

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования

«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ
Школа бакалавриата

ОТЧЕТ

По проекту
«Создание игрового ПО для ПК в жанре RTS»
по дисциплине «Проектный практикум»

Заказчик: доц. к.т.н. Исаков Дмитрий Николаевич
ФГАОУ ВО «УрФУ имени первого Президента России Б.Н.
Ельцина»

Куратор: доц. к.т.н. Исаков Дмитрий Николаевич

Студенты команды
SHARKS_Team: Якупов Артемий Ильич РИ-230941

Нагаев Николай Анатольевич РИ-230950

Привалов Вадим Александрович РИ-230914

Куликов Иван Григорьевич РИ-230936

Усольцев Матвей Александрович РИ-230950

Содержание

ВВЕДЕНИЕ	3
ОСНОВНАЯ ЧАСТЬ	5
АНАЛИТИЧЕСКАЯ ЧАСТЬ	10
Существующие подходы к решению задачи	10
Ограничения проекта	11
Стек для разработки	11
Анализ существующих проектов:	14
ОПИСАНИЕ РАБОТЫ	16
Якупов Артемий Ильич	19
Привалов Вадим Александрович	18
Нагаев Николай Анатольевич	20
Куликов Иван Григорьевич	21
Усольцев Матвей Александрович	22
ВЫВОДЫ И ИТОГИ	23
ЗАКЛЮЧЕНИЕ	25
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	26

ВВЕДЕНИЕ

1. Цель и задачи проекта

Основной целью проекта является разработка стратегической игры в реальном времени (RTS) Corrupted Realm, сочетающей элементы градостроительного симулятора и тактической обороны. Игра погружает игрока в средневековый мир, разрушенный древней скверной, где необходимо восстанавливать базы, управлять ресурсами, защищаться от искажённых существ и очищать заражённые территории.

Задачи проекта:

Реализация базовых игровых механик: сбор ресурсов (золото), строительство сооружений, управление юнитами и их улучшение. Создание сюжетной кампании из пяти уровней с уникальными целями, врагами и локациями. Разработка процедурной генерации вражеских волн для повышения реиграбельности. Балансировка экономических и боевых систем, включая характеристики юнитов, врагов и сооружений. Интеграция мультиплатформенного релиза (Steam, VKPlay) с поддержкой модификаций и пользовательского контента.

2. Актуальность и важность проекта

Жанр RTS сохраняет популярность благодаря таким хитам, как Age of Empires и Frostpunk, однако рынок испытывает дефицит проектов, сочетающих глубокий стратегический геймплей с уникальным сюжетным сеттингом. Corrupted Realm предлагает инновационные механики: Систему очистки территорий через строительство Храмов Чистоты. Процедурную генерацию врагов, обеспечивающую разнообразие игровых сессий. Нелинейный финал с моральными выборами, влияющими на исход игры. Проект актуален для русскоязычной аудитории (через VKPlay) и международного сообщества (через Steam), ориентируясь на поклонников тактических вызовов и средневековой эстетики.

3. Область применения программного продукта

Игра предназначена для платформы Windows и рассчитана на широкий круг пользователей. Развлекательная индустрия: Основная аудитория — игроки 20–40 лет, предпочитающие стратегии с глубоким управлением и сюжетом. Киберспортивное сообщество: Возможность создания пользовательских карт и модов через Steam Workshop поддерживает долгосрочный интерес. Образовательный аспект: Элементы исторического сеттинга и управления ресурсами могут использоваться в игровых обучающих программах.

4. Ожидаемые результаты и планируемые достижения

По завершении проекта планируется: Выпуск рабочей версии игры с пятью уровнями, базовыми механиками и сбалансированными системами. Запуск на Steam и VKPlay с поддержкой DLC (дополнительные кампании, фракции) и косметических обновлений. Формирование активного сообщества через бета-тесты, стримы и интеграцию с социальными сетями. Достижение коммерческого успеха за счёт премиум-модели (цена 200 ₽) и локализации для СНГ. Ключевым показателем успеха станет положительная обратная связь от игроков, подтверждающая глубину геймплея и уникальность концепции.

ОСНОВНАЯ ЧАСТЬ

Требования заказчика:

Создать RTS-игру с элементами градостроительного симулятора в средневековом сеттинге. Реализовать базовые механики: управление ресурсами, строительство зданий, тактические бои, процедурную генерацию врагов.

Требования пользователей:

Глубокий стратегический геймплей с балансом между экономикой и боями. Увлекательный сюжет с нелинейным финалом. Оптимизация под средние ПК.

Backlog. Основные задачи:

Реализация системы ресурсов (золото, артефакты). Разработка процедурной генерации вражеских волн. Создание пяти уровней с уникальными механиками (например, очистка территорий через Храмы Чистоты).

Дополнительные задачи:

Интеграция Steam Workshop для пользовательских модов.

Добавление мультиязычной поддержки (русский, английский).

Критические требования:

Балансировка характеристик юнитов и врагов.

Тестирование на производительность и устранение багов.

Анализ и сопоставление аналогов:

Stronghold (Firefly Studios)

Сильные стороны: Углублённое градостроительство, исторический сеттинг.

Слабые стороны: Линейный сюжет, отсутствие процедурной генерации.

Frostpunk (11 bit studios)

Сильные стороны: Атмосферный сеттинг, моральные выборы.

Слабые стороны: Ограниченная боевая система.

Age of Empires IV (Relic Entertainment)

Сильные стороны: Баланс стратегии и тактики, мультиплеер.

Слабые стороны: Высокие системные требования.

Уникальность Corrupted Realm:

Гибкий геймплей: Сочетание строительства базы, тактических боёв и сюжетных выборов.

Нелинейный финал: Три варианта завершения игры (уничтожение, подчинение, бегство).

Локализация для СНГ: Акцент на русскоязычную аудиторию через VKPlay.

Обзор архитектуры программного продукта, описание основных компонентов и связей между ними, обоснование выбора архитектурного решения.

Основные компоненты:

Игровой движок: Unity (C#) — выбран за кроссплатформенность и поддержку 3D-графики.

Ядро игры:

Система сетки: Основа для размещения зданий и юнитов.

Генератор врагов: Алгоритмы процедурного создания волн на основе сложности уровня.

Экономическая модель: Баланс добычи золота, затрат на строительство и улучшения.

Визуальная часть: 3D-модели зданий и юнитов (Blender). Эффекты скверны и очистки (шейдеры Unity).

Пользовательский интерфейс: Интерактивная карта, панель управления ресурсами, меню улучшений.

Взаимодействие компонентов: Игрок управляет юнитами через интерфейс → команды передаются в ядро → изменения отображаются в визуальной части. Генератор врагов взаимодействует с системой сетки, определяя точки спавна.

Обоснование архитектурных решений: Использование Unity позволило сократить время разработки за счёт готовых ассетов и инструментов. Система сетки упростила реализацию строительства и тактического планирования.

Описание методологии разработки, информация о процессе разработки, отчет о результатах тестирования на промежуточных этапах, разбор выявленных ошибок.

Этапы разработки:

Pre-production: Создание дизайн-документа, концепт-артов.

Production: Реализация механик, моделирование уровней, программирование ИИ.

Тестирование: Балансировка, поиск багов, оптимизация.

Методы:

Agile с двухнедельными спринтами.

Итеративное тестирование: юнит-тесты для механик, playtest-сессии с фокус-группой.

Выявленные ошибок:

Баг №1: Юниты игнорировали препятствия.

Решение: Настройка коллайдеров и алгоритмов поиска пути (A*).

Баг №2: Утечка памяти при генерации врагов.

Решение: Оптимизация пула объектов.

Результаты тестирования:

Средний FPS на ПК среднего уровня: 45–60 (после оптимизации).

Удовлетворённость игроков сюжетом: 4.2/5 (по данным опроса фокус-группы).

Информация о планировании деятельности в ходе разработки и распределении задач между участниками команды разработчиков. Команда разработчиков:

Геймдизайнер: Создание сюжета, балансировка механик.

Программисты: Реализация ядра игры, интеграция Steam/VKPlay.

3D-художник: Моделирование зданий, юнитов, эффектов.

Тестировщики: Поиск багов, сбор обратной связи.

План работ:

1–4 неделя: Прототип базовых механик (сбор ресурсов, строительство).

5–8 неделя: Разработка уровней, интеграция сюжета.

9–12 неделя: Тестирование, оптимизация, подготовка к релизу.

Инструменты управления:

Striveapp: Для трекинга задач.

GitHub: Совместная работа над кодом.

VK: Еженедельные созвоны для обсуждения прогресса.

АНАЛИТИЧЕСКАЯ ЧАСТЬ

Существующие подходы к решению задачи

Разработка игр в жанре RTS основана на трёх основных аспектах: игровом дизайне, технической реализации и UX/UI. Рассмотрим существующие подходы:

Игровой дизайн:

Большинство современных RTS используют асимметричные игровые фракции, каждая из которых обладает уникальными способностями, юнитами и стилем игры (например, "StarCraft"). Это создаёт стратегическую глубину, но усложняет балансирование.

Подход к экономике варьируется: некоторые проекты делают акцент на многоресурсных системах (например, "Age of Empires"), другие упрощают её до одного ресурса для упрощения управления (например, "Command & Conquer").

Системы боевой механики нередко базируются на принципе "камень-ножницы-бумага", где каждая единица эффективна против определённых типов врагов.

Техническая реализация:

Механика RTS требует высокой производительности из-за одновременного отслеживания множества объектов.

Подходы включают использование оптимизированных движков (Unity, Unreal Engine) или собственных решений.

Для ИИ широко применяются поведенческие деревья и алгоритмы A*, обеспечивающие реалистичное поведение юнитов.

UX/UI:

Большое внимание уделяется упрощению интерфейса: использование контекстных меню и быстрых клавиш.

Игроки позитивно воспринимают обучающие уровни, встроенные подсказки и систему прогрессивного усложнения.

Ограничения проекта

Технические ограничения:

Небольшая команда разработчиков и сжатые временные рамки ограничивают возможность создания масштабного игрового продукта с большим количеством механик.

Контентные ограничения:

Простой визуальный стиль из-за отсутствия бюджета на платные 3D-модели и анимацию.

Баланс и тестирование:

Отсутствие времени и доступа к массовой аудитории для полноценного тестирования баланса и геймплея.

Стеки для разработки

Unity — это кроссплатформенный игровой движок, разработанный компанией Unity Technologies, поддерживающий 2D и 3D графику, скрипты на C# и обладающий широкими возможностями для мультиплатформенной разработки. Unity предоставляет мощные инструменты для создания игр, включая редактор сцены, анимации, физику и возможность легкой интеграции пользовательских ассетов. Широкая поддержка сообщества, обширная документация и гибкость движка делают его идеальным выбором для разработки и тестирования RTS игр.

C# — это объектно-ориентированный язык программирования, разработанный компанией Microsoft. Он используется для написания скриптов в Unity и обеспечивает высокую производительность и гибкость. C# является основным языком программирования для Unity, предоставляя разработчикам мощные инструменты для создания сложных игровых механик и систем. Он легко изучается и поддерживается большим количеством ресурсов и документации.

GitHub — это веб-сервис для хостинга IT-проектов и их совместной разработки, использующий систему контроля версий Git. Она позволяет хранить код, отслеживать изменения и взаимодействовать с другими разработчиками. GitHub обеспечивает надежное хранение кода, управление версиями и совместную работу над проектом. Это позволяет команде синхронизировать изменения, отслеживать историю разработки и легко управлять проектом.

Visual Studio — это интегрированная среда разработки (IDE) от компании Microsoft, поддерживающая C# и множество других языков программирования. Visual Studio предоставляет мощные инструменты для написания, отладки и тестирования кода. Это улучшает качество и производительность разработки, предоставляя разработчикам все необходимые инструменты в одной среде.

Sketchfab — это онлайн-платформа, где можно просматривать, скачивать и загружать 3D-модели. Она поддерживает широкий спектр форматов моделей и позволяет интегрировать их в проекты, включая игры и визуализации. Sketchfab предлагает огромную библиотеку 3D-моделей высокого качества, включая бесплатные и коммерческие варианты. Удобная система предпросмотра моделей прямо в браузере помогает быстро оценить их качество и соответствие требованиям проекта. Это сокращает время на создание уникального контента и упрощает процесс наполнения проекта визуальными элементами.

Unity Asset Store — это онлайн-магазин, предлагающий широкий выбор ассетов для Unity, включая модели, текстуры, звуковые эффекты и скрипты.

Почему выбрали: Unity Asset Store позволяет быстро находить и интегрировать готовые ассеты в проект, что ускоряет процесс разработки и позволяет сосредоточиться на создании уникальных аспектов игры.

Blender — это мощный инструмент для создания 3D-графики, включающий функции моделирования, анимации, текстурирования, рендеринга и работы с видеоредакцией. Это бесплатное программное обеспечение с открытым исходным кодом, поддерживаемое активным сообществом разработчиков и художников.

Почему выбрали: Blender позволяет создавать уникальные 3D-модели, анимации и визуальные эффекты с нуля, что делает его незаменимым инструментом для кастомизации контента. Его бесплатная лицензия и обширный функционал делают Blender доступным для разработки проектов любого масштаба, от игр до сложных визуализаций.

Анализ существующих проектов

Проанализированы популярные RTS-игры, выявлены их сильные и слабые стороны. Это позволило определить, что игра должна сочетать стратегическую глубину с доступностью для новичков.

Frostpunk — игра о строительстве города в замерзшем мире, где игрок управляет ресурсами, инфраструктурой и принимает сложные решения для выживания. Как и наша игра, она имеет постапокалиптический сеттинг и акцент на управлении ресурсами. Однако в Frostpunk основной геймплей сосредоточен на строительстве города, тогда как наша игра включает контроль персонажей и боевые действия, делая процесс более разнообразным.

Cataclismo — RTS, где игроки строят замки и защищают их от врагов. Как и наша игра, она имеет постапокалиптический сеттинг с акцентом на строительство и оборону. Однако в Cataclismo основной упор делается на фортификацию, тогда как в нашей игре добавлены исследование мира и управление персонажами, что делает её более динамичной.

They Are Billions — RTS в стимпанковском мире, где игроки строят базы и защищают их от зомби. В отличие от акцента на оборону и выживание в They Are Billions, наша игра включает исследование и расширение территорий.

Ancestors Legacy — RTS о Средних веках, сосредоточенная на управлении армиями и завоевании. В отличие от исторической точности и традиционного стиля этой игры, наша будет разворачиваться в футуристическом постапокалипсисе.

Вывод:

Мы взяли основу RTS-геймплея с элементами защиты крепости из Cataclismo и постапокалиптический сеттинг с механиками выживания из Frostpunk, добавив Sci-Fi стилистику и расширив возможности игрока через исследование и управление.

Выводы по аналитической части:

На основе анализа и применённых методов был сформирован подход к разработке, который сочетает теоретические знания и их практическую реализацию. Учёт существующих ограничений позволяет сконцентрироваться на наиболее значимых аспектах: глубина стратегических решений, стабильность базовых механик и удобство управления. Благодаря использованию доступных инструментов и итеративного подхода команда уверенно продвигается к созданию игрового продукта, который может быть расширен в дальнейшем.

ОПИСАНИЕ РАБОТЫ

Информация о работе каждого участника в отдельности.

В рамках разработки Corrupted Realm были выполнены конкретные действия, которые привели к созданию измеримых и работающих результатов:

Проектирование концепции и базовой документации

Проведено несколько созвонов, результатом которых стало утверждение сеттинга, базовых механик и структуры игры. Также создан дизайн документ, в котором описаны: типы ресурсов и их функции, основные механики боя и управления базой.

Результат: Чётко структурированное описание проекта, которое использовалось всеми участниками команды для согласованной работы.

Создание прототипа

На движке Unity создан базовый прототип игры с функционалом: вражеские юниты (базовый ИИ, спавн, передвижение, атака), постройка зданий и добыча ресурсов, генерация грида. Прототип был предназначен для тестирования основных игровых механик и проверки жизнеспособности концепта. Изначально возникли проблемы с синхронизацией передвижения юнитов по сложным ландшафтам. Это было решено использованием A*Pathfinding project.

Результат: Рабочий прототип, демонстрирующий базовые механики игры, пригодный для внутреннего тестирования.

Разработка визуального оформления

С помощью Blender созданы 3D-модели зданий, юнитов и объектов окружения. Также, для экономии времени использовались готовые ассеты из открытых библиотек. Добавлены базовые текстуры и шейдеры, чтобы улучшить визуальное восприятие. Проработан минималистичный интерфейс с акцентом на читаемость и удобство управления. Ограниченный бюджет на

графику заставил отказаться от сложных анимаций. Решением стало использование упрощённой, но стилизованной графики, что также положительно сказалось на производительности.

Результат: Эстетически простой, но функциональный и понятный визуальный стиль игры, готовый для тестирования.

Балансировка игровых систем

Проведено несколько внутренних тестов, чтобы выявить проблемы с игровым балансом. Например, стоимость одних юнитов была не сильно больше других, но их эффективность была в разы больше в обороне базы, что исправили, увеличив их стоимость. Скорость добычи ресурсов показала себя слишком медленной, что замедляло геймплей. Внесены корректировки в скорость добычи и количество ресурсов. Балансирование без участия внешних тестировщиков оказалось сложным, так как команда привыкла к механикам игры. Это отразилось на скорости корректировок, но позволило выявить основные проблемы.

Результат: Заметно улучшенный баланс, в котором игроки могут пробовать разные стратегии, сохраняя равные шансы на победу.

Привалов Вадим Александрович

Роль — тимлид, разработчик.

В рамках проекта отвечал за следующие аспекты разработки игры:

Организация командной работы:

Выступал связующим звеном между куратором и командой: предоставлял отчеты о групповой работе и транслировал организационную информацию участникам проекта. Планировал и проводил еженедельные совещания для обсуждения текущего прогресса, анализа трудностей и постановки задач. Разработал систему управления задачами: назначал индивидуальные задания через Strive App, формировал семестровый план работ и контролировал его выполнение.

Техническая реализация:

Исправление критических багов и оптимизация существующего кода. Координация работы разработчиков: консультирование по архитектуре решений и обеспечение соблюдения сроков.

Трудности и их преодоление:

Основные сложности возникли в процессе отладки игровых механик, что потребовало значительных временных затрат. Благодаря слаженному взаимодействию с командой удалось систематизировать процесс тестирования и ускорить устранение ошибок.

Результаты работы:

Налажена эффективная система коммуникации и контроля задач. Стабилизирована игровая логика за счет устранения ключевых багов. Развиты навыки управления командой и углублено понимание работы в Unity. Обеспечено своевременное выполнение этапов проекта в соответствии с утвержденным планом.

Якупов Артемий Ильич

Роль — второй разработчик.

В рамках проекта отвечал за следующие аспекты разработки игры:

Значительно углубил знания в области работы с игровым движком Unity, что позволило освоить ключевые аспекты геймдева. На начальном этапе принимал активное участие в формировании концепции, включая определение визуального стиля и общего дизайна игры.

Отвечал за проектирование и реализацию игровых построек, а также за обеспечение их функциональности. Среди ключевых задач:

Разработка главного здания, играющего центральную роль в игровом процессе.

Создание второстепенных объектов, включая оборонительные сооружения (стены и пушки), казармы для персонажей и храм света с особой игровой механикой.

Обеспечение корректной работы пользовательского интерфейса (UI) для улучшения взаимодействия игроков с проектом.

В ходе работы сталкивался с трудностями, связанными с оптимизацией и интеграцией объектов, однако благодаря совместной работе с командой удалось успешно реализовать все запланированные системы.

Результатом работы стали: Полностью функциональные игровые объекты и механики. Участие в формировании концепции и визуального стиля игры. Поддержка командной разработки и решение возникающих технических задач.

Нагаев Николай Анатольевич

Роль — аналитик, геймдизайнер.

В рамках проекта отвечал за следующие аспекты разработки игры:

Создание дизайн-документа, включающего все важные элементы геймплея, интерфейсы и технические требования.

Разработка захватывающего сюжета, который стал основой для всего игрового опыта.

Концептуализация игровых механик, формирование концепции core-геймплея и создание уникального игрового мира.

Детальная проработка элементов игры и всех возможных взаимодействий игрока, чтобы сделать игровой процесс глубоким и увлекательным.

Анализировал конкурентов, чтобы выявить их сильные и слабые стороны, для того, чтобы в нашей игре было всё самое лучшее от различных представителей жанра.

В основном сталкивался с трудностями в виде недостатка информации, будь то для создания дизайн-документа или интересных механик игры, помогли разобраться с этими трудностями сокомандники, которые помогли найти нужную информацию. Кроме того, активно поддерживал команду, оказывая помощь в решении разнообразных задач, возникающих в процессе работы над проектом.

Результатом работы являются: создание готового дизайн-документа, данного отчета и помощь команде по иным вопросам.

Куликов Иван Григорьевич

Роль — 3Д дизайнер.

В роли 3Д дизайнера были выполнены следующие задачи:

С помощью Blender была изменена часть визуального стиля игры, изменена текстура на тайлах игровой сетки. Также был изменен layout карты, добавлены источники ресурсов и больше вариативности в структуре растительности и пейзажа. Были найдены, обработаны и частично анимированы модели нескольких новых противников-роботов, атакующих постройки игрока. На данной итерации был выполнен саунд-дизайн, подбирались и генерировались звуковые сэмплы, которые впоследствии объединились для создания звуковых эффектов для постройки зданий, нажатия на кнопки, а также фоновой музыки в меню и самой игре.

Трудности и их решение:

Проблема: Иногда в сети отсутствовали необходимые модели, анимации и текстуры, и их поиск сильно замедлял процесс разработки. Кроме того, нередко они не соответствовали стилю нашей игры.

Решение: Чтобы решить эту проблему, было принято решение приступить к созданию части моделей с нуля, а также некоторых анимаций для моделей с rig'ом и покраской текстур. Для создания моделей использовался Blender 3D, применяя модификаторы, способствующие соблюдению единого стиля и оптимизации. Используя некоторые уже имеющиеся анимации как отправные и конечные точки, были анимированы руки роботов для атаки построек.

Итог: Благодаря такому подходу игра стала гораздо более цельной и стильной, так как все элементы визуального оформления теперь гармонично сочетаются друг с другом. Кроме того, создание собственных ассетов позволило лучше оптимизировать их, снизив нагрузку на ПК.

Усольцев Матвей Александрович

Роль — первый разработчик.

В рамках проекта отвечал за следующие аспекты разработки игры:

Доработка игровых механик: Улучшение логики поведения юнитов и противников, включая систему выбора целей, движение к ним и атаку.

Реализация мини-карты с отображением союзных и вражеских юнитов, а также ключевых построек.

Визуальная и интерфейсная часть: Добавление 3D-моделей построек, персонажей и врагов. Создание и интеграция анимаций для моделей.

Разработка черновой версии главного меню, включающего: Кнопку "Новая игра" для запуска игрового процесса. Кнопку "Настройки" для перехода в меню параметров. Кнопку "Выход" для завершения работы с игрой.

Трудности и их решение:

В процессе работы столкнулся с необходимостью оптимизации алгоритмов поиска целей и синхронизации анимаций с игровыми событиями. Благодаря совместной работе с командой удалось найти эффективные решения и реализовать все запланированные функции.

Результаты работы:

Улучшенные и стабильные механики поведения юнитов. Функциональная мини-карта для удобства навигации. Готовые 3D-модели и анимации. Базовый интерфейс главного меню.

ВЫВОДЫ И ИТОГИ

Успехи проекта

Основная цель разработки минимально жизнеспособного продукта (MVP) достигнута: Реализована базовая функциональность, включая управление юнитами, строительство базы, добычу ресурсов и систему боёв. Создана игровая среда с работающим балансом между фракциями и экономическими системами. Подготовлен визуальный стиль, который, несмотря на упрощённость, успешно передаёт атмосферу игры. Прототип показал, что концепция имеет потенциал, а заложенные механики способны обеспечить увлекательный и тактический игровой процесс. Итеративный подход к разработке позволил оперативно выявлять и устранять проблемы, улучшая качество продукта на каждом этапе.

Извлечённые уроки и трудности

Баланс: Балансировка игровых систем оказалась сложной задачей из-за отсутствия массового тестирования. В дальнейшем необходимо предусмотреть возможность привлечения большего числа тестировщиков, чтобы получить репрезентативные данные о геймплее.

ИИ: Реализация более продвинутого искусственного интеллекта требует большего времени и ресурсов. Упрощённый ИИ в текущей версии работает, но его ограниченные возможности снижают уровень вызова для игроков.

Обучение: Пользовательский опыт (UX) критически важен для RTS-игры. Хотя обучение стало функциональным, дальнейшая доработка туториалов и подсказок поможет сделать игру доступной для широкой аудитории.

Ограничения ресурсов: Ограниченный бюджет и небольшая команда заставили сосредоточиться на ключевых аспектах проекта, что помогло избежать избыточного усложнения, но ограничило глубину проработки некоторых элементов.

Итоги работы

Команде удалось успешно применить теоретические знания на практике, реализовать рабочий прототип игры и изучить особенности разработки RTS-игр. Итоговая версия продукта готова к дальнейшему развитию, включая добавление новых фракций, улучшение ИИ, расширение контента (карты, миссии, юниты, особые события) и графической составляющей. Разработка выявила ключевые области, требующие улучшений: Углубление стратегической глубины и тактического разнообразия. Доработка интерфейса для более сложных сценариев игры. Улучшение производительности для масштабных сражений.

Планы на будущее

Провести внешний плейтест с привлечением целевой аудитории, чтобы получить обратную связь и проверить, насколько текущие механики соответствуют ожиданиям игроков.

Добавить новые игровые элементы: уникальные способности фракций, больше типов юнитов и зданий, а также разнообразные сценарии миссий.

Продолжить доработку ИИ, чтобы он мог адаптироваться к различным стилям игры, делая каждую партию более непредсказуемой.

ЗАКЛЮЧЕНИЕ

Оценка соответствия программного продукта поставленным требованиям заказчика и пользователя, анализ выполненных требований.

Тимлид научился управлять командой и планировать проект, геймдизайнер – балансировке и дизайну уровней, аналитик – сбору и анализу данных, а программист – оптимизации кода и разработке AI. Все улучшили навыки командной работы и решения сложных задач. Разработка RTS-игры показала, что даже в условиях ограниченных ресурсов можно создать работающий и потенциально интересный продукт. Проект заложил основу для дальнейшего расширения, а также дал команде важный опыт работы с игровыми механиками, балансом и взаимодействием игроков с игрой. Этот опыт может быть успешно применён в будущих проектах, повышая шансы на их успешную реализацию.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Создание базы RTS игры на Unity [Электронный ресурс] / URL: https://youtu.be/-GfdKB_7mrY?si=htT0nCsAtMweRgI4 (дата обращения: 07.10.2024).

Официальная документация Unity [Электронный ресурс] / Unity Technologies. URL: <https://docs.unity3d.com/ru/530/Manual/index.html> (дата обращения: 02.10.2024).

3D анимация на Unity [Электронный ресурс] / URL: <https://youtu.be/vApG8aYD5aI?si=5mStaJD4gELMfcBJ> (дата обращения: 11.11.2024).

Почему Warcraft 3 до сих пор популярен [Электронный ресурс] / Ярослав. URL: <https://dzen.ru/a/XZjNMBGNfwCthfCR> (дата обращения: 22.10.2024).

Десять заповедей создания RTS [Электронный ресурс] / Ярослав. URL: <https://dtf.ru/u/31295-yaroslav/2469313-desyat-zapovedei-sozdaniya-rts-i-istoriya-razrabotki-age-of-empires> (дата обращения: 16.11.2024).

Unity Tutorials [Электронный ресурс] / Официальный сайт Unity. URL: <https://learn.unity.com/tutorial/create-a-real-time-strategy-game-part-1> (дата обращения: 11.10.2024).

GameDev.net [Электронный ресурс] / Сообщество разработчиков игр. URL: <https://www.gamedev.net/categories/rtg/> (дата обращения: 12.10.2024).

Reddit r/gamedev [Электронный ресурс] / Подфорум Reddit, посвященный разработке игр. URL: <https://www.reddit.com/r/gamedev/> (дата обращения: 11.10.2024).

RTS Game Design Forum [Электронный ресурс] / Форум по дизайну RTS-игр. URL: <http://www.rtsgdesignforum.com/> (дата обращения: 11.10.2024).

Artificial Intelligence for Games / R. Millington, J. Funge. – 2-е изд. – Burlington: Elsevier, 2016. – 504 с.

Designing Real-Time Strategy Games / Gamasutra. URL: https://www.gamasutra.com/view/feature/131137/designing_realtime_strategy_games.php (дата обращения: 11.10.2024). Game AI Pro: Collected Wisdom of Game AI Professionals / S. Rabin (ed.). – CRC Press, 2015. – 416 с.

Game Programming Patterns / R. Nystrom. – Genever Ben, 2014. – 338 с.