProUGUI	Текстовый элемент интерфейса, показывающий очки игроков в текущей игре
taskText	public	TextMeshProUGUI	Текстовый элемент интерфейса, показывающий задачу в текущей игре
timerText	public	TextMeshProUGUI	Текстовый элемент интерфейса, показывающий время до конца текущей игры
totalScorePanel	public	GameObject	Панель интерфейса, показывающая результаты игры
totalScoreText	public	TextMeshProUGUI	Текст с результатами игры
exitButton	public	Button	Кнопка выхода из игры
chatPanel	public	GameObject	Панель с чатом
chatInput	public	TMP_InputField	Поле для ввода сообщения в чат
chatText	public	TextMeshProUGUI	Текстовый элемент интерфейса, показывающий сообщения в чате
chatMessages	public	List<string>	Список с сообщениями в чате
recentlyStoppedTyping	private	bool	Был ли в последнем кадре прекращён набор текста
lasttime	private	int	Время, которое сейчас показывается на таймере с обратным отсчётом
Свойства класса MainUIController			
Имя	Модификатор доступа	Тип	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

totalScoreTextUnformatted	private get; private set	string	Текст для результатов игры, в который нужно подставить оставшееся время	
Методы класса MainUIController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
ColorForPlayer	public	string	Player player	Возвращает цвет, в который нужно раскрашивать ник данного игрока
RedrawScore	private	void		Перерисовывает текст с очками игроков
AddScoreTable	private	void	StringBuilder text	Добавляет StringBuilder таблицу с результатами игры
ShowTotalScore	public	void	int gamesRemaining, int timeRemaining	Показывает таблицу с результатами игры
SetTotalScoreTimeRemaining	public	void	int time	Изменяет оставшееся время в таблице с результатами игры
HideTotalScore	public	void		Прекращает показ таблицы с результатами игры
ShowFinalResults	public	void		Показывает финальную таблицу с результатами игры
ExitButtonClicked	public	void		Входит из матча. Вызывается Unity при нажатии на кнопку выхода
SetupHandlers	public	void		Устанавливает обработчики событий

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Awake	private	void		Инициализирует переменные и перерисовывает интерфейс
SetTask	public	void	string text	Устанавливает текст задачи
StartTyping	public	void		Устанавливает состояние на набор текста
StopTyping	public	void		Выключает состояние набора текста
SendToChat	public	void		Отправляет сообщение в чат
AddChatMessage	public	void	Player player, string message	Добавляет сообщение в чат
DeleteMessageAfterTime	private	IEnumerator	float time, string message	Корутина для удаления сообщения из чата через время
RedrawChat	public	void		Перерисовывает текст чата
SetTimerTime	public	void	int time	Устанавливает время на таймере с обратным отсчётом
Update	private	void		Обновляет состояние интерфейса. Автоматически вызывается Unity каждый кадр

Таблица 3.84

Описание полей, методов и свойств класса Networking.ClientCommandsRoom

Поля класса Networking.ClientCommandsRoom			
Имя	Модификатор доступа	Тип	Назначение
lastMessage	private	int	Номер последней обработанной команды

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

losedMessages	private	OrderedDictionary	Словарь с потерянными сообщениями	
commandsSystem	private	CommandsSystem.CommandsSystem	CommandsSystem для кодирования сообщений	
lastTimeRequestSended	private	float	Когда последний раз был отправлен запрос на переотправку команды	
roomID	public	int	ID комнаты	
Методы класса Networking.ClientCommandsRoom				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
~ClientCommandsRoom	private	конструктор		Деструктор комнаты. Отправляет команду выхода из комнаты
RunSimpleCommand	public	void	ICommand command, MessageFlags flags	Отправляет простую команду на сервер
RunUniqCommand	public	void	ICommand command, int i1, int i2, MessageFlags flags	Отправляет уникальную команду на сервер
RunAskMessage	private	void	int firstIndex, int lastIndex, MessageFlags flags	Отправляет команду с запросом сообщений на сервер
RunJoinMessage	private	void		Отправляет команду присоединения к комнате
RunLeaveMessage	private	void		Отправляет команду покидания комнаты
ReceiveCommand	private	void	ICommand command	Выполняет команду
HandleCommand	public	void	int commandId, ICommand command	Обрабатывает полученную команду

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



# **RU.17701729.04.01-01 81 01-1**

Таблица 3.85

Описание полей, методов и свойств класса Networking.CommandsHandler

Поля класса Networking.CommandsHandler				
Имя	Модификатор доступа	Тип	Назначение	
webSocketHandler	public	WebSocketHandler	Обработчик WebSocket	
commandsSystem	private	CommandsSystem.CommandsSystem	CommandsSystem для сериализации команд	
matchmakingRoom	public	ClientCommandsRoom	Комната для поиска матча	
gameRoom	public	ClientCommandsRoom	Комната для матча (игры)	
gameModeRoom	public	ClientCommandsRoom	Комната для игрового режима	
Методы класса Networking.CommandsHandler				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Reset	public	void		Сбрасывает значения переменных
RoomById	public	ClientCommandsRoom	int id	Полчает комнату по ID
Update	public	void		Получает и отправляет команды
Stop	public	void		Отключается от сервера

Таблица 3.86

Описание полей, методов и свойств класса ObjectID:

Поля класса ObjectID:				
Имя	Модификатор доступа	Тип	Назначение	
ObjectData>	private	Dictionary<int,	Слоаврь объектов по ID	
int>	private	Dictionary<int,	Словарь ID по UnityID	
random	private	System.Random	Генератор случайных чисел	
RandomID	public	int	Случаный ID	
Методы класса ObjectID:				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
StoreObject	public	void	GameObject gameObject, int	Сохраняет объект

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

			id, int owner, int creator	
StoreOwnedObject	public	void	GameObject gameObject, int creator	Сохраняет объект, созданный локально
GetID	public	int	GameObject gameObject	Получает ID объекта
TryGetID	public	bool	GameObject gameObject, out int result	Пытается получить ID объекта
TryGetObject	public	bool	int id, out GameObject gameObject	Пытается получить объект по ID
GetObject	public	GameObject	int id	Получает объект по ID
RemoveObject	public	void	GameObject gameObject	Удаляет объект
ToString	public	string		Выводит информацию об объектах в строку
GetObjectData	private	ObjectData	int id	Получает данные об объекте по ID
GetObjectData	private	ObjectData	GameObject gameObject	Получает данные об объекте
GetOwner	public	int	int id	Получает владельца объекта
GetOwner	public	int	GameObject gameObject	Получает владельца объекта
TryGetOwner	public	bool	int id, out int owner	Пытается получить владельца объекта
IsOwned	public	bool	int id	Проверяет, владеет ли данный клиент объектом
IsOwned	public	bool	GameObject go	Проверяет, владеет ли данный клиент объектом
SetOwner	public	void	int id, int owner	Изменяет владельца объекта

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

GetCreator	public	int	int id	Получает создателя объекта
GetCreator	public	int	GameObject gameObject	Получает создателя объекта
TryGetCreator	public	bool	int id, out int creator	Пытается получить создателя объекта
Clear	public	void		Очищает значения переменных

Таблица 3.87

Описание полей, методов и свойств класса PlayerController

Поля класса PlayerController				
Имя	Модификатор доступа	Тип	Назначение	
plane	private	Plane	Переменная для внутреннего использования (нужна для уменьшения нагрузки на сборщик мусора)	
Методы класса PlayerController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	protected	void		Инициализирует переменные
Update	private	void		Управляет персонажем в соответствии с положением мышки и нажатыми клавишами. Автоматически вызывается Unity каждый кадр

Таблица 3.88

Описание полей, методов и свойств класса QueueExtension

Методы класса QueueExtension				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

TryDequeue<T>	public	bool	this Queue<T> queue, out T res	Пытается достать элемент из очереди
---------------	--------	------	-----------------------------------	--

Таблица 3.89

Описание полей, методов и свойств класса Triangle

Поля класса Triangle				
Имя	Модификатор доступа	Тип	Назначение	
a,	public	Vector3	Координаты треугольника	
Area	public	float	Вычисляет площадь треугольника	
Методы класса Triangle				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Triangle	public	конструктор		Конструктор треугольника
Triangle	public	конструктор	Vector3 a, Vector3 b, Vector3 c	Конструктор треугольника
RandomPoint	public	Vector3		Получает случайную точку в треугольнике

Таблица 3.90

Описание полей, методов и свойств класса TrianglePolygon

Поля класса TrianglePolygon				
Имя	Модификатор доступа	Тип	Назначение	
triangles	private	List<Triangle>	Список треугольников	
areaSum	private	float	Суммарная площадь всех треугольников	
Методы класса TrianglePolygon				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
TrianglePolygon	public	конструктор	List<Vector3> points	Конструктор
RandomPoint	public	Vector3		Находит случайную точку внутри многоугольника

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Таблица 3.91

Описание полей, методов и свойств класса UI.DebugUI

Поля класса UI.DebugUI				
Имя	Модификатор доступа	Тип	Назначение	
textPanel	public	Text	Элемент интерфейса для отладочного текста	
debugTextDirty	private	bool	Флаг, сообщающий, что нужно перерисовать текст	
_debugText	private	string[]	Массив с отладочным текстом	
Методы класса UI.DebugUI				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	private	void		Инициализирует переменные
Update	private	void		Перерисовывает отладочный текст, если требуется. Автоматически вызывается Unity каждый кадр

Таблица 3.92

Описание полей, методов и свойств класса UI.MultiImagePanel

Поля класса UI.MultiImagePanel				
Имя	Модификатор доступа	Тип	Назначение	
images	private	List<Image>	Массив изображений	
image	public	Image	Изображение	
panel	private	RectTransform	Панель, на которой показывается данный элемента	
Методы класса UI.MultiImagePanel				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OnEnable	public	void		Инициализирует переменные
SetMaxImagesCount	public	void	int count	Устанавливает максимальное число изображений, которое нужно показывать

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

SetActiveImagesCount	public	void	int count	Устанавливает число изображений, которые нужно показывать сейчас
----------------------	--------	------	-----------	--

Таблица 3.93

Описание полей, методов и свойств класса UI.StartUIController

Поля класса UI.StartUIController				
Имя	Модификатор доступа	Тип	Назначение	
nameInput	public	TMP_InputField	Поле для ввода ника	
matchInfoText	public	TextMeshProUGUI	Текстовый элемент с информацией о текущем матче	
JoinUI	public	GameObject	Панель с интерфейсом для ввода ника и кнопкой Play	
MatchUI	public	GameObject	Панель для интерфейса с информацией о чате	
specificName	public	bool	Ввёл ли пользователь ник	
Методы класса UI.StartUIController				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Awake	public	void		Инициализирует переменные
Start	private	void		Устанавливает обработчики событий
OnPlayClicked	public	void		Обрабатывает нажатие кнопки играть
OnDestroy	private	void		Переключает на главный интерфейс
Update	private	void		При нажатии клавиши enter переходит к поиску матча

Таблица 3.94

Описание полей, методов и свойств класса Util2.AutoDisableRendererOnStart

Методы класса Util2.AutoDisableRendererOnStart				
Имя	Модификатор доступа	Тип	Аргументы	Назначение

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

Start	private	void		Отключает Renderer
-------	---------	------	--	-----------------------

Таблица 3.95

Описание полей, методов и свойств класса Util2.AutoID

Поля класса Util2.AutoID				
Имя	Модификатор доступа	Тип	Назначение	
ID	public	int	ID объекта	
Методы класса Util2.AutoID				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Reset	public	void		Генерирует случайный ID
Awake	public	void		Сохраняет ID объекта
Update	public	void		Удаляет эту компоненту из объекта во время игры

Таблица 3.96

Описание полей, методов и свойств класса Util2.AutoMatchJoiner

Поля класса Util2.AutoMatchJoiner				
Имя	Модификатор доступа	Тип	Назначение	
isRunning	public	bool	Запущен ли автовыбор матча	
sneedWaitOtherPlayers	public	bool	Нужно ли ждать другого игрока для старта игры	
needWaitOtherPlayers	public	bool	Нужно ли ждать другого игрока для старта игры	
Методы класса Util2.AutoMatchJoiner				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Awake	public	void		Инициализирует переменные
Start	public	void		Автоматически входит в матч

Таблица 3.97

Описание полей, методов и свойств класса Util2.RotatingItem

Поля класса Util2.RotatingItem				
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

Имя	Модификатор доступа	Тип	Назначение	
rotationSpeed	public	float	Скорость вращения	
upDownSpeed	public	float	Скорость передвижения вверх-вниз	
upDownAmplitude	public	float	Амплитуда перемещения вверх-вниз	
phase	private	float	Сдвиг в фазе перемещения вверх-вниз	
startingY	private	float	Стартовая координата у	
Методы класса Util2.RotatingItem				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Start	public	void		Инициализирует переменные
Update	public	void		Вращает объект и перемещает его вверх-вниз

Таблица 3.98

Описание полей, методов и свойств класса Util2.gUtil

Методы класса Util2.gUtil				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Swap<T>	public	void	ref T lhs, ref T rhs	Меняет переменные местами

Таблица 3.99

Описание полей, методов и свойств класса Vector2Extension

Методы класса Vector2Extension				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Rotate	public	Vector2	this Vector2 v, float degrees	Поворачивает вектор

Таблица 3.100

Описание полей, методов и свойств класса WebGLEditorScript

Методы класса WebGLEditorScript				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



**RU.17701729.04.01-01 81 01-1**

[MenuItem	private	конструктор	"Tools/Setup webgl settings"	Устанавливает настройки сборки в webgl
DisableErrorMessageTesting	public	void		

Таблица 3.101

Описание полей, методов и свойств класса WebSocketHandler

Поля класса WebSocketHandler				
Имя	Модификатор доступа	Тип	Назначение	
clientToServerMessages	public	Queue<byte[]>	Очередь сообщений с клиента на сервер	
serverToClientMessages	public	Queue<byte[]>	Очередь сообщений с сервера на клиент	
connectTask	private	Task<WebSocket>	Ассинхронная задача подключиться к серверу	
sendTask	private	Task	Асинхронная задача отправить сообщение на сервер	
websocket	private	WebSocket	Вебсокет для общения с сервером	
Методы класса WebSocketHandler				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Stop	public	void		Отключается от сервера
Update	public	void		Обновляет состояние вебсокета
CreateWebSocket	private	Task<WebSocket>		Создаёт вебсокет
HandleWebSocketMessage	private	void	byte[] data	Обрабатывает сообщение, пришедшее в вебсокет

Таблица 3.102

Описание полей, методов и свойств класса sClient

Поля класса sClient			
Имя	Модификатор доступа	Тип	Назначение
isTyping	public	bool	Набирает ли пользователь сейчас текст

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

# **RU.17701729.04.01-01 81 01-1**

int	public	const	Количество сообщений в секунду, отправляемых по сети для синхронизации	
ID	public	int	ID клиента	
random	public	System.Random	Генератор случайных чисел	
gameStartTime	private	float	Время, когда началась игра	
state	public	STATE	Текущее состояния	
GameTime	public	float	Время, прошедшее с начала игры	
initialized	private	bool	Был ли инициализирован этот класс	
Методы класса sClient				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SetGameStarted	public	void		Переключает состояние на начало игры
Init	public	void		Инициализирует переменные
Reset	public	void		Сбрасывает значения переменных
StartFindingMatch	public	void		Переключает состояние в поиск матча
Awake	private	void		Инициализирует переменные при запуске игры
Update	private	void		Обновляет состояние
OnApplicationQuit	private	void		Обрабатывает выход из приложения
LoadScene	public	void	string sceneName	Загружает сцену

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**RU.17701729.04.01-01 81 01-1**

**ПРИЛОЖЕНИЕ 3**  
**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

- 1) ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 2) ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 4) ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 5) ГОСТ 19.105-78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 7) ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 8) ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 9) ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 10) ГОСТ 15150-69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.
- 11) ГОСТ 19.602-78 Правила дублирования, учета и хранения программных документов, выполненных печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 12) ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

[illegible]