

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ И  
УПРАВЛЕНИЯ им. К.Г. РАЗУМОВСКОГО (ПЕРВЫЙ КАЗАЧИЙ  
УНИВЕРСИТЕТ)

Кафедра информационных систем и цифровых технологий

Направление подготовки 09.03.03 Прикладная информатика

Направленность (профиль) Проектирование программного обеспечения для  
бизнеса в пищевой промышленности и отраслях агропромышленного комплекса

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
*(бакалаврская работа)*

«Разработка компьютерной игры в жанре crafting»

Обучающийся

Егоркин А.Д.

подпись

(фамилия, инициалы)

Руководитель выпускной  
квалификационной работы

к.т.н., доцент, Зырянова С.А.

подпись

(ученая степень, ученое звание, фамилия, инициалы)

Руководитель ОПОП

к.т.н., доцент, Зырянова С.А.

подпись

(ученая степень, ученое звание, фамилия, инициалы)

**Рекомендовать к защите**  
Заведующий кафедрой ИСиЦТ  
д.п.н., проф. \_\_\_\_\_ /Чванова М.С.  
(подпись)

Протокол заседания кафедры от  
« \_\_\_\_\_ » \_\_\_\_\_ 2024 г. № \_\_\_\_\_

Москва 2024 г.

Федеральное государственное бюджетное образовательное учреждение  
высшего образования «Московский государственный университет технологий  
и управления им. К.Г. Разумовского (Первый казачий университет)»

Факультет цифровых технологий

Кафедра информационных систем и цифровых технологий

Направление подготовки 09.03.03 Прикладная информатика

УТВЕРЖДАЮ

Зав. кафедрой

\_\_\_\_\_/д.п.н., проф., Чванова М.С.

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

**ЗАДАНИЕ  
НА ВЫПОЛНЕНИЕ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

1. Обучающемуся Егоркину Антон Дмитриевичу

2. Тема: Разработка компьютерной игры в жанре crafting

3. Срок сдачи ВКР: \_\_\_\_\_

4. Исходные данные к работе: фундаментальные научные труды, учебная литература, справочная литература, нормативные правовые документы, документы организации – базы преддипломной практики

5. Содержание ВКР:

Введение

Глава 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Глава 2. ПРАКТИЧЕСКАЯ ЧАСТЬ

Заключение

Список использованных источников

Дата выдачи задания \_\_\_\_\_

Руководитель ВКР \_\_\_\_\_ к.т.н., доцент, Зырянова С.А.

(подпись)

Задание принял к исполнению \_\_\_\_\_ Егоркин А.Д.

## КАЛЕНДАРНЫЙ ГРАФИК ВЫПОЛНЕНИЯ ВКР

№ раздела	Содержание раздела	Срок	
		Начала	Окончания
1.	Подбор и изучение литературы		
2.	Разработка 1-й главы		
3.	Проверка 1-й главы руководителем и доработка ее по замечаниям руководителя		
4.	Разработка 2-й главы		
5.	Проверка 2-й главы руководителем и доработка ее по замечаниям руководителя		
6.	Написание введения и заключения, представление законченной работы руководителю		
7.	Проверка руководителем законченной работы		
8.	Предзащита		
9.	Доработка ВКР по замечаниям руководителя и оформление работы		
10.	Представление законченной работы на кафедру		

Руководитель ВКР \_\_\_\_\_ к.т.н., доцент, Зырянова С.А.

*(подпись)*

Обучающийся \_\_\_\_\_ Егоркин А.Д.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	5
1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ.....	8
1.1 Анализ предметной области.....	8
1.2 Описание изучаемого объекта.....	11
1.3 Аргументация преимуществ выбранного проекта.....	15
1.4 Средства разработки приложений.....	16
1.4.1 Среда разработки Unity.....	17
1.4.2 Среда разработки Unreal.....	19
1.4.3 Среда разработки GameMaker.....	19
1.5 Сравнение аналогов.....	21
1.5.1 Игра Mindustry .....	21
1.5.2 Игра Крафт-кликер .....	23
1.6 Заключительный анализ результатов.....	24
2. ПРАКТИЧЕСКАЯ ЧАСТЬ.....	25
2.1 Моделирование предметной области.....	25
2.1.1 Создание мира .....	25
2.1.2 Разработка игровых объектов.....	27
2.1.3 Игровые механики .....	29
2.1.4 Технологическая реализация.....	32
2.2 Разработка и реализация прототипа.....	33
2.2.1 Подготовка рабочей среды.....	33
2.2.2 Создание игровой среды .....	34
2.2.3 Реализация игровых механик .....	36
2.2.4 Разработка интерфейса пользователя .....	44
2.2.5 Оптимизация игры .....	48
2.2.6 Тестирование.....	50
2.3 Перспективы развития программного продукта .....	53
ЗАКЛЮЧЕНИЕ.....	59
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	61

## ВВЕДЕНИЕ

За последние годы видеоигры укрепили свою роль в культурном ландшафте, открывая перед геймерами двери в захватывающие цифровые вселенные, где они испытывают эпические приключения. Экспансия гейминг-индустрии оказала глубокое влияние на различные сегменты повседневной жизни, подталкивая к изменениям в способах общения, обучении и профессиональном росте. Видеоигры, став превосходным источником развлечения, креатива и личного выражения, привлекают аудиторию счётных миллионов из разных уголков планеты. Их широчайший спектр жанров и игровых механик могут удовлетворить самые разнообразные вкусы и хобби.

Задачей данной дипломной работы является всесторонний анализ процессов разработки компьютерных игр в категории *crafting* с применением передовых программных и технологических инструментов. Особое внимание сосредоточено на изучении и систематизации важнейших этапов дизайна, программирования, и дальнейшего включения в игру уникальных механик, целями которых являются формирование захватывающего геймплея и предоставление геймерам простора для инициативы и творческого самовыражения в рамках симуляции. Жанр крафтинг выделяется уникальным акцентом на процессах создания и трансформации игровых предметов, построек и ландшафтов, что напрямую способствует реализации идей и мечтаний игроков в виртуальном измерении, предлагая им безграничные возможности для креативности и собственноручного создания уникального игрового опыта.

В сфере видеоигр, жанр крафтинга выделяется за счет своей вовлеченности и адаптивности. Проекты вроде *Minecraft*, *Terraria* и *Stardew Valley*, наряду с другими, заслужили популярность благодаря возможностям строительства, исследований и персонализации виртуальных пространств. Они активизируют фантазию, инновационное мышление и умение находить решения, обеспечивая основу для неограниченных творческих подвигов и новаторских подходов.

В исследовании анализируются как инженерные, так и творческие элементы в разработке игр жанра «крафтинг», рассматривая их роль в динамике и взаимосвязи с пользователем. Акцент делается на механизмах создания объектов в игре, обзоре системы крафтинга, изучении прогрессии игрока и эволюции игровой среды. Исследуются разнообразные стратегии воплощения указанных элементов, в том числе их влияние на игровой дизайн и синергию с остальными аспектами геймплея.

В процессе исследования будут детально изучены различные элементы, начиная от выбора подходящего игрового движка, занимаясь разработкой увлекательного игрового мира, оттачивая игровые механики и системы создания предметов (crafting), и заканчивая глубоким анализом психологических и визуальных факторов, которые делают крафтинговые игры особенно привлекательными для аудитории. Что касается технической стороны исследования, она будет включать в себя тщательное исследование инструментария и возможностей разработчика игр GameMaker, а также оценку его функциональных особенностей и удобства использования в контексте создания игр данного жанра.

Исследуемым процессом выступают этапы создания игрового приложения, а в качестве предмета изучения выбраны методики и подходы к разработке игрового приложения через платформу GameMaker. Задачей дипломной работы является разработка и реализация игрового приложения в категории crafting. Для достижения этой цели предусмотрено выполнение ряда конкретных задач:

1. Осуществить изучение особенностей создания компьютерной игры для платформы персональных компьютеров.
2. Осуществить изучение и подбор инструментариев для создания игрового приложения.
3. Выполнить сравнительный анализ схожих игровых приложений.
4. Анализировать и уточнить стадии разработки и дизайна игрового приложения.

5. Создать игровой проект с использованием платформы разработки GameMaker.

6. Выполнить верификацию функциональности разработанного игрового программного продукта.

Исследование уникальных аспектов создания игрового приложения для компьютеров включает оценку технических параметров и особенностей данной платформы, например, вычислительной мощности, визуальных качеств и пользовательских интерфейсов. Определение инструментария для разработки базируется на анализе и сопоставлении разнообразных игровых движков и инструментов разработки, оценивая их по критериям функционала, простоты в применении и предоставляемой поддержки разработчикам. Изучение существующих игр в жанре крафтинга помогает идентифицировать эффективные стратегии и игровые механики, которые нашли признание у аудитории, что способствует их адаптации и внедрению в рамках разработки нового проекта.

Процесс разработки игрового приложения охватывает всестороннее проектирование, начиная с идеи и визуализации, и завершая воплощением в жизнь и верификацией. Ключевым аспектом является создание механизма крафтинга, предполагающего разработку множества формул и методик производства предметов. Проведение тестирования завершеного продукта критично для выявления дефектов, обеспечения его надежности и простоты в использовании.

Следовательно, целью данного исследования является комплексный анализ разработки игры в категории крафтинг через платформу GameMaker, что обеспечит приобретение ключевых знаний и умений в сфере создания игр и кодирования, а также воплотит в жизнь захватывающий проект, привлекающий внимание разнообразной аудитории геймеров.

# 1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

## 1.1 Анализ предметной области

Crafting в игровой индустрии обозначает процесс, при котором игроки активно взаимодействуют с игровым пространством через сбор, обработку и использование ресурсов для создания новых предметов или улучшения существующих. Такой подход позволяет пользователям не только воплощать свои творческие идеи, но и тщательно планировать стратегию использования ресурсов для достижения определённых игровых задач. Растущая популярность этого жанра свидетельствует о его способности удовлетворять желание игроков к инновациям, самовыражению, а также к постоянному развитию и освоению новых территорий внутри игрового мира.

Выпуск "Minecraft" в 2011 году произвел колоссальный взрыв в сегменте игр на тему крафтинга, кардинально трансформировав этот жанр. По официальным данным от Mojang Studios, к исходу 2020 года продажи "Minecraft" превысили порог в 200 миллионов экземпляров, утвердив ее статус как одной из наиболее востребованных видеоигр в истории. Этот безусловный успех стал катализатором для появления и активной разработки многочисленных проектов с аналогичным механизмом и идеей, включая "Terraria", "Starbound" и "Don't Starve".

К примеру, "Terraria" разошлась тиражом свыше 35 миллионов копий к 2021 году, как указывает Re-Logic. В игре гармонично сочетаются крафтинг, исследование мира и боевые механики, предоставляя пользователям обширные способы взаимодействия с игровой вселенной.

Ключевые компоненты, определяющие игры с элементами крафтинга, охватывают:

1. Добыча и переработка материалов: Участникам требуется добывать разнообразные материалы (лесоматериалы, горные породы, металлические



руды и др.), применяемые последующим образом в конструировании новых объектов и сооружений.

2. Процесс крафтинга: Обладая необходимыми материалами, участники игры способны производить разнообразные артефакты, включая инвентарь, вооружение, защитное снаряжение, предметы интерьера и т.д., способствуя таким образом их успеху в последующих экспедициях.
3. Эксплорация виртуальных пространств: В цифровых развлечениях миры часто создаются посредством процедурной генерации, обеспечивая неповторимость каждой сессии. Участники разведывают неизведанные зоны, добывают ресурсы и сталкиваются с разнообразными сущностями.
4. Строительство: Участники имеют возможность возводить и адаптировать сооружения, формируя индивидуальные архитектурные объекты и укрепления, обеспечивающие защиту от противников и экстремальных сред.
5. Выживание: Разнообразие игр в стиле крафтинга включает аспекты выживания, задавая игрокам необходимость контролирования здоровья, голода, уровня энергии и других статусов.

Обзор популярных игровых хитов и анализ их показателей:

#### 1. **Minecraft:**

- Общий объем продаж превысил 200 миллионов экземпляров.
- Месячная аудитория составляет свыше 140 миллионов активных участников.
- Разработчик: Mojang Studios.
- Ключевые атрибуты включают в себя просторный исследуемый мир, процедурно сгенерированную местность, возможность взаимодействия с элементами путем их строительства или демонтажа, а также поддержку совместной игры.

#### 2. **Terraria:**

- Общий объем реализации экземпляров превысил 35 миллионов.

- Разработчик: Re-Logic.
- Ключевые характеристики включают 2D-визуализацию, экспедиции в подземелья, крафтинг арсенала и защитных средств, а также битвы с основными противниками.

### 3. **Don't Starve:**

- Объем продаж составил свыше 10 миллионов экземпляров.
- Разработчик: Klei Entertainment.
- Ключевые атрибуты включают: адаптация к экстремальным условиям, изучение процедурно созданной карты, контроль за уровнем сытости и состоянием здоровья персонажа.

Индустрия игр, включая жанр создания и крафтинга, демонстрирует устойчивую тенденцию к росту, как это отмечено в исследованиях от SuperData. В течение 2023 года, доход от игр этой категории достиг отметки в 4 миллиарда долларов США, увеличившись на 15% в сравнении с предыдущим годом. Эти цифры наглядно отражают неуклонный интерес аудитории к проектам, которые предоставляют возможности для глубокого погружения в процесс создания и проявления креативности.

Дополнительно, по данным исследования Ассоциации Интерактивного Развлечения и Игр (ESA), примерно 65% населения США увлекаются играми, причем большое количество из них отдают предпочтение проектам в стиле крафтинга. Среднестатистический возраст геймеров — 34 лет, что демонстрирует заинтересованность широкой категории людей в данной категории игр.

Базовые симуляционные игры, например, симуляторы работы фабрики, играют ключевую роль для начинающих разработчиков игр. Они включают в себя простые игровые механизмы, которые легки для понимания и внедрения, обеспечивая тем самым отличную платформу для изучения и развития начальных навыков в программировании и дизайне игр. Эти виды симуляторов обычно сосредотачиваются на контроле за ресурсами и выстраивании базовых

структур, тем самым плавно вводя игроков в более комплексные элементы геймплея.

"Factorio" является отличным примером удачного применения концепции простого симулятора, который воплощает в себе сложные технические процессы. Этот симулятор строительства и управления фабриками привлек не только из-за своего интригующего геймплея, но и благодаря обширным опциям для автоматизации операций, что позволяет игрокам оптимизировать производственные цепочки. Разработчик Wube Software сообщил, что к концу 2021 года объем продаж игры превысил отметку в 2,5 миллиона копий, подтверждая ее высокую популярность среди аудитории.

Исследование данной темы выявило, что игры в категории крафтинг сохраняют высокий уровень популярности у различных категорий геймеров. Отчеты о продажах и оценки критиков свидетельствуют в пользу целесообразности создания игровых приложений в этом направлении. Симуляторы производства, например, promышляют значимым потенциалом в плане обучающих и развлекательных возможностей, что делает такие игры превосходной отправной точкой для начинающих разработчиков, а также открывает пути для их будущего роста и достижения коммерческой выгоды.

## **1.2 Описание изучаемого объекта**

В рамках проведения данного исследования рассматривается игровая программа, относящаяся к категории крафтинговых игр, которая представляет из себя базовый индустриальный симулятор. Преимущественный интерес к этому объекту определяется рядом аспектов:

1. **Обучающая значимость:** Разработка базового фабричного симулятора дает новичкам в игроразработке шанс освоить фундаментальные аспекты игрового дизайна и кодирования. Эти начальные проекты обеспечивают не только приобретение практического опыта, но и понимание фундаментов разработки видеоигр.

2. Удержание интереса к жанру: Сегмент крафтинга продолжает набирать популярность среди геймеров благодаря глубоким механикам создания и модификации, обеспечивающим интенсивное погружение и возможности для личного выражения, что делает его притягательным для разнообразных игровых сообществ.
3. Легкость в разработке: Создание игры-симулятора производства не требует внедрения сложной игровой механики и высококачественной графики, что делает проект выполнимым даже для новичков в области программирования. Ключевые аспекты, такие как добыча ресурсов, крафтинг предметов и автоматизация рабочих процессов, могут быть успешно реализованы при помощи базового набора инструментов и платформ для разработки.

Разрабатываемое игровое приложение характеризуется следующими параметрами:

1. Жанр: Симулятор строительства завода.
2. Целевые пользователи: Начинающие программисты и разработчики игр, в дополнение к геймерам, увлеченным crafting жанром.
3. Платформы: Компьютер (ОС Windows, MacOS), а также возможность адаптации для мобильных платформ.
4. Инструмент для создания игр: GameMaker Studio выделяется своей доступностью и удобством обучения среди платформ для разработки видеоигр.

Ключевые компоненты мобильной игры:

1. Игровые объекты:

Буровые установки: Извлекают полезные ископаемые (к примеру, железную руду) из месторождений.

Печи: Превращают извлечённую руду в металлические слитки.

Автоматизированные сборочные линии: Конструирование итоговых изделий на основе переработанных материалов.

## 2. Игровые механики:

**Добыча ресурсов:** Участники контролируют буровые установки для экстракции минералов.

**Процесс переработки материалов:** Металлическая руда транспортируется в плавильные печи, где она преобразуется в металлические слитки.

**Производство изделий:** Изготовление финальных продуктов из слитков осуществляется при помощи автоматизированных сборочных линий.

Автоматизация производственных процессов позволяет пользователям конфигурировать системы для авто-добычи ресурсов, их обработки и сборки, облегчая тем самым контроль над производственными цепочками.

### Особенности игрового процесса

1. **Удобство контроля:** Участники без труда манипулируют игровыми элементами с помощью мыши и интерфейса, обеспечивая непринуждённый доступ к игровому процессу для новичков.
2. **Инструктивные компоненты:** В игре присутствуют инструкции и tutorиалы, демонстрирующие ключевые концепции функционирования объектов и игровых механик.
3. **Прогресс в игре:** Участники стартуют с минимальным инвентарем и базовыми миссиями, методично продвигаясь к более комплексным операциям и автоматизации процессов производства.
4. **Адаптивность и кастомизация:** В игре предусмотрена возможность для пользователей тестировать разнообразные конфигурации и изменения в элементах для улучшения эффективности процессов.

В процессе создания нашего приложения мы примем за основу установленную жанровую классификацию игр, сжатое описание которой содержится в таблице 1.

Таблица 1 –Классификация видов компьютерных игр

Жанр	Синонимы	Описание
Adventure	Приключения	Изучение виртуального пространства, разгадывание загадок.
Strategy	Стратегия	Ресурсное управление, стратегическое планирование, тактическое маневрирование
Racing	Гонки, рейсинг	Конкурсы на быстрое прохождение этапов
Action	Экшен, экшн	Динамичный игровой процесс, акцент на моторные навыки, точность оценки расстояния, оперативность в принятии решений и быстрота отклика.
RPG	Игровое воплощение роли, Roguelike жанр.	Фокус на прокачку основного героя игры
Crafting	Крафтинг, песочница	Создание и усовершенствование производственного процесса, внедрение автоматизированных систем.

В рамках выполнения выпускной квалификационной работы, предстоит разработка программного продукта, классифицируемого жанром крафтинга.

### 1.3 Аргументация преимуществ выбранного проекта

Перед тем как приступить к созданию игрового приложения, разработчики сталкиваются с задачей определения подходящей платформы разработки, выбором игрового движка и языка программирования, который лучше всего подойдет для реализации всех заложенных в проект игровых функций.

Следовательно, для того чтобы определиться с наиболее подходящим программным продуктом, важно дать ответы на нижеследующие запросы:

- какой будет графика в игре;
- каковы будут преобладающий жанр и, возможно, специфические поджанры предполагаемой видеоигры;
- какие ключевые игровые механики будут представлены;
- установить окончательную миссию проекта;
- навык разработчика в области создания игр.

Правильные ответы на вопросы позволяют уточнить требуемые ресурсы для создания проекта, выбор оптимальной IDE (Integrated Development Environment) и сформировать базовое понимание конечного продукта. В то же время, ошибки в ответах могут вызвать проблемы, замедляющие процесс разработки, ведущие к багам, и ограничивающие реализацию специфических функций и игровых механик.

Из анализа ответов на предыдущие вопросы возможно выявить цели, отражающие в общих чертах контуры задуманной игры, набор игровых механизмов, жанровую принадлежность и, в итоге, определить стратегическую цель разработки и выбор оптимального игрового движка.

Следовательно, определяются следующие цели:

- Двухмерная пиксельная графика, отображение сверху.
- Основной жанр - карфтинг, поджанр - управление ресурсами;
- Ключевые игровые механики включают генерацию основных материалов по достижении специфических критериев, их транспортировку по

игровой арене, а также крафтинг новых объектов, используя заданный ассортимент ресурсов.

– Основная задача разработки - разработать игру, сфокусированную на демонстрации игроком своих интеллектуальных навыков, способности эффективно управлять доступными ресурсами и успешно организовывать производственные процессы;

– компетенции программиста - базовые.

#### **1.4 Средства разработки приложений**

Выбор инструментария для разработки приложений стоит на первом месте в списке приоритетов. Обычно предпочтение отдаётся уже существующим решениям, но для больших проектов иногда разрабатывают уникальные игровые движки. Этот термин описывает комплекс программных инструментов и сервисов, построенных и интегрированных в единую систему. Несмотря на разнообразие функций, типичный игровой движок включает поддержку графического рендеринга, физики, аудио, скриптинга, сетевого взаимодействия, управления памятью и обработки параллельных задач. Применение одного и того же движка для разработки различных игр позволяет значительно уменьшить затраты и ускорить процесс разработки.

При работе с выбранным игровым движком встаёт задача подбора подходящего программного языка для скриптов. Рынок предлагает обширный спектр средств для создания игр от основы, включая в себя как встроенные функции, так и возможность интеграции материалов, разработанных с помощью стороннего ПО.

Обычно, эти программные продукты охватывают ряд компонентов для управления разносторонними элементами игры, включая визуализацию, аудиоэффекты, мультиплатформенность и скриптинг.

Каждый игровой движок, в основе своей, функционирует как комплексный редактор кода. В ходе создания игр разработчики зачастую вынуждены



разрабатывать индивидуальные подсистемы, процесс который может оказаться времязатратным и финансово обременительным. Однако, альтернативой служат стандартные игровые движки.

Давайте проанализируем несколько ведущих инструментов для программирования, актуальных в настоящее время.

#### **1.4.1 Среда разработки Unity**

Unity представляет собой эффективную платформу для разработки игровых и интерактивных проектов. Рассмотрим преимущества и недостатки применения Unity в процессе создания игр.

Плюсы:

1. Мультиплатформенный подход: Платформа Unity обладает функционалом разработки приложений, адаптирующихся под множество устройств, таких как смартфоны, персональные компьютеры, игровые консоли, а также системы виртуальной реальности.

2. Расширенная экосистема: Unity предоставляет обширную коллекцию готовых ресурсов, активно участвующих сообществ программистов, а также Unity Asset Store - платформу для приобретения предварительно разработанных компонентов и активов, необходимых для реализации проекта.

3. Визуальное программирование: Unity оснащено эффективным графическим пользовательским интерфейсом для создания и модификации игровых элементов, движений, визуализаций и пользовательского интерфейса.

4. Обучаемость: Unity отличается простотой освоения благодаря обширной документации и наличию учебных материалов, что делает его идеальным для новичков в сфере разработки.

Минусы:

1. Производительность: даже при внедрении усовершенствований, Unity иногда сталкивается с трудностями в обеспечении высокой производительности на мобильных платформах или при обработке больших массивов данных.

2. Ограниченное влияние на низкоуровневые операции: в определенных ситуациях у программиста может оказаться недостаточно возможностей для точной настройки и контроля элементарных компонентов приложения.

3. Ограничения вследствие привязки к Unity: Опираясь на Unity в проектах, разработчики могут столкнуться с проблемами, связанными с ограниченным выбором платформ и инструментария, что потенциально затрудняет адаптацию и масштабирование в дальнейшем.

4. Объем файлов: Проекты, разрабатываемые через платформу Unity, склонны к более значительному размеру файлов из-за интеграции требуемых библиотек и ассетов.

В общем, Unity является эффективным средством для создания игр и программ, располагая обширными функциональными возможностями и поддерживая множество платформ, однако сталкивается с определёнными ограничениями и недостатками, что важно учитывать при выборе инструмента разработки.

#### **1.4.2 Среда разработки Unreal**

Unreal Engine представляет собой высокопроизводительный инструментарий для разработки игр и создания реалистичных визуализаций. Давайте рассмотрим преимущества и недостатки применения Unreal Engine в игровой разработке.

Плюсы:

1. Фотореалистичная графика: Unreal Engine восхваляется за выдающиеся визуальные достижения, обеспечиваемые прогрессивными графическими функциями и визуальными эффектами.

2. Адаптивность и расширяемость: Unreal Engine идеально подхватывает как индивидуальные инди-разработки, так и объемные проекты AAA-класса, включая создание VR и AR.

3. Открытость и доступ к исходному коду: Unreal Engine дает свободу использования исходного кода, обеспечивая возможность его полной модификации и адаптации функционала.

4. Комплексные решения: Unreal Engine предоставляет обширный набор инструментария для создания игр, объединяя в себе редактирование мира, анимационные возможности, физическую симуляцию и прочие компоненты.

Минусы:

1. Изучение Unreal Engine может оказаться непростым заданием для начинающих, учитывая широкие возможности движка и высокие требования к графической производительности.

2. Ресурсоемкость: Разработка проектов высокой четкости в Unreal Engine часто требует использования высокопроизводительного оборудования из-за значительных системных требований.

3. Объем сборок: При разработке в Unreal Engine объем итоговых приложений может быть значительным, так как в них интегрируются обширные наборы визуальных и других медиаресурсов.

4. Плата за лицензию: Помимо бесплатного доступа к Unreal Engine для всех, при получении дохода от проектов, созданных на его основе, начисляются роялти.

В общем, Unreal Engine является эффективной платформой для разработки высокодетализированных игровых и визуальных проектов, однако его освоение требует значительного времени и усилий, а также может влечь за собой необходимость в инвестициях в высокопроизводительное оборудование.

### **1.4.3 Среда разработки GameMaker**

GameMaker Studio представляет собой интегрированную среду разработки, нацеленную на производство двумерных видеоигр. Давайте рассмотрим преимущества и недостатки применения GameMaker в процессе разработки игровых проектов.

#### Плюсы:

1. Удобство в освоении: GameMaker Studio выделяется своим пользовательским интерфейсом, который легко осваивается, и простым в изучении программным языком, делая его идеальным вариантом для новичков в разработке игр.

2. Ускоренная прототипирование игр: GameMaker Studio обеспечивает эффективную разработку начальных версий игр, предоставляя обширную коллекцию предварительно созданных ассетов и функциональных инструментов.

3. Поддержка многоплатформенности: GameMaker Studio обеспечивает создание игр для широкого спектра устройств, включая персональные компьютеры, смартфоны, планшеты и игровые консоли.

4. Живое сообщество: В рамках GameMaker Studio функционирует динамичное сообщество программистов, предлагающее взаимопомощь, возможности для обмена знаниями и доступ к ценным материалам.

#### Минусы:

1. Ограниченная функциональность для разработки 3D-игр: GameMaker Studio изначально задумывалась как платформа для разработки 2D-игр, из-за чего разработчики, стремящиеся создать трехмерные игровые проекты, могут столкнуться с ограничениями в доступных инструментах и возможностях.

2. Урезанные возможности: По сравнению с различными интегрированными средами разработки, GameMaker Studio может обладать ограниченными инструментами и потенциалом для масштабирования проектов.

3. Эффективность: Вопреки оптимизациям, GameMaker Studio может столкнуться с трудностями в обеспечении высокого уровня эффективности при реализации сложных проектов или обработке больших массивов данных.

4. Ограниченность ресурсной базы сообщества: У GameMaker Studio относительно меньшее сообщество разработчиков по сравнению с

альтернативными платформами разработки игр, что может привести к суженной доступности поддержки и ресурсов.

GameMaker Studio является превосходной платформой для разработки двумерных видеоигр, идеально подходит для начинающих разработчиков, однако она может не подойти для реализации более амбициозных проектов или создания игр с трехмерной графикой.

## **1.5 Сравнение аналогов**

Крафтинг-жанр, часто обозначаемый как игры с элементами создания или "sandbox"-игры, охватывает цифровые развлечения на ПК и мобильных устройствах, где главная задача направлена на проектирование и изготовление предметов с использованием материалов, обнаруженных в виртуальной среде.

Участники данной игровой версии властны воплощать в жизнь разнообразные архитектурные проекты, создавая полезные объекты, осваивать неизведанные территории и налаживать контакты между собой. Дистинктивной чертой данных игр является неограниченное пространство для проявления индивидуальности и инновационного мышления в рамках геймплея. Для иллюстрации подобных возможностей обратимся к примерам из числа мобильных представителей данного жанра.

### **1.5.1 Игра Mindustry**

Mindustry — это игра, предоставляемая бесплатно и принадлежащая к категории крафтинга, созданная разработчиком под псевдонимом Anuken (рисунок 1).

Mindustry представляет собой стратегическую игру, включающую в себя элементы построения промышленных комплексов, защиты объектов с использованием оборонительных башен и механики стратегии, основанной на действиях в реальном времени. В этой игре игроку предстоит разрабатывать эффективные логистические сети для доставки боеприпасов к оборонительным

сооружениям, а также осуществлять производство различных строительных материалов и машин. Управление боевыми подразделениями для захвата вражеских баз позволяет расширять свои промышленные мощности и укреплять позиции на карте.

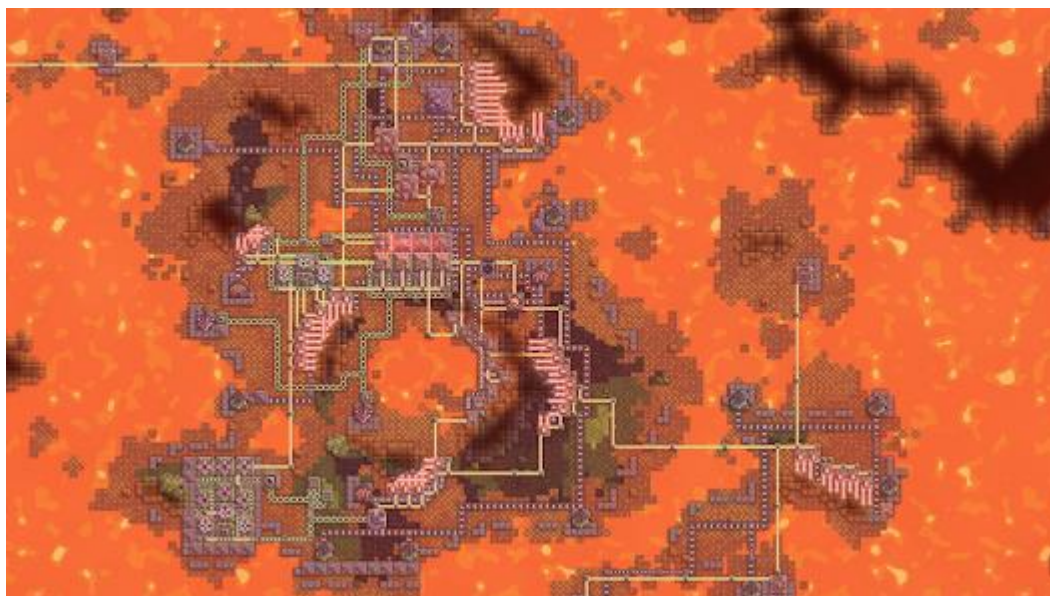


Рисунок 1 – Сцена из видеоигры Mindustry.

Преимущества и недостатки данной игры детально изложены в Таблице 2.

Таблица 2 – Сопоставительный обзор с конкурентом Mindustry

Достоинства	Недостатки
Простая и понятная графика	Некоторые сложные механизмы управления в игре остаются без пояснения
Полная свобода действий	Прогресс подчиняется специфическим условиям каждого уровня

## 1.5.2 Игра Крафт-кликер

Крафт-кликер — это видеоигра, основанная на создании предметов, разработанная студией Bling Bling Games GmbH, представленная на рисунке 2.



Рисунок 2 – Сцена игры Крафт-кликер

В "Крафт-кликере" задача игрока заключается в изготовлении изделий, используя начальные материалы, реализации полученной продукции для приобретения апгрейдов и постепенном расширении объемов производственных операций.

Процесс крафта осуществляется в автоматическом режиме, однако игроки могут ускорять процесс создания товаров через активные нажатия.

Плюсы и минусы игрового процесса представлены в таблице 3.

Таблица 3 – Сопоставительный обзор для конкурента Крафт-кликер.

Достоинства	Недостатки
Интуитивно понятный интерфейс	Невозможно самостоятельно организовать процесс производства.
Простая система развития	Затянутое развитие, основанное на ожидании

### 1.6 Заключительный анализ результатов

После тщательного анализа предметной области, рассмотрения достоинств и ограничений разнообразных игровых движков, определения целей создания игры, и применения знаний, полученных из опыта игры в похожих проектах, можно сформулировать ключевые аспекты грядущего продукта. Вот основные выводы:

- Данный проект относится к жанру создания и использования ресурсов, а его конкретное направление - управление ресурсами и процессами.
- Базовая двумерная графика в топ-даун перспективе.
- Основные механики: генерация первичных материалов и их интеграция для производства улучшенных объектов.
- Идеальный выбор инструмента разработки является GameMaker.
- Разработка интуитивно понятного UI, внедрение учебных модулей для новичков, гарантирование широких возможностей в пределах правил игры, избегание фокусировки на времени ожидания, чтобы прогресс определялся навыками и творческим потенциалом пользователя, а не количеством часов, проведенных в игровом процессе.



## **2 ПРАКТИЧЕСКАЯ ЧАСТЬ**

### **2.1 Моделирование предметной области**

В процессе создания игры, ориентированной на крафтинг, крайне важно детально изучить и воссоздать доменную область для достижения реалистичности и захватывающего геймплея. Этот этап охватывает выявление ключевых элементов, таких как материалы, инструментарий и рецепты создания предметов, проработку их взаимоотношений и функционирования внутри игровой вселенной. Разработка должна включать в себя формирование как логической, так и физической структуры игрового пространства, поддерживающей основные игровые механики: добычу материалов, изготовление объектов и взаимодействие с игровым миром.

#### **2.1.1 Создание мира**

Разработка игровой вселенной в жанре крафтинга представляет собой центральный момент в создании приложения, поскольку она задает фундамент для игровой механики в целом. В этом процессе участвуют различные ключевые элементы:

##### **1. Создание базовой карты:**

– Базовый ландшафтный дизайн: задействование статичных, предварительно разработанных или упрощённых процедурных ландшафтов для воплощения первоначального плана объекта. Это часто ограничивается ровной местностью с ограниченным количеством элементов - например, трассами и строительными площадками.

– Территориальное планирование: выделение участков под разнообразные производственные цеха, складские помещения и транспортные магистрали, способствующее оптимальному распределению пространства на промышленных объектах.

##### **2. Распределение производственных площадей:**

- Ключевые сооружения и элементы: разработка базовых моделей или спрайтов для индустриальных объектов, включая производственные помещения, хранилища и административные здания. Эти элементы предназначены для их размещения в специально выделенных локациях на карте.

- Месторасположение базовых элементов, символизирующих ресурсы и материалы, включая контейнеры с исходным материалом и финальным товарами.

### 3. Разработка алгоритма работы производственной линии:

- Управление потоком материалов: создание основных принципов передачи материалов между производственными единицами. Это достигается через написание базовых скриптов или использование графических методов, например, блок-диаграмм.

- Производственные процессы включают в себя серию операций по переработке сырья в конечную продукцию в установленный период времени.

### 4. Интерактивность мира:

- Ключевые действия пользователя заключаются в осуществлении основных взаимодействий, включающих расстановку сооружений, распределение ресурсов и активацию производственных цепочек. Для этих процессов могут быть разработаны интуитивно понятные интерфейсы и элементы управления.

- Применение основ физики в анимации улучшает понимание производственных операций через базовые визуализации, такие как работа транспортёрной системы или процесс погрузки материалов в оборудование.

### 5. Интерфейс пользователя:

- Интуитивный интерфейс: создание легко усваиваемого интерфейса, облегчающего управление производственным комплексом. Включает в себя элементы управления, такие как кнопки для эрекции сооружений, информационные панели для визуализации ресурсов и базовые схемы производственных цепей.

– Интеграция обучающих элементов: активация руководств и советов, предназначенных для ознакомления с фундаментальными принципами игры, позволяет новичкам освоить ключевые аспекты и начать управление производственным процессом на заводе.

### **2.1.2 Разработка игровых объектов**

В симуляторе производственного комплекса, элементы такие как экскаваторы, металлургические печи и конвейерные системы считаются неотъемлемыми компонентами. Они являются ключевым составляющим для реализации производственных цепочек и интерактивности, лежащих в фундаменте игрового процесса. В данном разделе представлен анализ начальных этапов создания таких элементов:

#### **1. Буры:**

– Применение: Буровые установки задействованы в экстракции природных ресурсов. Они обязаны иметь возможность соединения с узлами ресурсов и осуществлять автоматизированную экстракцию установленного объема ресурсов в определенный временной промежуток.

– Визуализация: дизайн базовых объектов или спрайтов для буровых устройств. Эти элементы могут напоминать сверла или автомобили с буровыми агрегатами. Критично обеспечить очевидность их дизайна для пользователя.

#### **Логика работы:**

– Позиционирование: игроку предоставляется возможность устанавливать дрель на узле с ресурсами. Это приводит к началу процесса экстракции данных материалов.

– Метод экстракции: используя механизм автобурения, ресурсы автоматически извлекаются и транспортируются в резервуар или на специфицированный участок вблизи. Это достигается через применение базовых алгоритмов, которые регулируют декремент объема ресурсов в месторождении и их инкремент в емкости устройства.

## 2. Печи:

– Назначение: Плавильные печи предназначены для проведения процесса плавления руды, с целью получения из неё готовых металлических изделий. Эти устройства задействуются для преобразования руды в слитки или другие виды металлургической продукции.

– Визуализация: разработка базовых конструкций или иконок, изображающих печи. Эти элементы могут напоминать либо печи, либо плавильные агрегаты.

### Логика работы:

– Подача руды: Игрокам предоставляется функциональность для загрузки руды в печь, используя либо построенный интерфейс, либо метод перетаскивания.

– Процесс переплавки включает автоматическое запускание печи для переплавки, длительность которой строго регламентирована. По завершении, печь выдает переработанные материалы, доступные к сбору пользователем или их автоматической транспортировке в хранилище.

## 3. Сборочные автоматы:

– Применение: Автоматизированные сборочные машины служат для комплексной обработки простых исходных материалов с целью изготовления комплексных объектов. Эти аппараты обладают возможностью использования разнообразных ресурсов для выпуска завершённой продукции.

– Визуализация: разработка базовых моделей или спрайтов для монтажных станций. Элементы могут представлять собой машинное оборудование или транспортерные системы с роботизированными механизмами.

### Логика работы:

– Импорт ресурсов: пользователь игры должен обладать функциональностью для ввода ресурсов в конструкторское устройство. Это достигается либо через пользовательский интерфейс, либо путем

автоматизированной трансфертации сырья из дробителей и плавильных агрегатов.

– Процесс монтажа: автоматизированная сборочная линия самостоятельно активирует монтаж, потребляя предварительно заготовленные ресурсы. После завершения этапа производства, готовая продукция доступна для ручную сбора пользователя или её автоматической транспортировки на складское хранение.

#### 4. Интерактивность и управление:

– Интерактивный интерфейс: объекты обязаны быть оснащены легко усваиваемыми и очевидными управляющими механизмами. К ним относятся элементы управления, такие как кнопки для добавления данных, визуальные указатели активности и информационные панели, показывающие объем сгенерированных материалов.

– Автоматизация процессов: критически необходимо интегрировать автоматизационные решения. К примеру, дрели автоматически транспортируют извлеченные ресурсы в печи для переработки, а затем печи направляют обработанные материалы к сборочным линиям. Это достижимо через разработку базовых программ и активацию событий.

### **2.1.3 Игровые механики**

Игровые механики обеспечивают основу для взаимодействия между игроком и игровым миром. В элементарном симуляторе завода основные механики охватывают экстракцию ресурсов, трансформацию сырья, сборку продуктов и автоматизацию производственных операций.

В тексте изложены ключевые игровые элементы, способствующие разработке захватывающего и поддающегося управлению пользователем игрового опыта:

#### 1. Добыча ресурсов:

– Размещение буровых установок: участник может развертывать добывающие машины на залежах полезных ископаемых. Эти устройства автоматически приступают к экстракции материалов, включая оре или камни. Критично обеспечить игрока обратной связью через графические и текстовые индикаторы, отображающие процесс добычи и объем накопленных ресурсов.

– Ресурсные узлы - это специализированные точки на карте, предназначенные для размещения источников материалов. В каждом таком узле находится фиксированный объем ресурсов, доступных для экстракции с помощью буровых установок.

– Обратная связь: буровые установки должны сигнализировать о своём рабочем статусе, например, через индикаторы "в процессе", "добыча окончена", "емкость заполнена".

## 2. Переработка материалов:

– Применение печей в игровом процессе предусматривает загрузку добытой руды для ее дальнейшей переплавки. Эти устройства функционируют автоматически, трансформируя руду в финальные материалы в течение установленного временного промежутка.

– Процесс переработки металлов включает в себя визуальные маркеры, например, модификацию цвета печи или визуализацию через анимацию, сигнализирующие о его ходе. По окончании этой стадии, переработанные материалы становятся доступны для последующего применения.

– Сбор готовой продукции: разработка удобного пользовательского интерфейса для извлечения обработанных материалов из печи.

## 3. Сборка изделий:

– Сборочные машины: участник имеет возможность установки и пополнения сборочных машин ресурсами. Эти устройства автономно запускают производственный процесс, обрабатывая загруженные ресурсы.

– Руководства по сборке: основы и применение базовых инструкций, детализирующих требуемые компоненты для производства конкретных предметов. К примеру, "1 блок железа + 2 блока дерева = инструмент из железа".

– Процесс производства: интеграция визуальных и текстовых сигналов, отображающих этапы сборки и состояние аппарата. Важно, чтобы пользователь наблюдал за используемыми ресурсами и оставшимся временем до конца сборки.

#### 4. Автоматизация процессов:

– Конвейерные системы обеспечивают разработку элементарных транспортёров и других механизмов автоматизации для транспортировки материалов между буровыми установками, плавильными печами и сборочными машинами. Это дает возможность автоматизировать рабочие процессы, минимизируя трудоемкие ручные операции.

– Автоматизационная логика: применение элементарных программ или алгоритмов для организации потока материалов через производственную линию. К примеру, добыча самостоятельно транспортируется из экскаватора в металлургическую печь, а обработанные изделия - в агрегат для сборки.

#### 5. Интерфейс и управление:

– Интерфейсы управления: создание интуитивно понятных и доступных интерфейсов для управления различными объектами. Эти интерфейсы должны отображать актуальные данные об объекте, наличие ресурсов и ход выполнения производственных задач.

– Обучающая поддержка: интеграция системы направлений и инструкций, способствующих адаптации игрока к фундаментальным игровым процессам. Это может представлять собой базовое руководство или динамические подсказки, активирующиеся при начале взаимодействия с различными элементами игры.

#### 6. Прогресс и цели:

– Разработка целеполагания: конструирование набора целей и задач, способствующих направлению и стимулированию пользователя к улучшению

заводского производства. Это может охватывать, к примеру, экстракцию установленного объема материалов, изготовление номинированного количества продукции или автоматизацию специфических операций.

– Развитие и успехи: Имплементация механизма трекинга успехов игроков, награждающего их за достижения определенных целей. Это может оформляться как список достижений или в форме графических призов.

#### **2.1.4 Технологическая реализация**

Разработка игрового приложения охватывает ряд ключевых этапов:

1. Определение платформы и инструментария для разработки:

– Игровая платформа: Выбор подходящей системы для разработки игр - это критический момент. Новичкам в сфере создания игр советуют использовать легко осваиваемые и широко известные средства, например, Gamemaker Studio.

– Для разработки базовых спрайтов и начальных моделей доступны бесплатные программные решения, в частности GIMP, предназначенный для работы с 2D-изображениями.

2. Разработка игровых объектов:

– Буровые установки: разработка программного обеспечения для контроля размещения буров, инициации процесса добычи и администрирования хранения извлеченных минералов. Для обеспечения базовой функциональности могут применяться такие инструменты, как задержки выполнения и итеративные процессы.

– Печные установки: создание скрипта для управления печами, автоматизирующего ввод руды, её плавление и выдачу продукции. Этап плавления реализуется через корутины или таймеры.

– Автоматизация производства: разработка программного обеспечения для автоматизированных линий, управляющего загрузкой компонентов, выполнением сборки и выдачей продукции. Алгоритмы функционирования могут базироваться на применении инструкций и итерациях.



### 3. Интерфейс пользователя:

– Интерфейсы управления: создание базовых пользовательских интерфейсов (UI) через применение стандартных средств. Интерфейсы предназначены для визуализации данных о статусе объектов и интеграции элементов управления.

– Интеграция обучающего механизма: разработка интерактивной системы, предоставляющей контекстную помощь через информационные всплывающие окна при инициировании взаимодействия с различными игровыми элементами. Это достигается через интеграцию модальных окон и сообщений.

### 4. Автоматизация и логика:

– Проектирование базовых конвейерных систем: создание конвейерных лент с применением анимаций и программных кодов для задачи автоматизации передачи материалов меж объектами. Эти ленты выполняются в виде движущихся моделей, транспортирующих грузы по заранее заданному маршруту.

– Автоматизированное управление процессом: разработка алгоритмов для авто-трансфера материалов из дрелей в плавильни, после чего в установки для монтажа. Это достигается через применение событийных механизмов и активаторов.

## **2.2 Разработка и реализация прототипа**

### **2.2.1 Подготовка рабочей среды**

При выборе инструмента для создания имитационного моделирования промышленного предприятия предпочтение было отдано GameMaker Studio благодаря ее доступности и адекватным функциональным возможностям для новичков в сфере программирования игр. Первым шагом стала установка актуальной версии программного обеспечения, загруженной с официального портала разработчика. Процесс разработки начался с инициализации нового проекта, ориентированного на использование языка программирования

GameMaker Language (GML), с последующим присвоением проекту наименования.

Инициатива была реализована путем структурирования проекта через стандартизованную систему каталогов: "Sprites", предназначенную для архивации визуальных элементов, "Objects" для размещения объектов, участвующих в игровом процессе, "Rooms" служащую для компоновки уровней игры, и "Scripts", предусмотренную для хранения кода. В процессе работы над проектом были разработаны элементарные спрайты, примером которых является "spr\_Drill" для изображения буровых установок, используя либо интегрированные возможности редактирования, либо импортируя уже готовые варианты.

Затем осуществлена разработка ключевых элементов игры. К примеру, создан игровой элемент под названием "obj\_Drill", к которому был ассоциирован визуальный элемент - спрайт "spr\_Drill". Для каждого элемента определены уникальные события и описана их функциональная логика с использованием языка программирования Game Maker Language (GML). Для компоновки игровых уровней был разработан основной игровой пространство - комната "rm\_Main", где были размещены разработанные элементы, включая "obj\_Drill".

Программные скрипты были разработаны для эмуляции поведения и функционирования объектов в среде. Эти программные модули ассоциируются с определенными событиями объектов, гарантируя корректное взаимодействие и выполнение предназначенных действий. Чтобы убедиться в стабильности и корректности реализованных механик, проект подвергался регулярным процедурам тестирования и отладки через специализированный инструмент внутрипрограммного запуска.

### **2.2.2 Создание игровой среды**

Разрабатывая игровую платформу для симулятора промышленного комплекса, важно было сконструировать ключевые компоненты и алгоритмы,

поддерживающие эффективное взаимодействие пользователя с игровым процессом. Это пространство должно быть интуитивно управляемым и высоко функциональным, чтобы даже начинающие разработчики смогли без труда воплотить в жизнь первостепенные игровые механизмы.

Инициацией проекта стало формирование основной комнаты, выступающей в роли игрового уровня. В среде GameMaker Studio была воплощена комната "rm\_Main", которая служит местом для расположения ключевых игровых элементов. Дизайн уровня был тщательно спроектирован, чтобы обеспечить стратегически оправданное размещение игровых объектов.

В дальнейшем была разработана графика в виде спрайтов и визуальных образов для различных категорий технического оснащения: экскаваторов, горнов и конвейерных систем. Дизайн этих спрайтов отличался простотой и практичностью, целью которого было обеспечить легкость в идентификации и визуальном различии между ними. В качестве примера, экскаватор был изображен в форме крутящегося бура, горн — как куб с пламенем внутри, а конвейерная система — как лента для транспортировки с роботизированными руками.

Значительное внимание было сосредоточено на развитии механизма управления ресурсами и их извлечении. Автоматизированные буровые установки занимаются экстракцией полезных ископаемых с назначенных на игровой карте месторождений. Полученные материалы впоследствии подвергаются обработке в металлургических печах и применяются на производственных линиях сборочных машин для производства конечных товаров. Процедура извлечения ресурсов, их обработки и сборки продукции была внедрена через написание кода на языке программирования Game Maker Language (GML).

Для обеспечения управления игроком над объектами созданы контрольные интерфейсы. На панели каждого объекта представлена информация о его ключевых характеристиках и функциональном состоянии. Пользователь может

отслеживать доступное количество материалов, состояние переработки или производства, а также прочую значимую информацию.

Была разработана автоматизационная система процессов. Пользователь имел возможность сооружать транспортные линии для беззаботной передачи материалов между структурами, что облегчало контроль и способствовало фокусировке на экспансии производства.

### **2.2.3 Реализация игровых механик**

Внедрение игровых процессов в симуляторе производства охватывало конструирование и включение ключевых операций, позволяющих пользователю взаимодействовать с игровым миром. Данные процессы позволяли осуществлять экстракцию ресурсов, их последующую трансформацию и производство финальных изделий.

#### **1. Добыча ресурсов.**

Для осуществления процесса извлечения ресурсов были разработаны специализированные устройства - буровые установки. Каждая из них, идентифицированная в системе как "obj\_Drill", оборудована функцией автоматического изъятия ресурсов с нод, которые местоположены на игровой карте. Эти устройства регулярно проводят аудит своего состояния и объема экстрагированных материалов, визуализируя полученные данные на интерфейсе пользователя.

Для функционала игрового элемента был написан соответствующий программный код. В качестве начального шага определено событие Create (рисунок 3), активирующееся при инициализации игры и инстанцировании элемента. Этот процесс устанавливает и подготавливает к работе переменные, задействованные в этом элементе.

Событие Right Down включает в себя базовый скрипт, который позволяет изменять состояние элемента при активации правой кнопкой мыши (рисунок 4).

```
obj_drill: Create
Create
1 // Устанавливаем начальные значения переменных
2 is_active = false; // Состояние бура (включен или выключен)
3 resource_amount = 0; // Количество добытых ресурсов
4 drill_speed = 1; // Скорость добычи (ресурсов в секунду)
5 drill_timer = 0; // Таймер для отсчета времени до следующей добычи
6 drill_interval = 60; // Интервал времени для добычи (в шагах, 60 шагов = 1 секунда)
7 is_dragging = false; // Состояние перетаскивания ресурсов
8 dragged_resource_amount = 0; // Количество ресурсов, перетаскиваемых мышью
9 dragged_resource_x = 0; // Координаты перетаскиваемого ресурса
10 dragged_resource_y = 0; // Координаты перетаскиваемого ресурса
11
```

Рисунок 3 – Код события Create объекта obj\_Drill

```
obj_drill: Events
Create Step
1 // Переключаем состояние бура
2 is_active = !is_active;
3
```

Рисунок 4 – Код события Right Down объекта obj\_Drill

Затем был создан объект Step, отображаемый на рисунке 5. Описанный его код предоставляет объекту следующие функциональные возможности:

- извлечение железорудных ресурсов из залежей с периодичностью.
- Транспортировка извлеченных материалов из бурения в альтернативные контейнеры через использование левого клика.

Так, создан полноценный игровой элемент, достигающий всех его задач, в дополнении, создан удобный пользовательский интерфейс для предоставления данных об элементе игроку (рисунок 6).

```
obj_drill: Events
Create x Step x
1 // Проверка включен ли бур
2 if (is_active) {
3     // Увеличиваем таймер на каждый шаг
4     drill_timer += 1;
5
6     // Проверяем, прошло ли достаточное количество времени для добычи ресурса
7     if (drill_timer >= drill_interval) {
8         // Сбрасываем таймер
9         drill_timer = 0;
10
11     // Проверяем, касается ли бур месторождения (предполагаем, что месторождение имеет объем)
12     if (place_meeting(x, y, obj_OreDeposit)) {
13         // Увеличиваем количество добытых ресурсов
14         resource_amount += drill_speed;
15
16         // Отображаем добытые ресурсы в консоли (можно заменить на отображение на экране)
17         show_debug_message("Iron ore mined: " + string(resource_amount));
18     }
19 }
20 }
21
22 // Перетаскивание ресурсов
23 if (is_dragging) {
24     // Ресурс следует за мышью
25     dragged_resource_x = mouse_x;
26     dragged_resource_y = mouse_y;
27 }
28
29 // Отпускание ресурсов
30 if (mouse_check_button_released(mb_left)) {
31     if (is_dragging) {
32         is_dragging = false;
33
34         // Проверка на нахождение над печью
35         if (place_meeting(mouse_x, mouse_y, obj_Furnace)) {
36             var target_furnace = instance_place(mouse_x, mouse_y, obj_Furnace);
37             if (target_furnace != noone) {
38                 // Перемещаем ресурсы в печь
39                 target_furnace.iron_ore += dragged_resource_amount;
40                 dragged_resource_amount = 0;
41             }
42         } else {
43             // Возвращаем ресурсы в бур
44             resource_amount += dragged_resource_amount;
45             dragged_resource_amount = 0;
46         }
47     }
48 }
```

Рисунок 5 – Код события Step объекта obj\_Drill



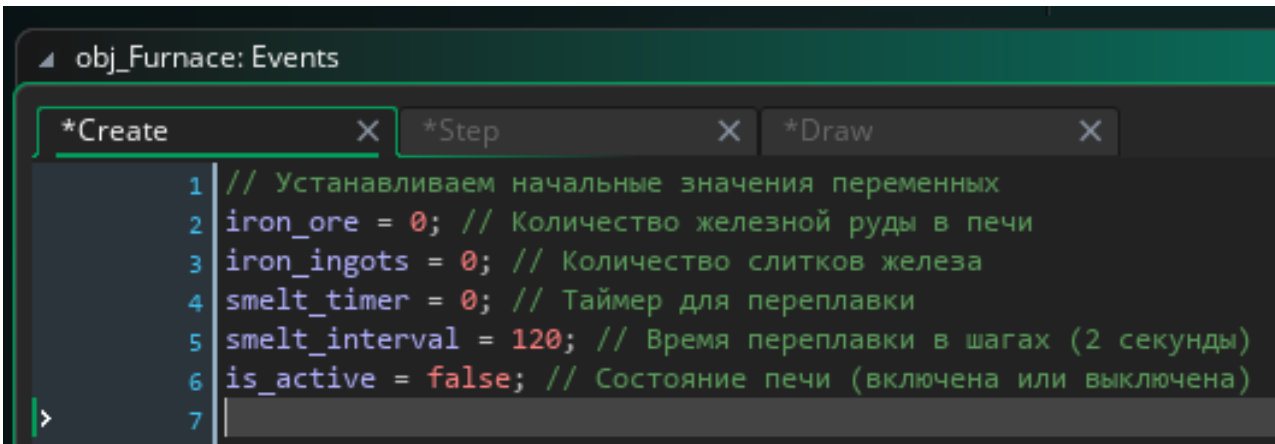
Рисунок 6 – Демонстрация работы объекта бура в игре

## 2. Переработка ресурсов.

Процесс трансформации извлеченных из недр минералов в использованные в продукции материалы выполнялся через технологическое оборудование, указанное в системе как "obj\_Furnace". Эти устройства загружались рудой и по заложенной программе автоматизированно осуществляли ее переплавку в конечные продукты.

Аналогично, для объекта obj\_Furnace была создана программа, необходимая для его функционирования. Событие Create выполняет аналогичные задачи, как и в случае с объектом бура, возможность инициализации переменных, применяемых в данном объекте. Программный код события Create для obj\_Furnace демонстрируется на рисунке 7.

Встроенный алгоритм события Step моментально активирует функционирование плавильной печи при обнаружении в её камере руды, и деактивирует после завершения её обработки (рисунок 8). Перемещение руды из добычного устройства в плавильный агрегат осуществляется с использованием drag-and-drop механизма через левый клик мыши.



```
obj_Furnace: Events
*Create
1 // Устанавливаем начальные значения переменных
2 iron_ore = 0; // Количество железной руды в печи
3 iron_ingots = 0; // Количество слитков железа
4 smelt_timer = 0; // Таймер для переплавки
5 smelt_interval = 120; // Время переплавки в шагах (2 секунды)
6 is_active = false; // Состояние печи (включена или выключена)
7
```

Рисунок 7 – Код события Create объекта obj\_Furnace

Процедура Draw визуализирует на дисплее данные о состоянии печи, включая текущий статус работы, объем загруженной железной руды и количество удачно отлитых металлических слитков (рисунок 9).

```
obj_Furnace: Events
*Create X *Step X *Draw X
1 // Проверяем, есть ли железная руда для переплавки
2 if (iron_ore > 0) {
3     is_active = true; // Включаем печь
4     smelt_timer += 1; // Увеличиваем таймер
5
6     // Проверяем, прошло ли достаточно времени для переплавки руды
7     if (smelt_timer >= smelt_interval) {
8         smelt_timer = 0; // Сбрасываем таймер
9         iron_ore -= 1; // Уменьшаем количество руды
10        iron_ingots += 1; // Увеличиваем количество слитков
11        show_debug_message("Iron ingot produced: " + string(iron_ingots)); // Отобр
12    }
13 } else {
14     is_active = false; // Выключаем печь, если руды нет
15 }
16
```

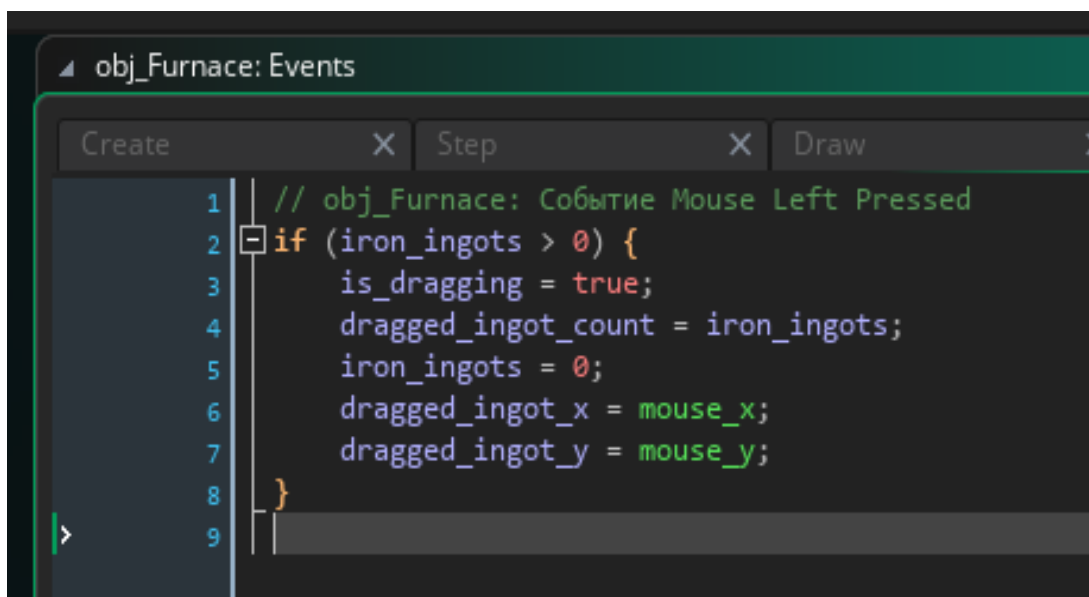
Рисунок 8 – Код события Step объекта obj\_Furnace

```
obj_Furnace: Events
*Create X *Step X *Draw X
1 // Рисуем спрайт объекта
2 draw_self();
3
4 // Отображаем количество руды и слитков над объектом
5 var ore_text = "Iron ore: " + string(iron_ore);
6 var ingot_text = "Iron ingots: " + string(iron_ingots);
7 draw_text(x, y - 16, ore_text);
8 draw_text(x, y - 32, ingot_text);
9
10 // Отображаем состояние печи (включена или выключена)
11 var status_text = is_active ? "Active" : "Inactive";
12 draw_text(x, y - 48, status_text);
13
```

Рисунок 9 – Код события Draw объекта obj\_Furnace



Реализован механизм драг-энд-дроп для перемещения обработанных материалов из печи напрямую в конструктор, используя левый клик мыши, подобно тому, как это работает для перемещения железной руды (рисунок 10).



```
obj_Furnace: Events
Create X Step X Draw X
1 // obj_Furnace: Событие Mouse Left Pressed
2 if (iron_ingots > 0) {
3     is_dragging = true;
4     dragged_ingot_count = iron_ingots;
5     iron_ingots = 0;
6     dragged_ingot_x = mouse_x;
7     dragged_ingot_y = mouse_y;
8 }
9
```

Рисунок 10 – Код события Left Pressed объекта obj\_Furnace

Таким образом, был создан элемент для виртуальной кузницы, который может получать железную руду и трансформировать ее в металлические слитки. Механизм транспортировки руды от дрели до кузницы разработан так, чтобы быть простым и интуитивно понятным для пользователя. Пользовательский интерфейс элемента предоставляет возможность отслеживания процесса обработки и отображает информацию о наличии материалов внутри (рисунок 11).

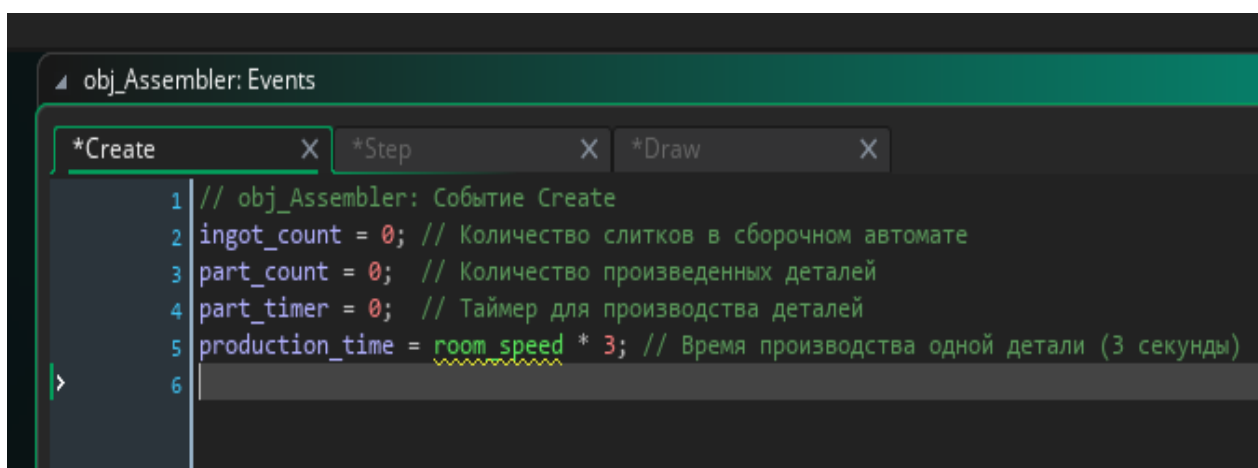


Рисунок 11 – Демонстрация работы объекта бура и печи в игре

### 3. Создание изделий.

В процессе производства изделий задействовали автоматические сборочные установки, идентифицированные как "obj\_Assembler". Эти устройства получали на вход обработанные материалы и, следуя установленным алгоритмам, формировали готовую продукцию.

Аналогично процедурам, реализованным для двух предшествующих устройств, в случае сборочного робота было разработано событие Create, которое служит для инициализации применяемых в процессе переменных (рисунок 12).



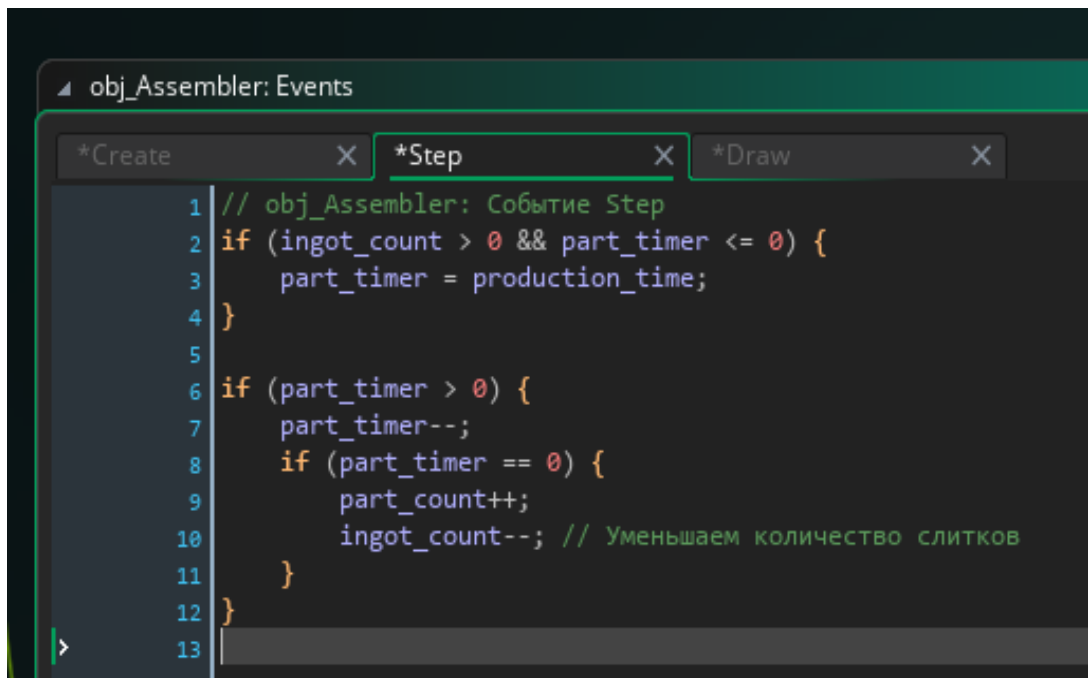
```
obj_Assembler: Events
*Create X *Step X *Draw X
1 // obj_Assembler: Событие Create
2 ingot_count = 0; // Количество слитков в сборочном автомате
3 part_count = 0; // Количество произведенных деталей
4 part_timer = 0; // Таймер для производства деталей
5 production_time = room_speed * 3; // Время производства одной детали (3 секунды)
6
```

Рисунок 12 – Код события Create объекта obj\_Assembler

Этап Step предназначен для трансформации железных слитков в компоненты, как показано на рисунке 13.

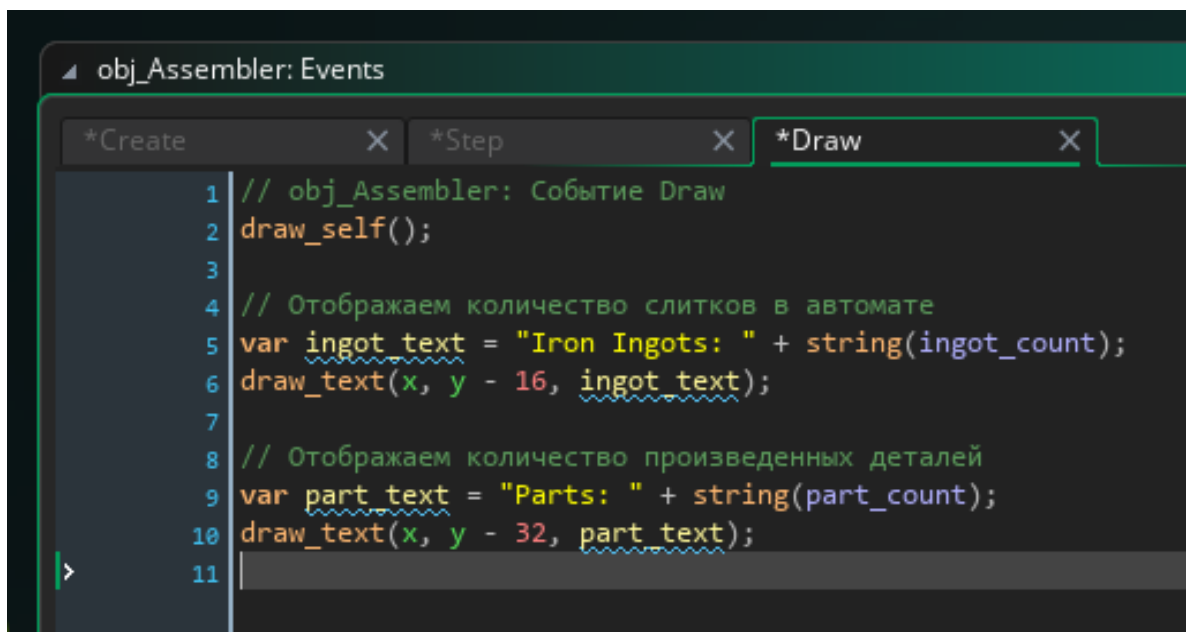
В событии Draw (рисунок 14) осуществляется визуализация статуса объекта, индикация уровня его ресурсов и отображение количества изготовленных компонентов.

Разработанный элемент успешно прошел проверку функциональности непосредственно в игровом процессе (см. рисунок 15).



```
obj_Assembler: Events
*Create
*Step
*Draw
1 // obj_Assembler: Событие Step
2 if (ingot_count > 0 && part_timer <= 0) {
3     part_timer = production_time;
4 }
5
6 if (part_timer > 0) {
7     part_timer--;
8     if (part_timer == 0) {
9         part_count++;
10        ingot_count--; // Уменьшаем количество слитков
11    }
12 }
13
```

Рисунок 13 – Код события Step объекта obj\_Assembler



```
obj_Assembler: Events
*Create
*Step
*Draw
1 // obj_Assembler: Событие Draw
2 draw_self();
3
4 // Отображаем количество слитков в автомате
5 var ingot_text = "Iron Ingots: " + string(ingot_count);
6 draw_text(x, y - 16, ingot_text);
7
8 // Отображаем количество произведенных деталей
9 var part_text = "Parts: " + string(part_count);
10 draw_text(x, y - 32, part_text);
11
```

Рисунок 14 – Код события Draw объекта obj\_Assembler

Процесс трансфера материалов из печи в устройство для сборки, конвертация слитков в компоненты, а также корректное отображение текущего состояния элемента эффективно действуют.



Рисунок 15 – Демонстрация работы объектов в игре

#### 2.2.4 Разработка интерфейса пользователя

В сфере геймдизайна, элементы пользовательского интерфейса (UI) являются фундаментальными для обеспечения эргономичного и интуитивно понятного взаимодействия пользователя с игровым процессом. В процессе разработки нашего фабричного симулятора в жанре крафтинг, особое внимание было уделено реализации такого интерфейса, который обеспечивал бы лёгкость освоения и доступность даже для тех игроков, кто ранее не сталкивался с подобными игровыми механиками.

##### Анализ требований к интерфейсу

Перед тем как приступить к разработке пользовательского интерфейса, мы провели исследование требований, включающее:

- 1) Анализ целевой аудитории видеоигры и понимание их предпочтений.
- 2) Анализ пользовательских интерфейсов известных крафтинг-игр.
- 3) Сбор и анализ отзывов от будущих пользователей.

##### Основные компоненты интерфейса

Исходя из проведённого исследования, были идентифицированы следующие основные элементы пользовательского интерфейса:

##### 1. Главное меню:

а) Содержит кнопки для запуска новой партии, восстановления ранее сохранённого процесса, конфигурации системных параметров и завершения сеанса игры.

б) Создано следуя концепциям минимализма, избегая избыточного информационного наполнения для пользователя.

## 2. Игровой экран:

а) Главный интерфейс, предназначенный для взаимодействия пользователя с виртуальной средой.

б) Содержит компоненты пользовательского интерфейса, включая отображение ресурсов, панель инструментов и панель опций.

## 3. Панель ресурсов:

а) Отображает текущий объем добытых и переработанных ресурсов.

б) Имплементирована через символические изображения с цифровыми показателями, соответствующими различным категориям ресурсов.

## 4. Панель инструментов:

а) Включает инструменты и объекты, доступные игроку для взаимодействия.

б) Включает элементы управления для установки дробилок, плавильных комплексов и конвейерных систем.

## 5. Меню действий:

а) Контекстное меню, активируемое при пользовательских действиях с элементами игры, такими как активация дрели или использование печи.

б) Обеспечивает контроль над элементами, активацию/деактивацию, транспортировку материалов и прочее.

Процесс разработки пользовательского интерфейса включал следующие этапы:

### 1. Прототипирование:

а) В начальной фазе разработки проекта выполнили рисование эскизов и создание прототипов пользовательского интерфейса, применив для этого инструментарий Figma.

б) Макеты способствовали наглядному представлению архитектуры пользовательского интерфейса и осуществлению предварительной проверки на эргономичность.

### 2. Разработка визуальных компонентов:

а) Разработали компоненты пользовательского интерфейса, включая кнопки, значки и панели инструментов.

б) Графическое оформление было выполнено с помощью программ Adobe Photoshop и Illustrator.

в) Акцентировалось на соблюдении единства стилевых и цветовых решений, с целью придания интерфейсу гармоничного и профессионального облика.

### 3. Интеграция интерфейса в игру:

а) Созданные визуальные компоненты успешно встроены в платформу для создания игр GameMaker Studio.

б) С применением внутренних функций GameMaker Studio разработали пользовательский интерфейс и интегрировали его в различные уровни игры.

### 4. Тестирование интерфейса:

а) Было выполнено юзабилити-тестирование и проверка на ошибки программного обеспечения.

б) Были созданы тестовые сценарии, включающие различные взаимодействия с элементами интерфейса.

в) Полученные отзывы были применены для корректировок и совершенствования.

#### Оптимизация интерфейса

По окончании первичного этапа разработки и проведения тестов, осуществили меры по совершенствованию пользовательского интерфейса:

#### 1. Улучшение эффективности работы:

а) Было выполнено анализирование профилей интерфейса для обнаружения и решения проблем с производительностью, связанных с узкими местами.

б) Сокращены объемы визуальных компонентов, сохраняя их высокое разрешение, что привело к уменьшению затрат ресурсов компьютера.

#### 2. Оптимизация кода:

а) Выполнен аудит исходного кода, улучшены методы обновления и визуализации пользовательского интерфейса.

б) Оптимизированы вызовы и сокращено число операций для улучшения производительности.

#### Пути эволюции пользовательского интерфейса

В будущей стратегии по усовершенствованию пользовательского интерфейса игры запланированы следующие улучшения:

##### 1. Добавление новых функций:

а) Запланировано улучшение пользовательского интерфейса, включая интеграцию дополнительных инструментов управления и панелей с данными.

б) Разработка и имплементация интерактивного туториала для обучения начинающих участников.

##### 2. Кроссплатформенная оптимизация:

а) Адаптация дизайна интерфейса под множество размеров экрана и типов разрешений гарантирует удобное взаимодействие с игровым контентом на смартфонах и планшетах.

б) Разработка гибкого веб-дизайна, оптимизированного для различных разрешений экрана.

##### 3. Эстетическое совершенствование:

а) Предстоит продолжение разработки в области визуализации и динамики для улучшения эстетики пользовательского интерфейса.

б) Реализация функциональности по настройке тем и скинов для индивидуализации пользовательского интерфейса.

Разработка пользовательского интерфейса включает в себя сложную серию шагов, начиная от инициативной стадии планирования до глубокого дизайна и аккуратного исполнения на каждой фазе. В результате этих стараний мы получили интерфейс, который позволяет игрокам интуитивно и удобно взаимодействовать с игровым миром. Это, в свою очередь, улучшает игровое впечатление и повышает уровень удовлетворенности среди пользователей.

### **2.2.5 Оптимизация игры**

Усовершенствование фабричного симулятора подразумевало действия для увеличения скорости работы и оптимизации интерфейса для пользователя:

#### **1) Оптимизация графики.**

Оптимизация графических элементов проведена для снижения системных требований. Применялись спрайты в минимально достаточном разрешении и с ограниченным числом анимационных кадров.

Каждый визуальный компонент был эффективно уплотнён через методы сжатия без вреда для качества, в результате чего уменьшился размер используемой памяти.

К тому же использовались методы ленивой загрузки, что позволило подгружать изображения лишь по мере необходимости.

#### **2) Сокращение числа элементов.**

Для повышения эффективности работы системы было уменьшено число действующих элементов в пространстве.

Введены функции деактивации или перевода в режим ожидания элементов, выходящих за пределы области наблюдения камеры, понижая тем самым нагрузку на CPU. Это критично для компонентов типа буров и транспортёров, обычно эксплуатируемых в значительных объёмах.

#### **3) Оптимизация кода.**

Исходный код видеоигры прошел детальный анализ и последующую оптимизацию, что способствовало улучшению его производительности.

Фокусировка производилась на важнейших задачах: экстракции материалов, их трансформации и монтаже продукции. Проведена оптимизация за счет удаления лишних операций и вызовов процедур. Для повышения скорости обращения данные, требующиеся регулярно, помещались в кэш.

#### **4) Снижение частоты обновления.**



Оптимизация темпа обновления отдельных компонентов игрового мира была осуществлена без потери качества геймплея. Так, мониторинг статуса элементов инвентаря и рефреш графического пользовательского интерфейса теперь выполняются не при каждом отображении кадра, а через фиксированные промежутки времени. Этот подход эффективно сократил ресурсоемкость программы.

#### 5) Оптимизация пути к ресурсам.

Для повышения эффективности процесса, оптимизация алгоритма транспортировки ресурсов по транспортным линиям была осуществлена. Реализация более продуктивного доступа к ресурсам и совершенствование системы их аллокации способствовала уменьшению количества столкновений и дефектов, ассоциированных с процедурами загрузки и разгрузки материалов.

#### 6) Тестирование и профилирование.

Игра прошла этапы оптимизации профиля для выявления и исправления буттлнеков в производительности.

Используя встроенные функции GameMaker Studio и дополнительные профилировщики, проанализировали разные элементы игрового процесса, что обеспечило целенаправленное усовершенствование недостатков.

#### 7) Кроссплатформенная оптимизация.

В процессе оптимизации игры основное внимание было направлено на её адаптацию для разнообразных платформ и экранных форматов. Реализация функций динамического масштабирования GUI и гибкой компоновки интерфейсных элементов гарантировала удобство использования как на устройствах с широкоформатными дисплеями, так и на портативных гаджетах.

Описанные ранее действия гарантировали стабильное функционирование игры даже на устройствах с низкой производительностью.

## 2.2.6 Тестирование

Качественная проверка является критически важной стадией в процессе создания ПО, особенно в индустрии компьютерных игр, где ключевыми факторами являются надежность, эффективность системы и уровень довольства конечных пользователей. Специально для нашего проекта - симулятора работы завода в жанре крафтинга, был разработан всесторонний план проверки, охватывающий ряд уровней: проверка функционала, аудит интерфейса пользователя, оценка производительности, выявление дефектов и ошибок, а также пробный запуск в реальных условиях с привлечением живой аудитории. В данной секции приведено детальное описание процедур тестирования, применяемых техник и инструментов, а также полученных итогов этих действий.

Тестовый план содержал следующие ключевые шаги:

1. Подготовка тестовой среды.
2. Тестирование функционала игровых элементов и механизмов.
3. Испытание интерфейса пользователя.
4. Производительное тестирование.
5. Проверка на наличие дефектов и неполадок.
6. Процесс бета-тестирования и анализ отзывов.

Подготовка тестовой среды

Для того чтобы гарантировать высокое качество и тщательность тестирования, наша команда создала тестовую инфраструктуру, состоящую из множества рабочих станций, оборудованных разнообразными конфигурациями техники и операционными системами. Это обеспечило возможность адекватно оценить производительность и надежность игрового продукта на различных платформах, а также идентифицировать потенциальные проблемы совместимости.

Тестовая среда включала:

- а) Рабочие места на основе ОС Windows, macOS и Linux.

б) Разнообразие комплектаций техники, охватывая экономичные и высокоэффективные решения.

в) Устройства с разнообразными диагоналями и разрешениями дисплеев для тестирования адаптации пользовательского интерфейса.

Функциональное тестирование целью имело подтверждение адекватной функциональности всех элементов интерфейса и игровых процессов. Ключевые компоненты, включая дрилли, горны и ленточные конвейеры, подверглись анализу на соответствие установленным спецификациям.

Кейс интеграционного анализа дрели:

а) Тестирование функционирования дрели при её запуске.

б) Исследование эксплуатации железной руды при взаимодействии с залежью.

в) Мониторинг аккумуляции извлеченной руды во время бурения.

г) Остановка работы буровой установки и прекращение эксплуатационной деятельности.

д) Иллюстрация теста функциональности микроволновки:

е) Контроль за загрузкой руды из бурного устройства в период перемещения.

ж) Тестирование включения доменной печи при обнаружении железной руды.

з) Мониторинг процедуры переработки минеральной руды в металлические слитки.

и) Контроль запасов и использования материалов в доменной печи.

Анализ пользовательского интерфейса (UI) сфокусирован на изучении его удобства использования и интуитивной понятности. В рамках данной процедуры испытывались основные аспекты:

а) Адекватное функционирование интерфейсных элементов, включая кнопки и пункты меню.

- б) Повышенная удобность управления игровыми элементами посредством интерфейса.
- в) Визуализация и рефреш данных по статусу элементов (запасы, состояние сверления и работы плавильной установки).
- г) Испытание функционала переноса руды из дрели в плавильный агрегат с использованием курсора.

#### Производительное тестирование

Производительное аудиторирование задач имело выявление уровня производительности игры на множестве устройств и операционных систем. В ходе оценки осуществлялись ключевые испытания, которые охватывали:

- а) Время загрузки игры и уровней.
- б) Количество кадров в секунду (FPS) варьируется в зависимости от технических характеристик устройств и параметров графических настроек.
- в) Нагрузочный анализ уровня производительности под пиковыми условиями нагрузки для оценки взаимодействия между игровыми элементами и механиками.

#### Анализ на наличие программных дефектов и неисправностей

В эту фазу проекта входило обнаружение и исправление дефектов, способных нарушить геймплей. Применение систем учёта ошибок способствовало аккуратной фиксации и решению выявленных препятствий. Проверка охватывала множество пользовательских кейсов, охватывая:

- а) Взаимосвязь между объектами в игре.
- б) Актуализация и визуализация пользовательского интерфейса.
- в) Надежное функционирование игры при продолжительном взаимодействии и адаптации к новым игровым сценариям.
- г) Обнаруженные и устраненные ошибки:
- д) Проблема с отображением количества материалов на панели инструментов.

- е) Проблемы при транспортировке руды от дрели к плавильному агрегату.
- ж) Трудности при включении и выключении дрели.

Разработка и улучшение продуктов часто включает бета-тестирование и сбор отзывов.

По окончании фазы альфа-тестирования, в ходе которой проводились проверки внутри команды разработчиков, мы перешли к этапу бета-тестирования, привлекая к процессу целевую аудиторию. Это позволило нам собрать обратную связь и предложения для доработки проекта, обеспечивая его оптимизацию и усовершенствование перед официальным релизом. В ходе бета-тестирования акцент был сделан на:

- а) Интуитивности и удобству пользовательского интерфейса.
- б) Стабильного функционирования и высокой производительности игрового процесса на разнообразных платформах.
- в) К общей игровой практике и удовлетворению игроков.

#### Заключение

Качественное проведение тестирования нашего геймпроекта на превосходном уровне обеспечило идентификацию и устранение большинства ошибок до момента релиза. Воспользовавшись интегрированным методом тестирования и анализом обратных связей от альфа- и бета-тестеров, мы смогли разработать стабильное и высококачественное игровое приложение, которое отвечает запросам и предвкушениям аудитории. Процесс тестирования также выявил аспекты для последующего усовершенствования и проектирования развития игры, что стало ключевым элементом к ее триумфальному дебюту на рынке и обеспечению долговременного привлечения внимания игроков.

### **2.3 Перспективы развития программного продукта**

В рамках проекта разработанный симулятор производственного предприятия обладает обширными возможностями для прогресса и

оптимизации. Будущее улучшение предполагает не только инженерные инновации, но и углубление игрового материала:

1) Внедрение свежих игровых элементов и функций.

Извините, но без предоставления конкретного текста, я не смогу выполнить вашу просьбу. Вам необходимо предоставить текст, который вы хотите переписать.

а) Разнообразие буровых установок: предложение широкого спектра буров с уникальными параметрами, включая добычную скорость, энергопотребление и износостойкость. Игрокам будет предоставлена возможность подбирать оборудование в соответствии с задачами и тактикой эксплуатации.

б) Многообразие печных устройств: Печи отличаются по скорости обработки материалов, объему загрузки и потреблению ресурсов. К примеру, передовые печные агрегаты способны осуществлять быстрый переплав металлических руд, однако их эксплуатация связана с повышенным энергопотреблением.

г) Специализированные сборочные устройства: оборудование, нацеленное на работу с разнообразными компонентами или имеющее расширенные возможности, включая автоматизированную логистику комплектующих или синхронизацию с системами перемещения.

г) Дополнительные механизмы, такие как регулирование энергетического баланса, контроль за транспортными маршрутами или эффективное управление ресурсными цепочками между производственными узлами, обогатят игровую динамику. Это может охватывать проектирование инфраструктуры электроснабжения, координацию логистики и настройку доставки сырья.

2) Расширение карты и уровней.

Расширение игровой карты и интеграция дополнительных уровней предоставят игрокам шанс погрузиться в более захватывающие и многогранные миссии. Данные изменения могут охватывать:

а) Многообразии географии: арены боя предлагают различные климаты, рельефы и эксклюзивные материалы. Игрокам придется подстраиваться под новые обстоятельства и прорабатывать уникальные тактики.

б) Вызовные миссии: задачи, предусматривающие достижение специфических целей, таких как наращивание уровня производительности, эффективное распределение ограниченных активов или преодоление последствий стихийных бедствий.

в) Динамические события: интеграция стохастических элементов, например, метеорологических изменений, сейсмической активности или атак противников, оказывающих эффект на ход игры и требующих незамедлительных действий.

### 3) Разработка и внедрение научно-технического прогресса.

Разработка научно-исследовательской системы даст игрокам возможность совершенствовать свои производственные центры путем освоения передовых технологий и инновации. Возможности могут охватывать:

а) Развитие технологического древа: интеграция научных разделов, способствующих модернизации текущих средств и разработке инновационных видов техники и механизации.

б) Научно-исследовательские лаборатории: это уникальные постройки, задействованные для выполнения научных исследований, потребляющих значительные ресурсы и время, однако гарантирующих долговременные выгоды.

в) Специфические модернизации: индивидуализированные оптимизации для определенных элементов, включая повышение производственной эффективности, минимизацию энергопотребления и укрепление защитных характеристик от внешних воздействий.

### Мультиплеер.

Внедрение мультиплеерного режима обогатит игровой процесс новыми аспектами коммуникации и состязательности среди участников:

а) Коллективное руководство: Участники могут вместе руководить производством, распределяя ресурсы и обязанности, тем самым стимулируя совместную игровую активность.

б) Конкуренция: внедрение режимов, позволяющих игрокам соперничать за превосходство в аспектах производства, экономической стратегии и эффективного распределения ресурсов.

в) Интерактивность в социуме: обмен знаниями, тактическими подходами и инвентарем посредством коммуникаций в игровом чате или на тематических интернет-площадках.

5) Внедрение связи с сервисами облачных технологий.

Связывание игровых платформ с cloud-решениями расширит перспективы:

а) Синхронизация игровых достижений: реализация функции сохранения данных игры в облачном хранилище, давая возможность пользователям возобновлять игровой процесс на различных устройствах.

б) Мировые рейтинги: создание международных списков лидеров и наград, мотивирующих участников бороться за верхние строчки.

в) Обновления в облаке: игры автоматически обновляются через облачные сервисы, предоставляющие игрокам свежие улучшения и новый контент.

б) Поддержка модов и контента от пользователей.

Реализация механизма модов откроет возможности для пользователей генерировать и распространять индивидуальные модификации и улучшения в игровом процессе.

а) Разработка уникальных конструкций и игровых систем: участники получают возможность проектировать собственные добычные агрегаты, термические устройства и механизмы сборки, вводя при этом инновационные игровые процессы.

б) Создание пользовательских карт и сценариев: функционал, позволяющий игрокам разрабатывать уникальные карты и игровые моды, доступные для скачивания и использования сообществом.



в) Комьюнити моддинга: формирование динамичного сообщества любителей создания модификаций, стимулирующего непрерывное появление новинок и улучшений для игрового процесса.

7) Оптимизация визуального рендеринга и пользовательского интерфейса.

С прогрессом технологических инноваций и повышением функциональности аппаратного обеспечения, есть возможность существенно повысить качество графического оформления и пользовательского интерфейса в видеоиграх.

а) Усовершенствование визуализации: повышение качества изображения через использование моделей с большим количеством полигонов для объектов и персонажей.

б) Усовершенствованные анимации: все игровые элементы и действия теперь представлены анимациями с повышенной плавностью и реализмом.

в) Интуитивное управление: создание более интуитивно понятного и удобного интерфейса, улучшающего игровой процесс и взаимодействие пользователя с игрой.

8) Продвижение и дистрибуция.

Создание и реализация маркетинговой стратегии, включающей в себя методы продвижения, способствует притоку новых пользователей, укрепляя позиции игры на рынке:

а) Социальные медиа: интенсивное применение платформ социальных медиа в целях маркетинга игры, размещение актуальной информации, обновлений игры и налаживание контакта с аудиторией игроков.

б) Экспозиции и мероприятия по видеоиграм: присутствие на ключевых мероприятиях индустрии, включая E3 и Gamescom, позволяет представить продукт обширному кругу заинтересованных лиц.

в) Взаимодействие с инфлюенсерами: Совместная работа с известными представителями игрового сообщества, такими как блогеры и стримеры, которые

могут демонстрировать игру своей аудитории, способствуя привлечению новых пользователей.

9) Поддержка и обновления.

Постоянное обновление и поддержка гарантируют непрекращающийся интерес к видеоигре на протяжении длительного времени.

а) Обновление контента и функционала: Регулярное внедрение свежего контента, нововведений и оптимизаций для сохранения заинтересованности аудитории.

б) Быстрая реакция на обратную связь от игрового сообщества и оперативное устранение обнаруженных ошибок и программных сбоев.

в) Коммуникация с пользовательской базой: активное общение с геймерской аудиторией для сбора откликов и идей, направленных на совершенствование игрового процесса.

Эти стратегии эволюции способствуют формированию более увлекательной, многогранной и многообразной игровой среды, что способствует привлечению и удержанию обширного круга пользователей.

## ЗАКЛЮЧЕНИЕ

В рамках проекта осуществлялась многогранная работа, сфокусированная на разработке фундамента для интенсивного и захватывающего геймплея, охватывавшего следующие основные этапы:

Исследование игровых принципов в сфере крафтинга: выполнено тщательное изучение видеоигр указанного жанра, что способствовало идентификации ключевых тенденций и склонностей аудитории.

Формирование критериев для игрового дизайна: Исходя из аналитического исследования, были выработаны критерии для создания прототипа, подразумевающие разработку интуитивного интерфейса для крафтинга, а также создание оптимизированных средств контроля над производственными процессами.

Разработка игровых механик: исходя из установленных критериев, созданы основные игровые механики, в том числе механизмы взаимодействия с элементами игры, что добавило сложности и многообразия игровому опыту.

Создание игрового прототипа: сформирован функциональный макет игры, показывающий внедрение спроектированных игровых систем и нарратива. Этот макет послужил фундаментом для исходного испытания и сбора реакций от потенциальных пользователей.

Завершение проекта подтвердило, что исходная задача была успешно решена: создан начальный вариант игры в категории Крафтинг, охватывающий важные элементы, необходимые для развития в полноценный игровой продукт. Реализация целей выпускной квалификационной работы не только позволила претворить в жизнь академические знания, но и обогатилась значимым опытом в сфере создания игр и их дизайна.

Следовательно, завершающая академическая диссертация подтверждает эффективность использования интегрированной стратегии в разработке элементов геймплея и игрового мира, тем самым раскрывая перспективы для его

последующего совершенствования и прототипирования, а также для изучения новых направлений в дизайне игр.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. YoYo Games. (2023). GameMaker Studio 2 Manual. Available at [Электронный ресурс]. URL: <https://manual.yoyogames.com> (дата обращения 23.05.2024)
2. YoYo Games (2022). Official GameMaker Blog. Available at [Электронный ресурс]. URL: <https://www.yoyogames.com/blog> (дата обращения 23.05.2024)
3. GameMaker Community Forums (2023). Available at [Электронный ресурс]. URL: <https://forum.yoyogames.com> (дата обращения 23.05.2024)
4. Gamasutra (2023). The Art & Business of Making Games. Available at [Электронный ресурс]. URL: <https://www.gamasutra.com> (дата обращения 23.05.2024)
5. Shaun Spalding's YouTube Channel (2023). Tutorials on GameMaker Studio 2. Available at [Электронный ресурс]. URL: <https://www.youtube.com/user/999Greyfox> (дата обращения 23.05.2024)
6. HeartBeast's GameMaker Tutorials (2023). Available at [Электронный ресурс]. URL: <https://www.youtube.com/c/uheartbeast> (дата обращения 23.05.2024)
7. GameDev.net (2023). Resources and Community for Game Developers. Available at [Электронный ресурс]. URL: <https://www.gamedev.net> (дата обращения 23.05.2024)
8. YoYo Games (2023). Official GameMaker Tutorials and Demos. Available at [Электронный ресурс]. URL: <https://www.gamedev.net> (дата обращения 23.05.2024)
9. Udemy (2023) Complete GameMaker Studio 2 Guide: From Beginner to Pro. Online Course. Available at [Электронный ресурс]. URL: <https://www.udemy.com> (дата обращения 23.05.2024)
10. Шелл Дж. "Искусство создания игр" / Дж. Шелл. — Санкт-Петербург: Питер, 2020. — 448 с.

11. Бертон Л. "Программирование игр под Android" / Л. Бертон. — Москва: Эксмо, 2018. — 320 с.
12. Тодд Г. "Level Up! Гайд по разработке компьютерных игр" / Г. Тодд. — Москва: Манн, Иванов и Фербер, 2019. — 464 с.
13. Костер Р. "Теория игр: Основы успеха в проектировании видеоигр" / Р. Костер. — Москва: Альпина Паблишер, 2020. — 336 с.
14. Скотт Роджерс. "Level Up! Гид по созданию великолепных видеоигр" / Роджерс С. — Москва: Эксмо, 2016. — 400 с.
15. Джесси Шелл. "Искусство создания игр" / Шелл Дж. — Москва: ДМК Пресс, 2016. — 512 с.
16. Эрнест Адамс, Эндрю Роллингс. "Основы проектирования компьютерных игр" / Адамс Э., Роллингс Э. — Санкт-Петербург: Питер, 2017. — 576 с.

Выпускная квалификационная работа выполнена мной самостоятельно.  
Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

«\_\_»\_\_\_\_\_2024 г. \_\_\_\_\_ Егоркин Антон Дмитриевич