

Главное управление образования Гродненского облисполкома

Учреждение образования

«Гродненский государственный политехнический колледж»

Специальность: 2 – 40 01 01 «Программное обеспечение информационных технологий»

Специализация: 2 – 40 01 01 35 «Программное обеспечение обработки экономической и деловой информации»

Дисциплина: «Технология разработки программного обеспечения»

Группа: ПЗТ – 36

ТЕХНО-РАБОЧИЙ ПРОЕКТ

Техническое задание на тему: Разработка игрового приложения «Токуо Ghoul»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Разработал

М.О. Ковалёнок

Руководитель проекта

Е.В. Заяц

2022

Содержание

1	Постановка задачи	3
1.1	Организационно-экономическая сущность задачи.....	3
1.2	Функциональные требования.....	4
1.2.1	Регистрация и вход в аккаунт.....	4
1.2.2	Игровой процесс.....	4
1.2.3	Настройки.....	5
2	Техническое задание	8
3	Реализация.....	14
4	Проектирование ПИ.....	18
4.0.1	UX – Проектирование.....	18
4.0.2	UI – Проектирование.....	19
4.0.3	Документирование ПО	20
5	Приложение А.....	29
5.1	Диаграмма вариантов использования.....	30
5.2	Функциональная модель (1).....	31
5.3	Функциональная модель (2).....	31
5.4	Функциональная модель (3).....	32
5.5	Модель данных.....	33
5.6	Диаграмма объектов.....	34
5.7	Диаграмма последовательности.....	35
5.8	Диаграмма деятельности.....	36
5.9	Диаграмма компонентов.....	37
6	Прототип ПП.....	38

						КП 2-40 01 01.35.36.04.22 ПЗ					
Изм.	Кол	Лист	№ док	Подпись	Лпмл	Техническое задание на тему: Разработка игрового приложения "То- кью Ghoul"			Стадия	Лист	Листов
Разраб.		Ковалёнок									
Пров.		Заяц								2	
									УО ГГПК		
Н. контр.											
Утв.											

1 Постановка задачи

1.1 Организационно-экономическая сущность задачи

Наименование задачи: Игра тамагочи “Токийский гуль”

Иконка игры: (Рисунок -1)

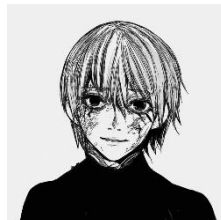


Рисунок - 1

Цель разработки: Дать пользователю возможность почувствовать себя в роли владельца питомца, роль которого будет выполнять известный персонаж из аниме по имени “ Kaneki Ken”.

Назначение: Данный игровой продукт имеет развлекательную цель для всех типов игроков. Игра понравится тем, кто не смотрел аниме, а в особенности ее оценят фанаты данного тайтла.

Периодичность использования: В любое свободное время.

Источники и способы получения данных: Play Марке и по специализированным ссылкам.

Обзор аналогичных ПП: На данный момент имеется большое количество игр данного жанра с большим количеством различных персонажей, но именно для фанатов аниме практически ничего нет. Рассмотрим игры подобного жанра:

My anime girl 2: Игра является довольно-таки слабым представителем данного жанра, но при этом у нее достаточно пользователей. В игре не хватает многих функций, с которыми игра могла бы стать более увлекательной. Также нет какого-либо определенного стиля, персонаж отрисован в аниме-рисовке, но остальные объекты игры полностью отличаются.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		3

1.2 Функциональные требования:

Игрок:

1.2.1 Регистрация и вход в аккаунт:

После установки и открытия игры будет выскакивать окно, в котором пользователю будет предлагаться выбрать аккаунт, который привязан к Google-Play.

1.2.2 Игровой процесс

Игра начнется с того, что пользователю представят персонажа, которого он будет развивать по ходу игры. Появится объяснение каждого показателя его состояния:

1)Сон

Показатель сна будет зависеть от того, как долго или как мало персонаж отдыхал. При прохождении мини-игр и при употреблении пищи показатель сна будет падать, в таком случае персонажа надо будет уложить спать. Если показатель сна упадет до минимума – персонаж умрет.

2)Еда

Показатель еды будет зависеть от того, как много или как мало персонаж будет есть. По лору манги и аниме, персонаж может употреблять только 2 вещи: Человечина и кофе. Если показатель упадет до минимума – персонаж умрет.

3)Гигиена

Показатель гигиены будет падать после употребления пищи и сна. Чтобы повышать показатель надо будет мыть персонажа. Так же гигиена будет резко уменьшаться, если психическое состояние будет минимальным. Если показатель уйдет в минимум – сразу же упадет психическое состояние, персонаж умрет.

4)Психическое состояние

Будет зависеть от количества сыгранных мини-игр и от выше перечис-

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

ленных показателей. Если состояние достигнет минимума – персонаж сойдет с ума и умрет.

5)Статус “Кагуне”

Показатель “Кагуне” является чисто косметическим фактором. Разные скины на этот показатель можно будет открыть в мини-играх.

В целом вышеперечисленные факторы и есть геймплей игры. Когда персонаж достигнет 993 уровня – игра будет являться пройденной. Уровень будет расти от пройденных мини-игр. В игре будет 3 мини-игры:

1) Игра по типу классической змейки
“Сколопендра”

-Суть игры будет очень похожа на игру “Змейка”. Игрок на карте должен будет собирать очки, играя за стилизованную змейку. При смерти будет выводиться количество очков, заработанных до столкновения со стенкой.

2) Игра по типу игры Doodle Jump
“Сова”

-Суть игры будет похожа на игру Doodle Jump. Игрок должен будет прыгать по островам, зарабатывая очки, при падении игрока с острова игра будет окончена.

3) Игра по типу Ping Pong
“Кагуне”

-Классический пинг-понг, но со стилизацией под геймдизайн всей игры.

Уровень будет расти не по 1 единице, а по 7. Данное решение обусловлено лором манги.

1.2.3 Настройки

В игре будет лишь одна настройка – регулятор громкости игры.

Периодичность использования: Постоянное использование.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

Требования к применению:

Продукт имеет материал исключительно для лиц не моложе 16 лет. Пользователю требуется войти или зарегистрироваться в Google аккаунт.

Требования к производительности:

Программа должна запускаться на слабо конфигурированных системах.

Минимальные системные требования:

- Оперативная память не менее 1Гб ОЗУ;
- Процессор не ниже модели: Snapdragon 425 MSM8917;
- Версия Android не ниже версии 6.0;
- 200 МБ свободного места на жестком диске;

Максимальное время на запуск программы на запуск программы при минимальных требованиях должно составлять не более 10 секунд.

Требования к реализации:

Для достижения максимальной производительности и оптимизации приложение должна быть разработана на движке Unity3d на языке C# в среде Visual Studio 2022.

Требование к надежности:

Приложение должно производить самовосстановление после сбоя работы.

Требования к интерфейсу:

Интерфейс должен быть с приятной цветовой гаммой и понятной для пользователя. Следовательно, каждое окно должно иметь ясную визуальную иерархию своих элементов. Фрагменты текста должны располагаться на экране так, чтобы пользователя было просто и понятно принимать информацию. Пользователь не должен испытывать какого-либо дискомфорта в плане восприятия информации, отображённой на экране. Объекты (рисунки и символы) не должны быть слишком мелкие.

Геймдизайн:

В игре будет несколько комнат, нахождение в которых будет зависеть от

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		6

нажатой иконки показателя. Будет: Кухня, Спальня, Игровая, Ванная, Комната со стулом, кабинет Антейку.

Вот пример главной комнаты: (Рисунок - 2)



Рисунок – 2

Требования к программной документации:

Для удобства использования программного продукта пользователем,

необходимо добавить файл справки, в котором будет находиться вся необходимая информация о работе с программой.

Технико-экономические показатели:

Ориентировочная экономическая эффективность не рассчитываются.

Предполагаемое число использования программой в год – 365 сеансов на одном рабочем месте.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		7

2 Техническое задание

Введение

Наименование программы:

Наименование программного продукта –Тамагочи «Токуо ghoul».

Краткая характеристика области применения:

Данный игровой продукт имеет развлекательную цель для всех типов игроков.

Основания для разработки:

Основанием для разработки является задание на практику на тему «Разработка программного обеспечения, предоставляющего планирование задач пользователя». Тема согласована с преподавателем.

Назначение разработки:

Дать пользователю возможность почувствовать себя в роли владельца питомца, роль которого будет выполнять известный персонаж из аниме по имени “Kaneki Ken”.

Игровой процесс:

Игра начнется с того, что пользователю представят персонажа, которого он будет развивать по ходу игры. Появится объяснение каждого показателя его состояния:

1)Сон

Показатель сна будет зависеть от того, как долго или как мало персонаж отдыхал. При прохождении мини-игр и при употреблении пищи показатель сна будет падать, в таком случае персонажа надо будет уложить спать. Если показатель сна упадет до минимума – персонаж умрет.

2)Еда

Показатель еды будет зависеть от того, как много или как мало персонаж будет есть. По лору манги и аниме, персонаж может употреблять только 2 вещи: Человечина и кофе. Если показатель упадет до минимума – персонаж умрет.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		8

3) Гигиена

Показатель гигиены будет падать после употребления пищи и сна. Чтобы повышать показатель надо будет мыть персонажа. Так же гигиена будет резко уменьшаться, если психическое состояние будет минимальным. Если показатель уйдет в минимум – сразу же упадет психическое состояние, персонаж умрет.

4) Психическое состояние

Будет зависеть от количества сыгранных мини-игр и от выше перечисленных показателей. Если состояние достигнет минимума – персонаж сойдет с ума и умрет.

5) Статус “Кагуне”

Показатель “Кагуне” является чисто косметическим фактором. Разные скины на этот показатель можно будет открыть в мини-играх.

В целом вышеперечисленные факторы и есть геймплей игры. Когда персонаж достигнет 993 уровня – игра будет являться пройденной. Уровень будет расти от пройденных мини-игр. В игре будет 3 мини-игры:

4) Игра по типу классической змейки

“Сколопендра”

5) Игра по типу игры Doodle Jump

“Сова”

6) Игра по типу Ping Pong

“Кагуне”

Уровень будет расти не по 1 единице, а по 7. Данное решение обусловлено лором манги.

Геймдизайн:

В игре будет несколько комнат, нахождение в которых будет зависеть от нажатой иконки показателя. Будет: Кухня, Спальня, Игровая, Ванная, Комната со стулом, кабинет Антейку.

Вот пример главной комнаты: (Рисунок – 3)



Рисунок - 3

Требования к программе или программному изделию:

Требования к применению:

Продукт имеет материал исключительно для лиц не моложе 16 лет. Пользователю требуется войти или зарегистрироваться в Google аккаунт.

Требования к производительности:

Программа должна запускаться на слабо конфигурированных системах.

Минимальные системные требования:

- Оперативная память не менее 1Гб ОЗУ;
- Процессор не ниже модели: Snapdragon 425 MSM8917;
- Версия Android не ниже версии 6.0;
- 200 МБ свободного места на жестком диске;

Максимальное время на запуск программы на запуск программы при минимальных требованиях должно составлять не более 10 секунд.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		10

Требования к реализации:

Для достижения максимальной производительности и оптимизации приложение должна быть разработана на движке Unity3d на языке C# в среде Visual Studio 2022.

Требование к надежности:

Приложение должно производить самовосстановление после сбоя работы.

Требования к интерфейсу:

Интерфейс должен быть с приятной цветовой гаммой и понятной для пользователя.

Следовательно, каждое окно должно иметь ясную визуальную иерархию своих элементов. Фрагменты текста должны располагаться на экране так, чтобы пользователя было просто и понятно принимать информацию.

Пользователь не должен испытывать какого-либо дискомфорта в плане восприятия информация, отображённой на экране. Объекты (рисунки и символы) не должны быть слишком мелкие.

Требования к программной документации:

Для удобства использования программного продукта пользователем, необходимо добавить файл справки, в котором будет находиться вся необходимая информация о работе с программой.

Технико-экономические показатели:

Ориентировочная экономическая эффективность не рассчитываются.

Предполагаемое число использования программой в год – 365 сеансов на одном рабочем месте.

Стадии и этапы разработки

Стадии разработки:

Разработка должна быть проведена в три стадии: разработка технического задания, рабочее проектирование, внедрение.

Этапы разработки:

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		11

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

разработка программы;

разработка программной документации;

испытания программы.

На стадии внедрения должен быть выполнен этап разработки – подготовка и передача программы.

Содержание работ по этапам:

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

1. Постановка задачи;
2. Определение и уточнение требований к техническим средствам;
3. Определение требований к программе;
4. Определение стадий, этапов и сроков разработки программы и документации на неё;
5. Выбор языков программирования;
6. Согласование и утверждение технического задания.

На этапе разработки программной документации должна быть выполнена работа по программированию (кодированию) и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77 с требованием п. Предварительный состав программной документации настоящего технического задания.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		12

Разработка, согласование и утверждение программы и методики испытаний;

Проведение приемо-сдаточных испытаний;

Корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа

по подготовке и передаче программы и программной документации в эксплуатацию на объектах Заказчика.

Порядок контроля и приемки:

Виды испытаний:

Приемо-сдаточные испытания программы должны проводиться согласно разработанной Исполнителем и согласованной Заказчиком программы и методик испытаний.

Ход проведения приемо-сдаточных испытаний Заказчик и Исполнитель документируют в Протоколе проведения испытаний.

Общие требования к приемке работы:

На основе Протокола проведения испытаний Исполнитель совместно с Заказчиком подписывают Акт приемки-сдачи программы в эксплуатацию.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		13

3 Реализация

№	Название парадигмы	Сущность парадигмы, основные идеи, принципы, объекты.	Языки поддеживающие данную парадигму. (2-3 языка)	Достоинства	Недостатки	Примеры программ
	Императивное программирование	Это парадигма, основанная на составлении алгоритма действий (инструкций/команд), которые изменяют состояние (информацию/данные/память) программы.	C, C++, java	Выполняется компьютером сразу, без предварительной компиляции	При императивном подходе к написанию кода (в отличие от функционального подхода, относящегося к декларативной парадигме) широко используется присваивание. Наличие операторов присваивания увеличивает сложность модели вычислений и делает императив-	Fortran, Ada

					ные программы подверженными специфическим ошибкам, не встречающимся при функциональном подходе	
	Декларативное программирование	Это парадигма, при которой описывается желаемый результат, без составления детального алгоритма его получения.	HTML, SQL	При создании HTML мы с помощью тегов описываем, какую хотим получить страничку в браузере, а не то, как нарисовать на экране заголовки статьи, оглавление и текст.	Декларативные компьютерные языки часто не полны по Тьюрингу, так как теоретически не всегда возможно порождение исполняемого кода по декларативному описанию.	DOM, React
	Структурное программирование	Парадигма программирования, в основе которой лежит представление	Basic, Cg, Pascal	Серьезно облегчилось понимание программ, появилась возможность разработки программ в нормальном промышлен-	Главный недостаток структурного подхода заключается в следующем: процессы и данные	<u>goto</u> , Cgi

		ние программы в виде иерархической структуры блоков.		ном режиме, когда программу может без особых затруднений понять не только её автор, но и другие программисты.	существуют отдельно друг от друга (как в модели деятельности организации, так и в модели программной системы), причем проектирование ведется от процессов к данным. Таким образом, помимо функциональной декомпозиции, существует также структура данных, находящаяся на втором плане.	
	Процедурное программирование	Программирование на императивном языке, при котором последо-	Tcl, Perl, Lua	Процедурное программирование отличается подходом для программирования общего назначения. Закодиро	Код программы труднее писать, когда используется процедурное программиро-	Torogun, Coat

		<p>вательно выпол- няе- мые опер ато- ры можн о со- брать в подпро граммы, то есть более крупные целост- ные еди- ни- цы кода, с помо- щью ме- ханиз- мов са- мого языка.</p>		<p>ванная про- стота наряду с простотой реализации компилято- ров и интер- претато- ров.Большое разнообразие книг и он- лайн- материалов курсов по проверенным алгоритмам, облегчающих обучение в процес- се.Исходный код перено- сим, поэтому его можно использовать и для другого процессо- ра.Код может быть повтор- но использо- ван в разных частях про- граммы, без необходимос- ти копиро- вать его. Бла- годаря мето- дике проце- дурного про- граммирова- ния требова- ния к памяти также сокра- щаются .Ход</p>	<p>вание. Проце- дурный код часто не может быть ис- пользован повторно, что может привести к необходи- мости вос- создания кода, если это необ- ходимо для ис- пользова- ния в дру- гом при- ложе- нии.Слож- но об- щаться с реальны- ми объек- тами. Важное значение придается операции, а не дан- ным, что может со- здавать проблемы в некото- рых слу- чаях, свя- занных с данными. Данные</p>	
--	--	---	--	--	--	--

				программы можно легко отследить	открыты для всей програм- мы, что делает их не очень безопас- ными	
	Модульное программиро- вание	Это ор- ганизаци- я про- граммы как со- вокупно- сти не- больших незави- симых блоков, называ- емых модуля- ми, структу- ра и по- ведение которых подчи- няются опреде- лённым прави- лам.	Обе- рон, Ком- по- нент- ный пас- каль, Zonnon	Модульные программы легко состав- лять и отла- живать. Функцио- нальные компоненты такой про- граммы мо- гут быть написаны и отлажены порознь.	Модуль- ность тре- бует большей дополни- тельной работы. Чтобы пи- сать мо- дульные програм- мы, про- граммист должен быть зна- чительно более ак- куратным на этапе проекти- рования программ- ной разра- ботки. Он должен проекти- ровать свои про- граммы по нисходя- щей схеме, начиная с	Xerox Parc

					<p>верхних уровней всей программы и затем продвигаясь вниз к более детальному проектированию отдельных подпрограмм.</p>	
	<p>Объектно-ориентированное программирование</p>	<p>методология программирования, основанная на представлении программы в виде совокупности взаимодействующих объектов, каждый</p>	<p>Java, C++, Python, C#</p>	<p>Первые программы на языках программирования высокого уровня, по сути, не были структурированы, и это не вызывало проблем, потому что объёмы кода были, по современным меркам, ничтожны.</p>	<p>В процессе трансляции объектно-ориентированных программ в исполняемый код центрального процесса возникает ряд неоптимальностей по использованию памяти и вычисли-</p>	<p>Тпрогер, S.O.L.I.D.</p>

		из кото- рых яв- ляется экзем- пляр опреде- лённо- го класса , а клас- сы обра- зуют иерар- хию наследо- вания			тельного времени процес- сорных ядер.	
	Функциональ- ное программиро- вание	пара- дигма про- грамми- рования, в кото- рой про- цесс выч исле- ния трак- туется как вы- числение значе- ний фун- кций в матема- тическом понима- нии по- следних (в отли- чие от функц ий как подпро- грамм	Липс, Erlang	Во-первых, для функци- ональных языков нет эффективно- го неупоря- доченного словаря и множества. Чисто функ- циональные словари ра- ботают мед- леннее хэш- таблицы, и для некото- рых прило- жений это может быть критично. Во-вторых, не существуе- т чисто функцио- нальных сла- бых хэш- таблиц, хотя	высоко- уровневые абстрак- ции, кото- рые скры- вают большое количе- ство по- дробно- стей таких рутинных операций, как, например, итериро- вание. За счет этого код получается короче, и, как след- ствие, га- рантирует меньшее количе- ство оши-	ST, Goto

		в процедурном программировании).		для большинства разработчиков этот недостаток может остаться незамеченным.	блок, которые могут быть допущены.	
	Логическое программирование	парадигма программирования, основанная на математической логике — программы в ней задаются в форме логических утверждений и правил вывода. Наиболее известный язык логического программирования — Пролог.	Data-log, Mercury, Oz	операции, вершаемые в логическом программировании всегда понятны; Результат практически всегда не зависит от выбранного пути реализации;	Если брать за пример логического языка программирования Prolog, то на лицо невозможность создания комплексных задач. То есть в реальности логический язык может идти дополнением к процедурному, но самостоятельно используется крайне редко;	Prolog, GGB, NV D

4 Проектирование ПИ

4.0.1 UX – Проектирование (Рисунок - 4)



Рисунок - 4

4.0.2 UI – Проектирование (Рисунок – 5)



Рисунок – 5

6.0 Тестирование и отладка ПО

Таблица – 1 Тестирование и отладка программного продукта

	Описание	Ожидаемый результат	Фактический результат
.	Поддаётся ли модель деформированию	Модель под- даётся деформиро- ванию	Модель под- даётся деформиро- ванию
.	Поддаётся ли модель ретопологии под анимацию	Топология модели сделана под анимацию	Топология модели сделана под анимацию
.	Поддается ли тополо- гия модели дальнейшему текстурированию	На модель хо- рошо накладываются текстуры без де- формаций	На модель хо- рошо накладываются текстуры без де- формаций
.	Поддается ли модель качественной анимации	Модель ани- мируется без дефек- тов и артефактов	Модель ани- мируется без дефек- тов и артефактов
.	Находится ли модель в нужных форматах для экс- порта в игровой движок	Модель нахо- дится в формате стл и обж	Модель нахо- дится в формате стл и обж
.	Находятся ли текстуры для модели в режиме ПБР для правильного отражения глобального освещения	Текстуры находятся в режиме пбр	Текстуры находятся в режиме пбр
.	Имеются ли дефекты на карте нормалей	На карте нор- малей дефектов нет	На карте нор- малей дефектов нет
.	Есть ли возможность	Текстуры ка-	Текстуры ка-

.	наложения текстур на модель в игровом движке	чественно наклады- ваются на модель находясь в игровом движке	чественно наклады- ваются на модель находясь в игровом движке
.	Работает ли анимация в игровом движке	Анимация мо- дели отлично функ- ционирует внутри игрового движка	Анимация мо- дели отлично функ- ционирует внутри игрового движка
0.	Имеются ли искажения на топологии модели при экспорте в другие програм- мы	На модели не имеется деформа- ций топологии при экспорте	На модели не имеется деформа- ций топологии при экспорте

Таблица 1 – Руководство пользователя

Движение вперед	Нажатие кнопки w
Движение назад	Нажатие кнопки s
Движение влево	Нажатие кнопки a
Движение вправо	Нажатие кнопки d
Ускорен- ный бег	Нажатие кнопки shift
Танец	Нажатие кнопки g
Удар	Нажатие кнопки ЛКМ

1. При движении в определенном направлении, нажав соответствующие кнопки – w,s,a,d, воспроизводится следующая анимация

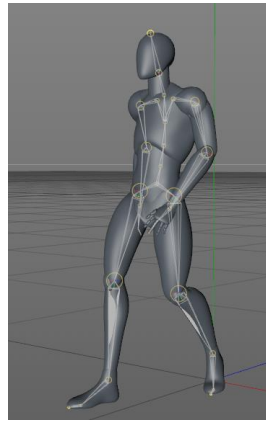


Рисунок 6 – анимация движения

2. При нажатии кнопок передвижения, зажав при этом кнопку shift, воспроизводится анимация бега

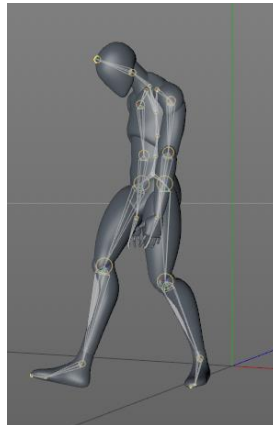


Рисунок 7 – анимация бега

3. При нажатии на кнопку g, воспроизводится анимация танца

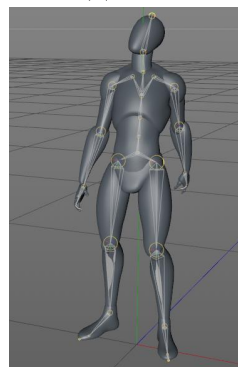


Рисунок 8 – анимация танца

4. При нажатии кнопки ЛКМ на мышке, воспроизводится анимация удара

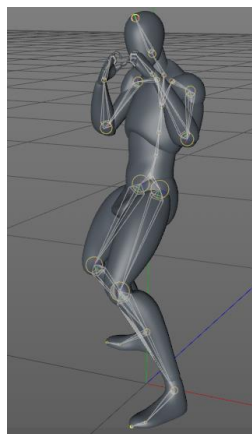


Рисунок 9 – анимация удара

5. При бездействии персонажа, воспроизводится анимация стойки

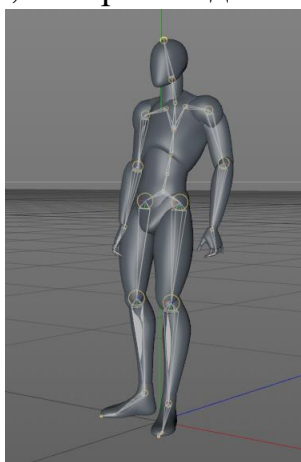


Рисунок 10 – анимация стойки

Таблица 2 – Руководство программиста

ZBrush	В этой программе было реализовано: скульптинг и моделирование
RizonUV	В этой программе было выполнено раз- вертка модели
Topogun	В этой программе было выполнено рето- пология
Sabstunce Painter	В этой программе были сделаны тексту- ры и запекание
Cinema 4D	В этой программе была сделана анима- ция

Unity3D	В этой программе была реализован функционал модели (Анимация)
---------	---

1. В программе Zbrush были реализованы основные формы модели и мелкие детали модели.
2. После скульптинга Была сделана развертка топологии модели на плоскость в программе Torogun.
3. Далее была совершена полная ретопология низкого полигонажа модели в программе Sabstunce Painter.
4. После всех предыдущих действий материалы были перенесены в программу Cinema 4D, после чего была реализована анимация модели.
5. В итоге анимация и материалы готовой модели перенесены в игровой движок, где далее происходит манипуляция с кодом.

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		28

Приложение А

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		29

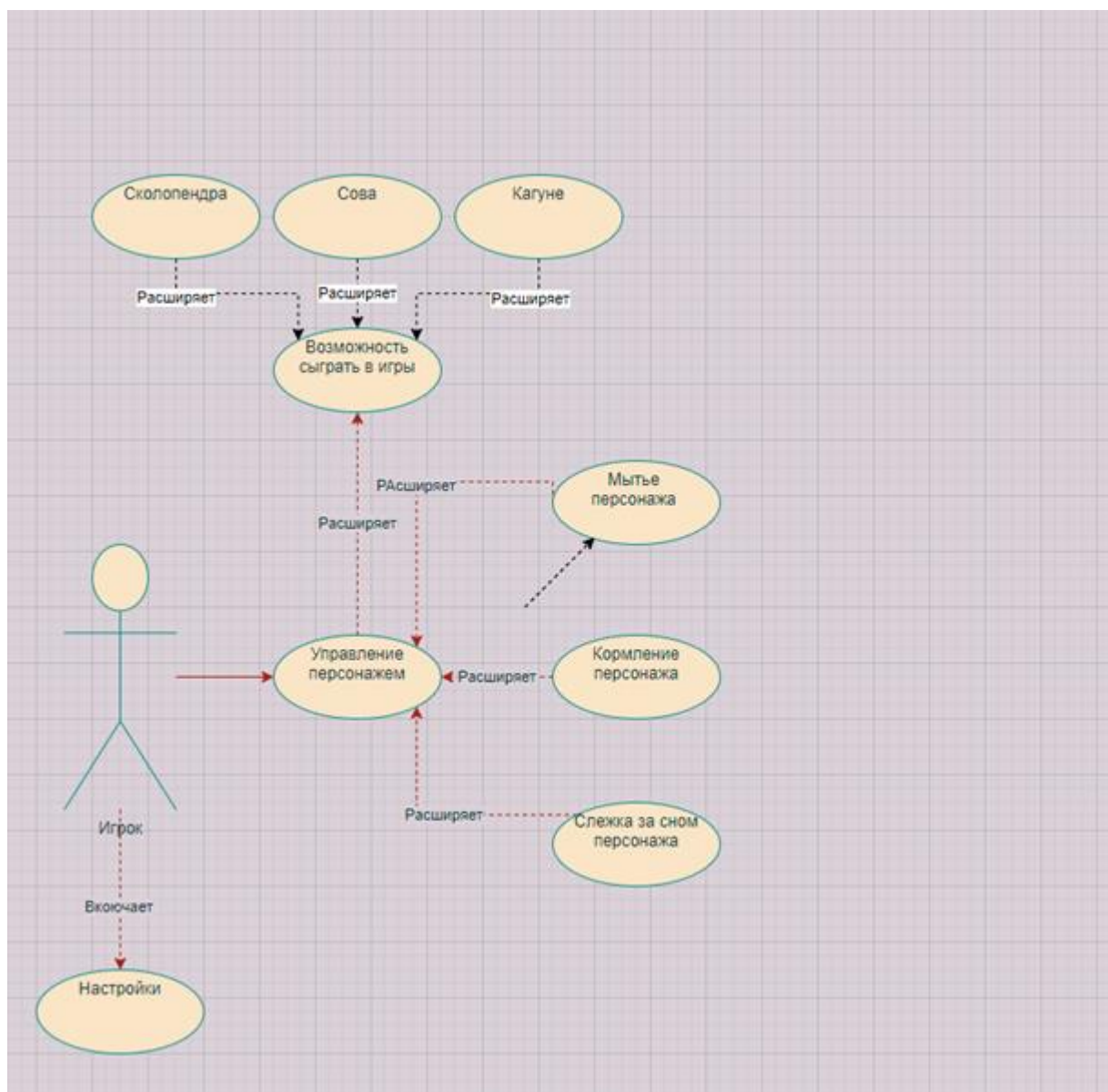


Рисунок 11 — Диаграмма вариантов использования

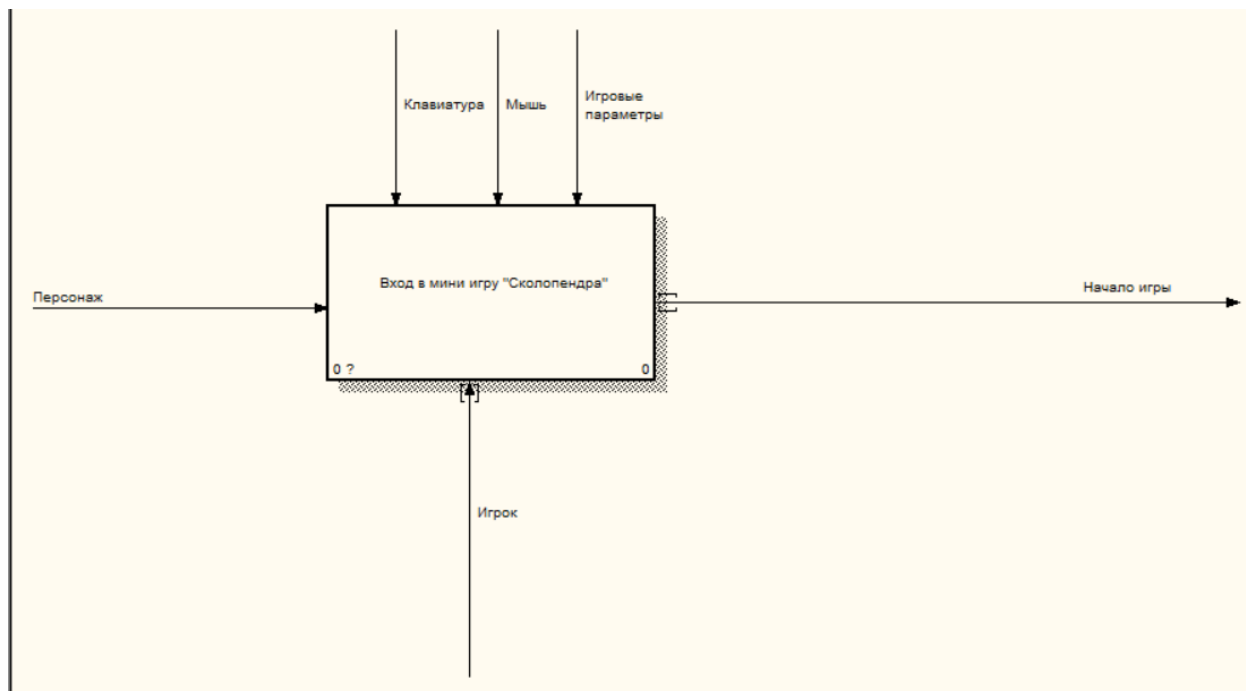


Рисунок 12 - Функциональная модель (1)

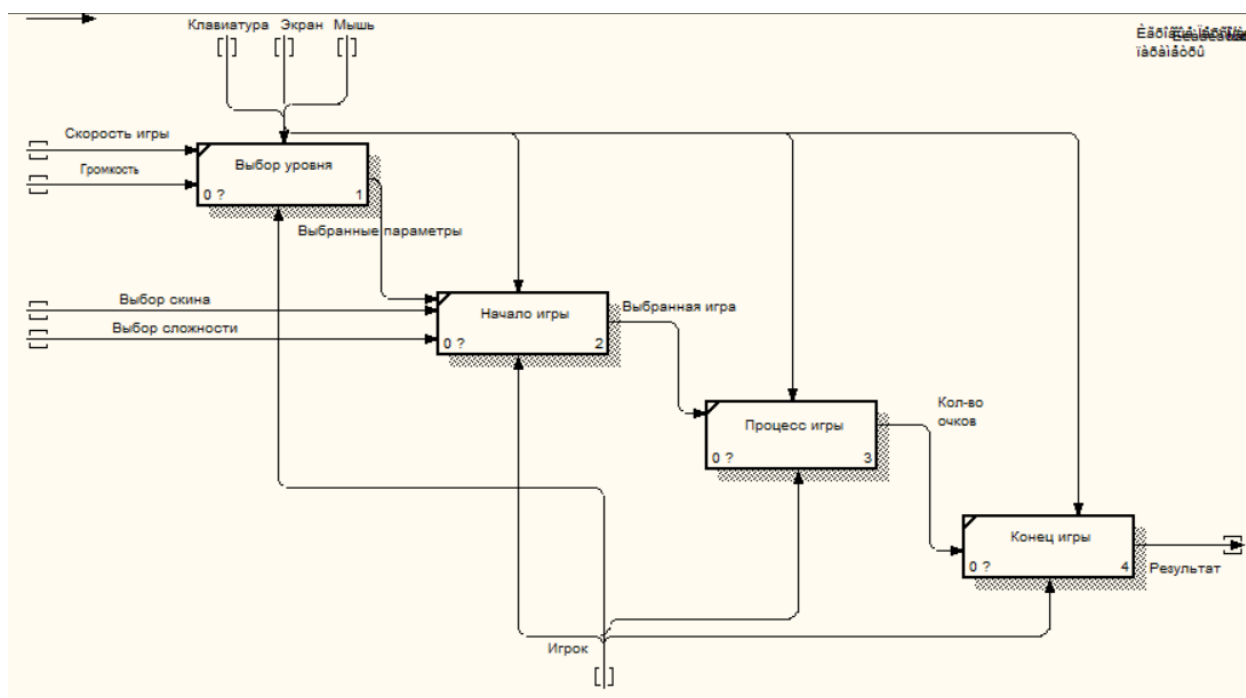


Рисунок 13 - Функциональная модель (2)

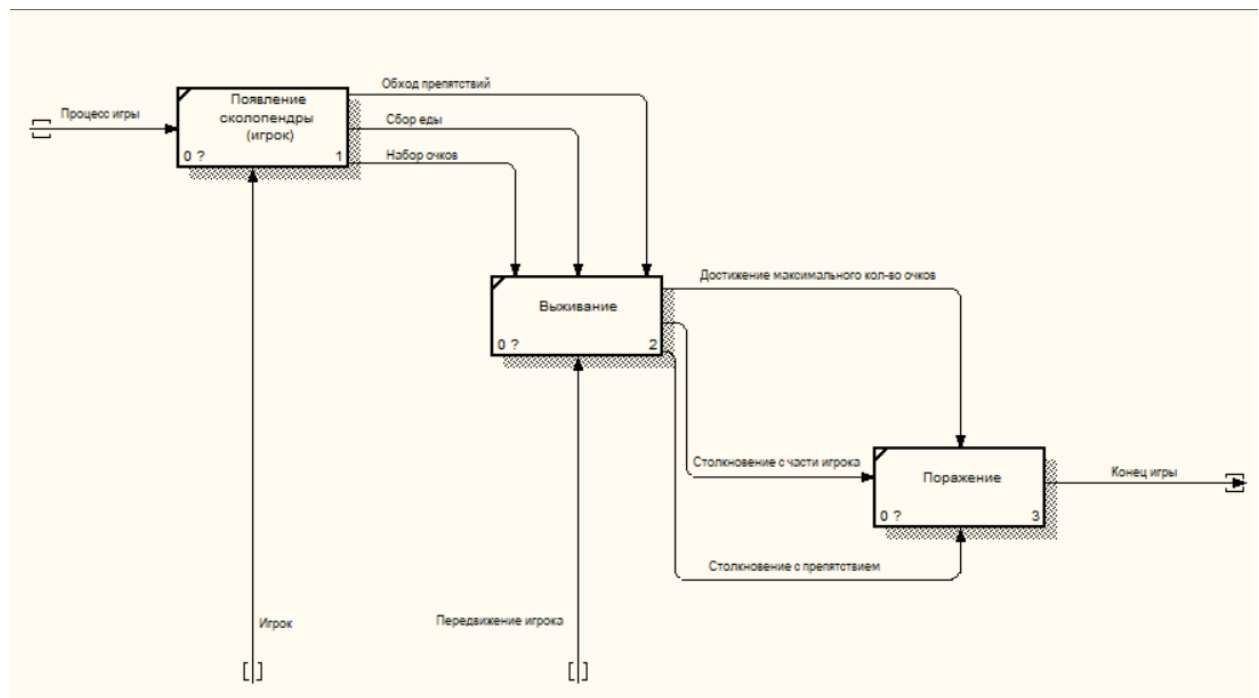


Рисунок 14 - Функциональная модель (3)

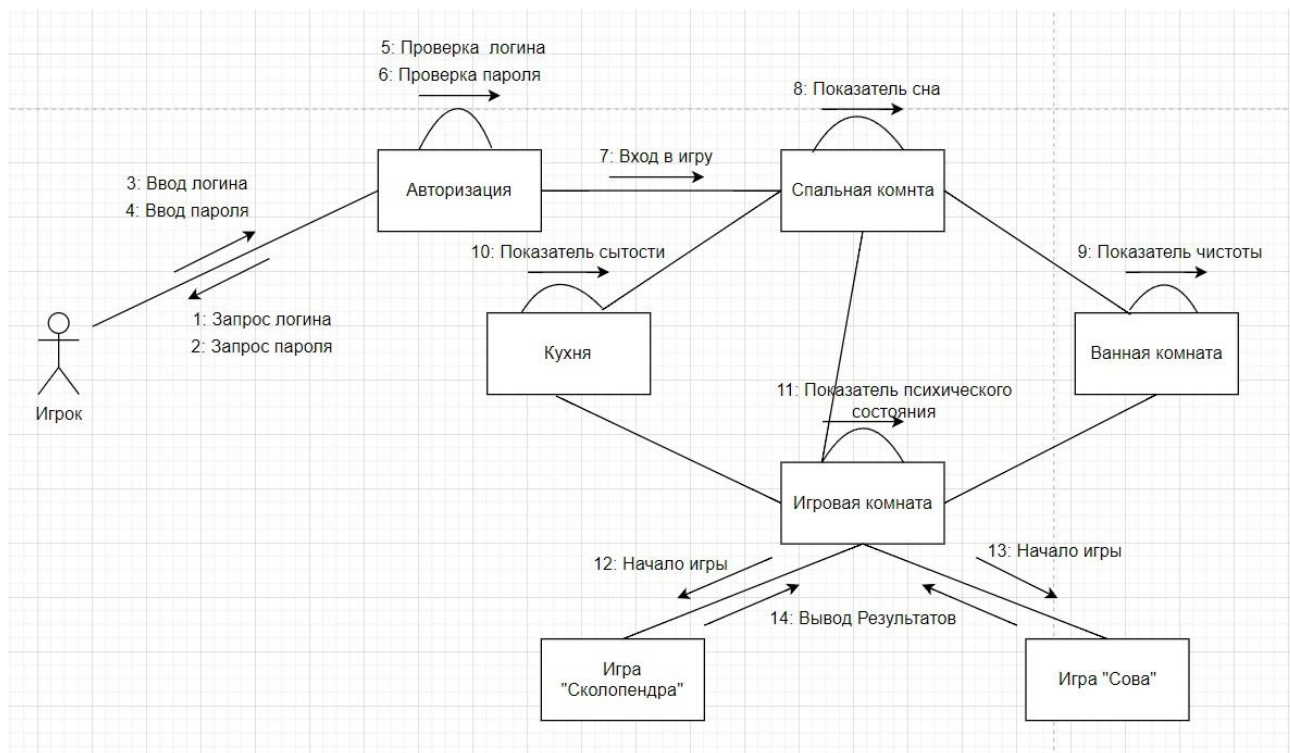


Рисунок 15 — Модель данных

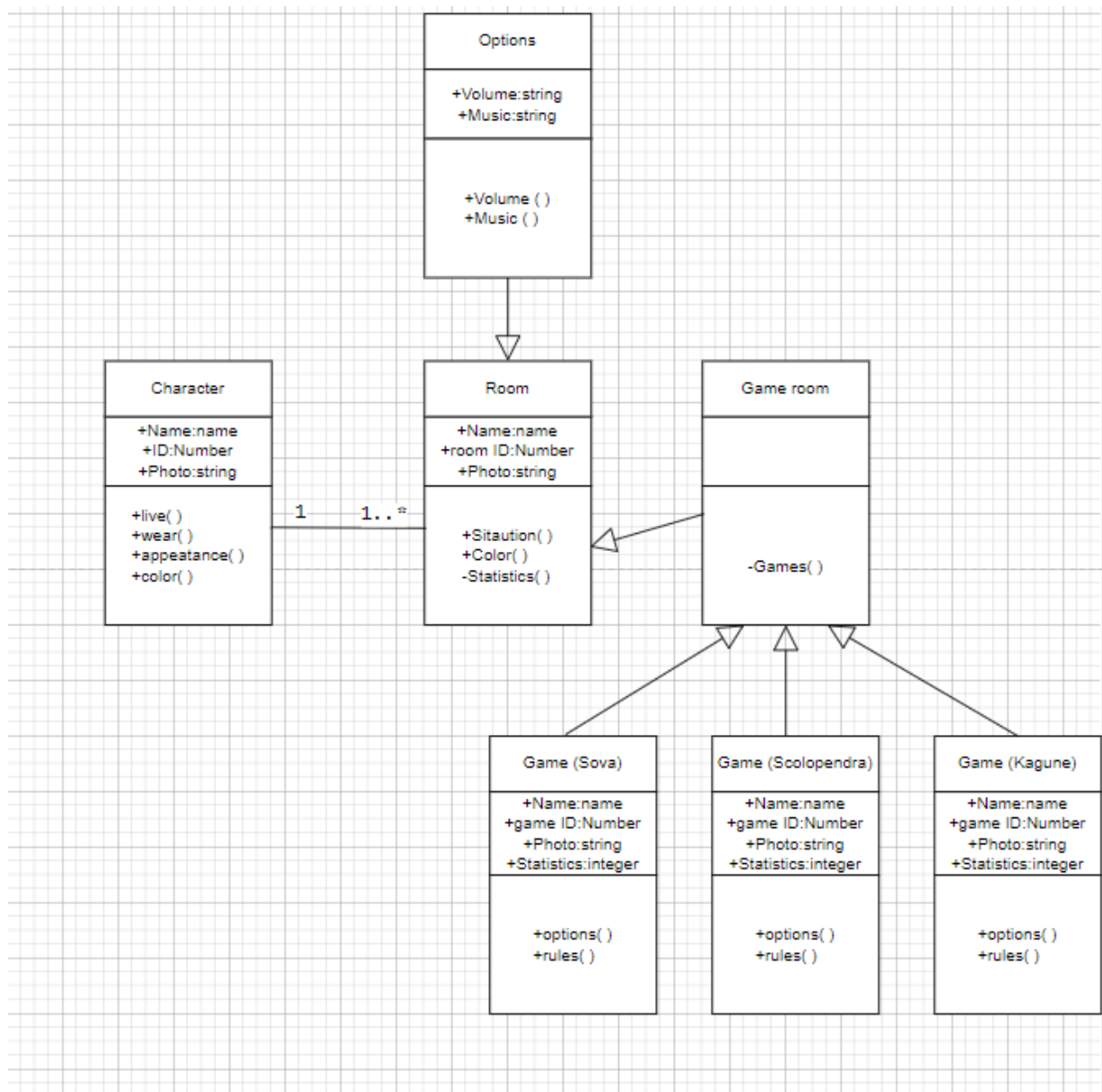


Рисунок 16 — Диаграмма объектов

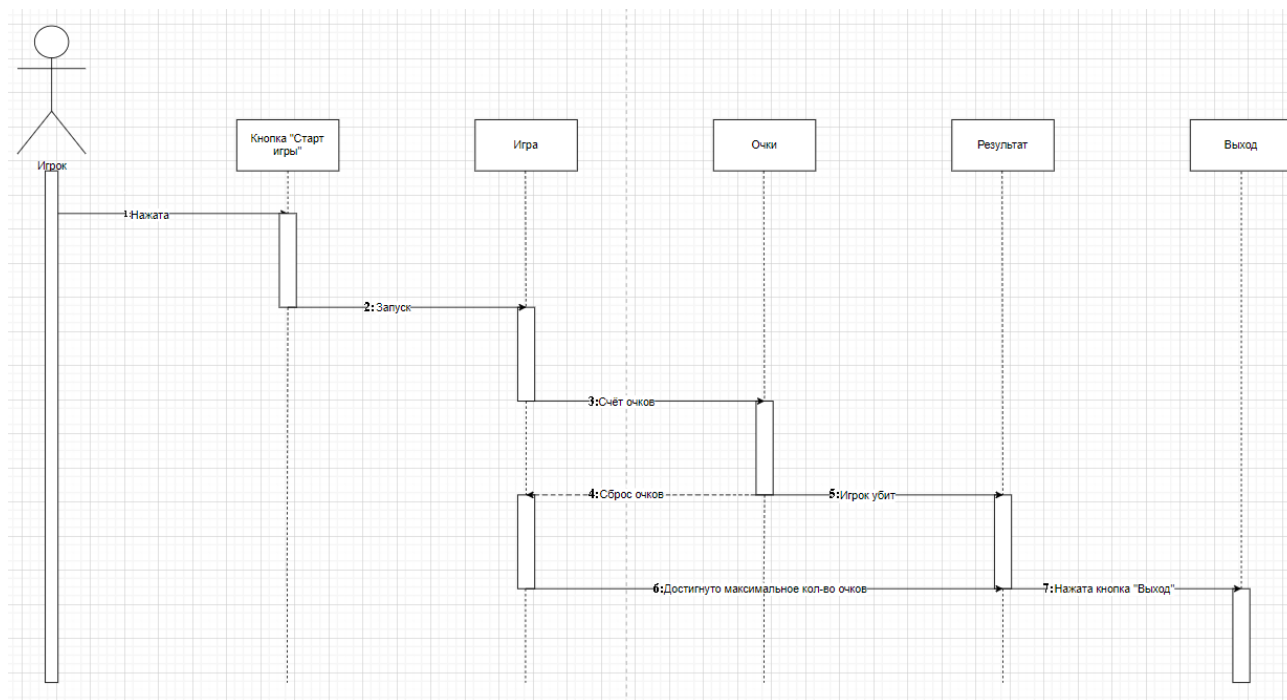


Рисунок 17 — Диаграмма последовательности

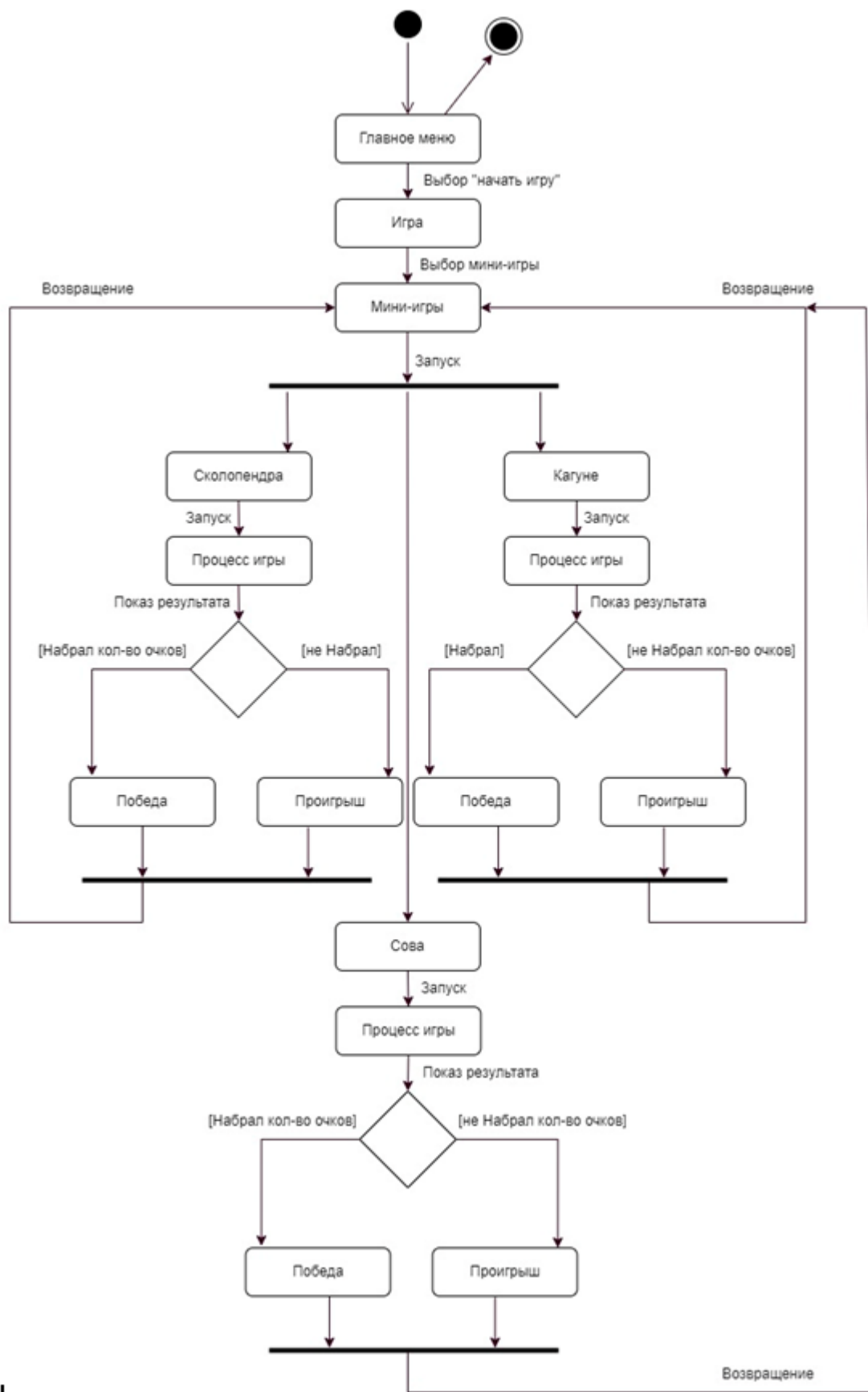


Рисунок 18 — Диаграмма деятельности

Изм.	Лист	№ докум.	Подпись	Дата

ТРПО 2-40 01 01.32.36.04.21 ТЗ

Лист

36

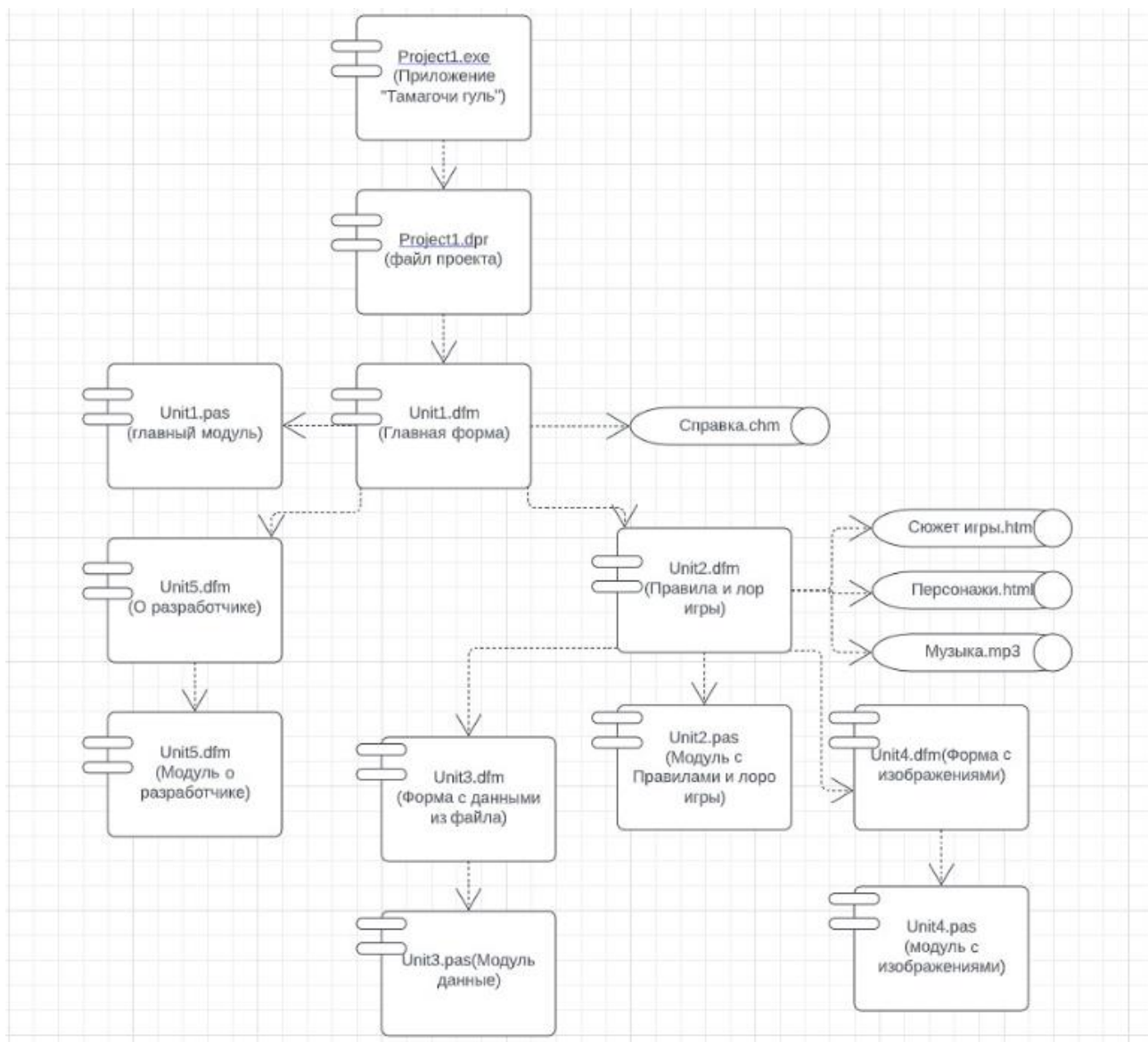


Рисунок 19 – Диаграмма компонентов

2. Прототип III

Функции модели:

1. Может отражать различные шейдеры



Рисунок – 20

2. Ретопология построена под текстурирование и анимацию

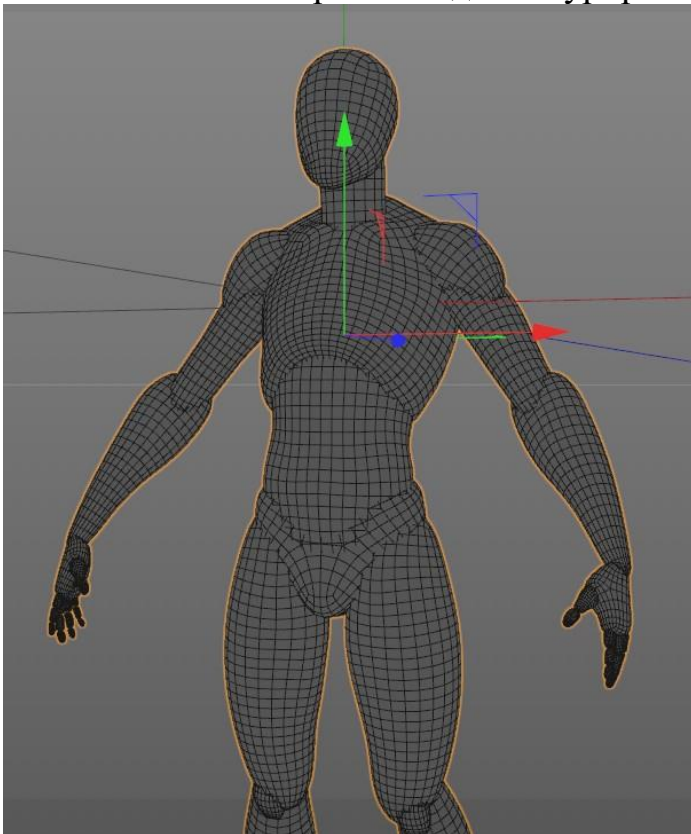


Рисунок - 21

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докм.	Подпись	Дата		38

3.Анимация бега

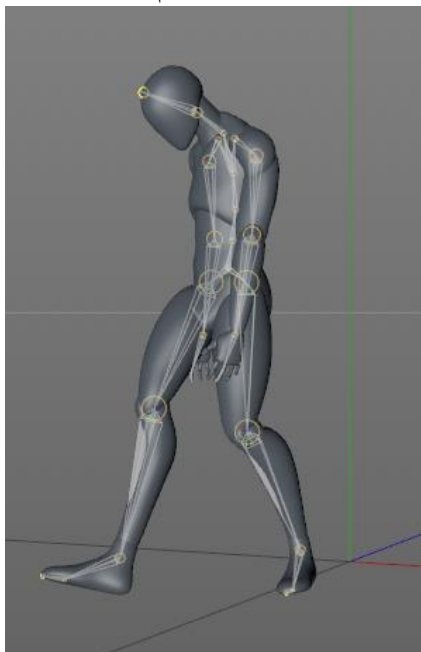


Рисунок - 22

4.Анимация танца

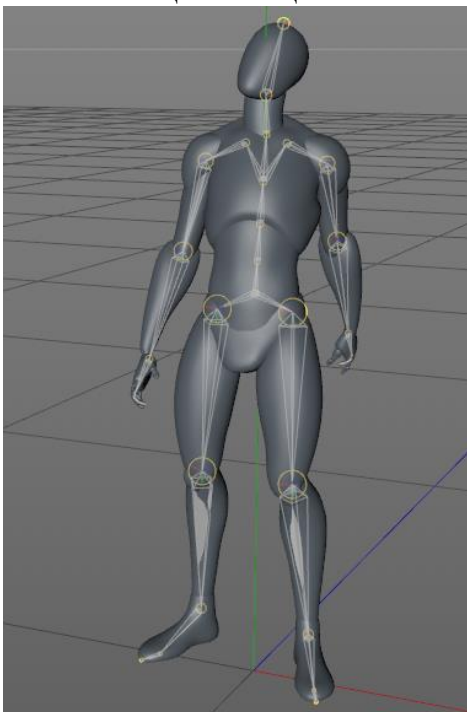


Рисунок - 23

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		39

5. Анимация удара

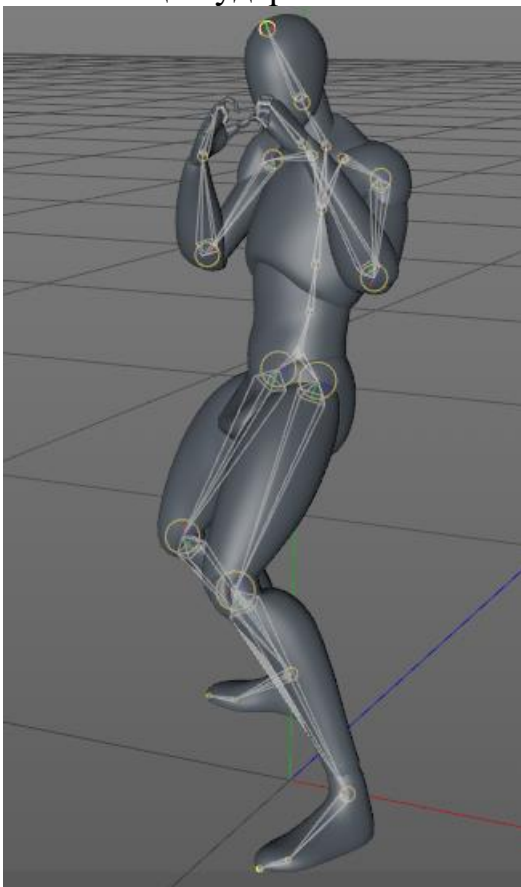


Рисунок - 24

6. Анимация ходьбы

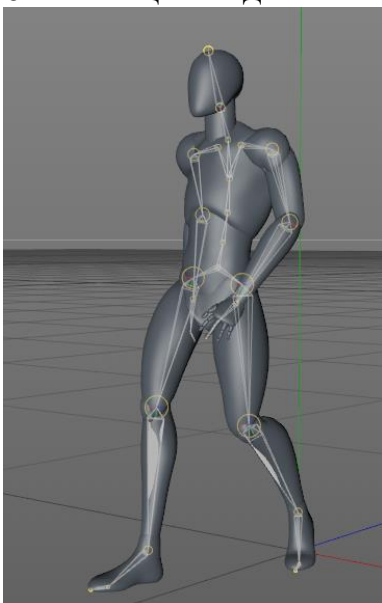


Рисунок – 25

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		40

7.Анимация стойки

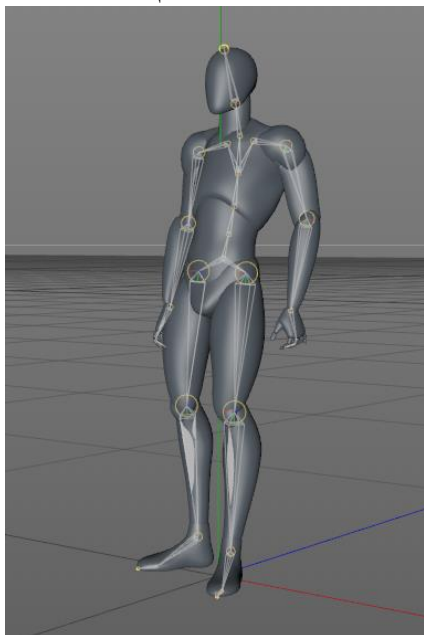


Рисунок - 26

8.Риггинг сделан под анимацию

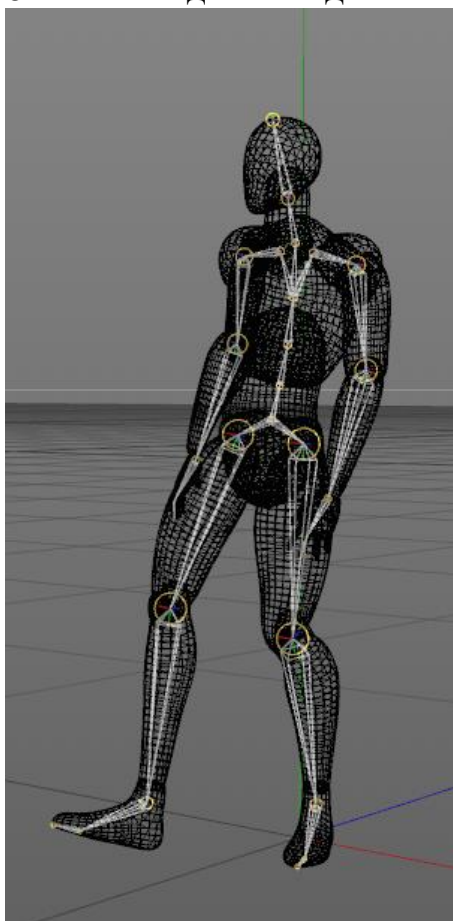


Рисунок - 27

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		41

9.Текстуры и шейдинг модели нормально отражает глобальное освещение

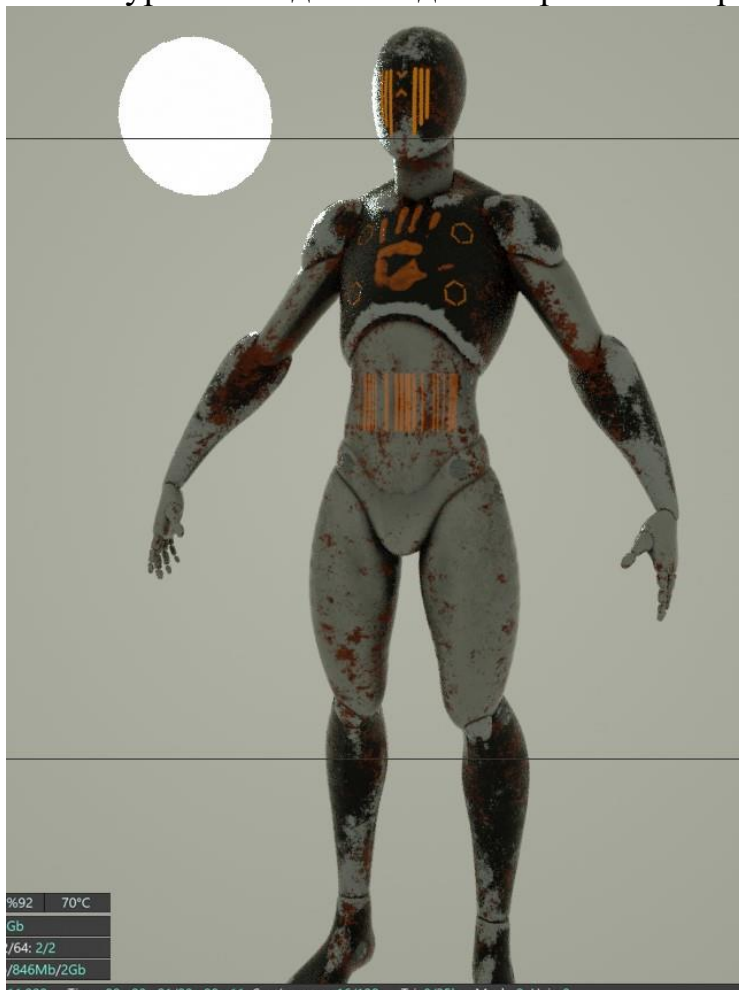


Рисунок – 28

10.Модель имеет лоуполи и хайполи полигонаж

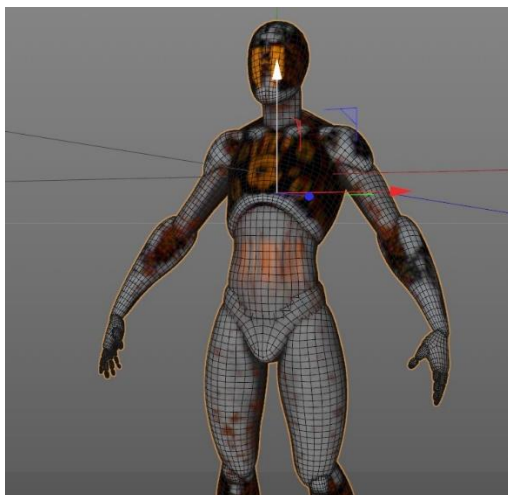


Рисунок - 29

					ТРПО 2-40 01 01.32.36.04.21 ТЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		42