

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Алтайский государственный университет»
Школа развития цифровых компетенций «Digital Up» (цифровая кафедра)

Отчет о выполнении производственной практики по ДПП ПП
«Основы Gamedev и VR-разработки»
«РАЗРАБОТКА ПОШАГОВОЙ БОЕВОЙ СИСТЕМЫ»

Исполнитель:
Скоснягин Михаил Сергеевич

Руководитель проекта
Козлов Д.Ю.

г. Барнаул 2025

Введение

Целью практического проекта являлась разработка функционального прототипа пошаговой стратегической игры с элементами тактического боя. Основной фокус был направлен на реализацию ключевых механик, указанных в техническом задании: системы ограниченных ходов (10 ходов), механики действий юнитов (атака, перемещение, суперудар), системы карт, модифицирующих характеристики отряда, и определения победителя на основе суммарного здоровья отрядов. Проект нацелен на демонстрацию практических навыков разработки игровой логики, работы с Unity, программирования на C# и решения комплексных задач игрового дизайна в условиях, приближенных к реальному рабочему процессу.

Задачи проекта:

1. Проектирование архитектуры игры
2. Реализация базовых игровых объектов и механик
3. Разработка системы ходов и фаз игры
4. Реализация механики действий юнитов
5. Разработка системы выбора карт
6. Реализация определения победителя и UI

Подготовка перед разработкой прототипа

Концепция и геймдизайн

Проект представляет собой пошаговую тактическую игру с элементами классического фэнтезийного сеттинга. Основная механика строится на управлении отрядом из трёх уникальных юнитов (Танк, Стрелок, Маг), каждый из которых обладает своими характеристиками, сильными и слабыми сторонами. Игра ведётся на поле 7×7 клеток, что обеспечивает достаточно пространства для манёвров, но не делает игру излишне затянутой.

Ключевые референсы и вдохновение:

Шахматы – взята базовая пошаговая структура, где каждый юнит имеет свои правила перемещения и атаки. Однако, в отличие от шахмат, здесь добавлены дополнительные механики.

Классические RPG, тактические игры и архетипы персонажей: танк (Рыцарь) – высокий уровень защиты, медленное перемещение (1 клетка), мощная ближняя атака. Стрелок (Лучник/Разбойник) – средние показатели, дальняя атака по прямой (как ладья в шахматах), увеличенная зона перемещения (2 клетки). Маг (Волшебник) – слабая защита, но уникальная механика атаки (может бить любую цель) и мощный групповой суперудар.

Баланс и игровая механика:

Продумана простая карточная система. В начале каждого хода игроки получают по 2 случайные карты, но выбирают только одну. Эффекты карт временные (действуют 1 ход), но могут кардинально изменить ситуацию усилением атаки/защиты юнитов, лечением или увеличением дальности хода и зарядом суперудара.

Суперудары активируются после накопления заряда (3 успешные атаки). У каждого юнита свой уникальный эффект: танк – мощный удар с увеличенной дальностью, стрелок – атака + лечение союзников (поддержка), маг – телепортация + урон всем врагам (контроль поля). Это поощряет агрессивную игру, так как без атак невозможно использовать суперспособности.

Разработка прототипа пошаговой боевой системы

Основой игрового процесса стала четко структурированная система взаимодействующих компонентов. На первом этапе создавалось ядро игры - модуль GameManager, который взял на себя контроль над всеми ключевыми процессами: сменой фаз игры (выбор карт, ходы игроков), подсчетом ходов и определением условий победы. Этот центральный скрипт координировал работу всех остальных систем, выступая своеобразным дирижером игрового процесса.

Для реализации игрового поля был разработан GridManager, который инициализировал сетку 7×7 и управлял всеми клетками. Каждая клетка представляла собой отдельный объект с прикрепленным скриптом GridCell, отвечающим за визуальную подсветку, обработку кликов и хранение информации о находящемся на клетке юните. Особое внимание уделялось системе навигации - методы проверки доступности клеток для перемещения и атаки были вынесены в отдельные функции для обеспечения гибкости и простоты модификации.

Система юнитов строилась вокруг базового класса Hero, содержавшего все основные параметры персонажей: здоровье, показатели атаки и защиты, дальность перемещения. Каждый тип юнита (Танк, Стрелок, Маг) имел свои уникальные особенности в методах атаки и перемещения, реализованные через систему условных проверок. Например, метод CanAttack для Стрелка проверял, находится ли цель на одной линии, а для Мага таких ограничений не было. Логика суперспособностей была вынесена в отдельные методы (TankUltimate, ShooterUltimate, MageUltimate), что упрощало балансировку и модификацию каждой способности.

Особую сложность представляла реализация системы действий игрока, за которую отвечал ActionSystem. Этот компонент управлял последовательностью активации юнитов (строгий порядок: Танк → Стрелок → Маг), обрабатывал выбор действий через UI-панель и координировал подсветку доступных клеток. При выборе действия (атака, перемещение или

суперудар) система запрашивала у GridManager список доступных клеток и визуализировала их, используя разные цвета для разных типов действий.

Карточная система, реализованная в CardManager, состояла из нескольких ключевых элементов: базы данных всех возможных карт, системы случайной генерации предложений и механизма применения эффектов. Особенностью реализации стало использование делегатов для описания эффектов карт, что позволяло легко добавлять новые карты без изменения основной логики игры. Каждая карта при выборе немедленно применяла свой эффект ко всем юнитам соответствующего отряда, модифицируя их параметры до конца текущего хода.

Интерфейс пользователя был тесно интегрирован с основной игровой логикой. UI-элементы не просто отображали информацию, но и активно участвовали в игровом процессе. Например, панель действий динамически обновляла доступные кнопки в зависимости от выбранного юнита и текущей ситуации на поле. Особое внимание уделялось визуальной обратной связи - все действия сопровождалось четкими индикаторами: изменение здоровья, заполнение шкалы суперудара, подсветка доступных ходов.

Отдельно стоит отметить систему обработки ввода, которая обеспечивала интуитивное управление. Клик по юниту выбирал его для действий, клик по подсвеченной клетке инициировал соответствующее действие (перемещение или атаку). Все взаимодействия проходили через единую точку входа в ActionSystem, что гарантировало согласованность состояний и предотвращало конфликты ввода.

Тестирование и балансировка проводились итеративно: после реализации каждой ключевой системы проводилась серия тестовых партий, в ходе которых уточнялись параметры юнитов, корректировались эффекты карт и оптимизировалась пользовательская логика.

Заключение

В результате проделанной работы был создан полноценный прототип пошаговой стратегической игры, сочетающий элементы классических тактических игр с современными механиками. Удалось реализовать все ключевые системы: управление юнитами с уникальными характеристиками, пошаговый бой с ограниченным числом ходов, карточные модификаторы и мощные суперспособности. Архитектура игры получилась гибкой – при необходимости можно легко добавить новых персонажей, карты или даже изменить размер поля без переписывания основной логики.

Особую ценность представляет баланс между простотой и глубиной. Правила легко понять, но для победы требуется продуманная стратегия: нужно учитывать позиции юнитов, контролировать заряды суперударов, вовремя использовать карты и предугадывать действия противника. Это делает игру интересной как для новичков, так и для опытных игроков.

Цели, поставленные в начале выполнения работы были достигнуты. В будущем систему можно расширить – добавить новые типы юнитов, карт и мультиплеер, но уже сейчас она служит отличной основой для дальнейшего развития.

Приложение

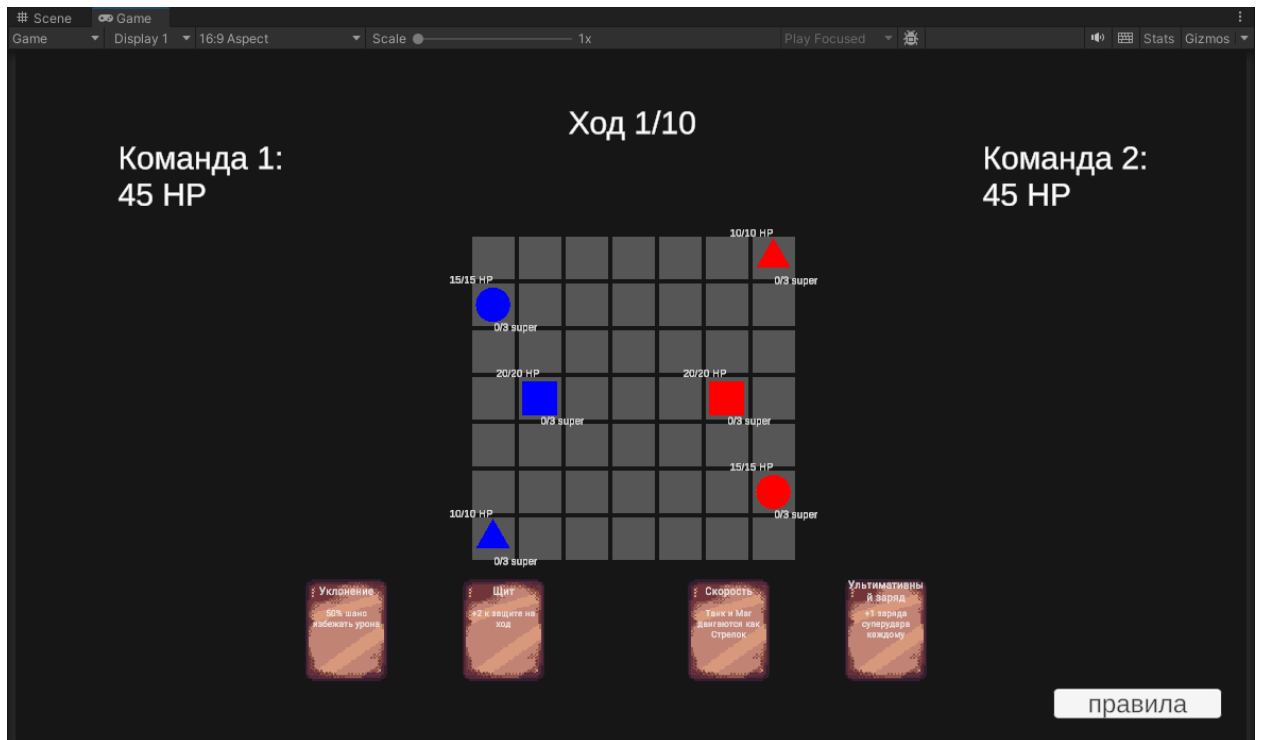
Приложение 1

Видео с демонстрацией геймплея: https://drive.google.com/file/d/19Vx9-nTXrRbg3ICU0xHZjPbL9YoTK7_I/view?usp=sharing

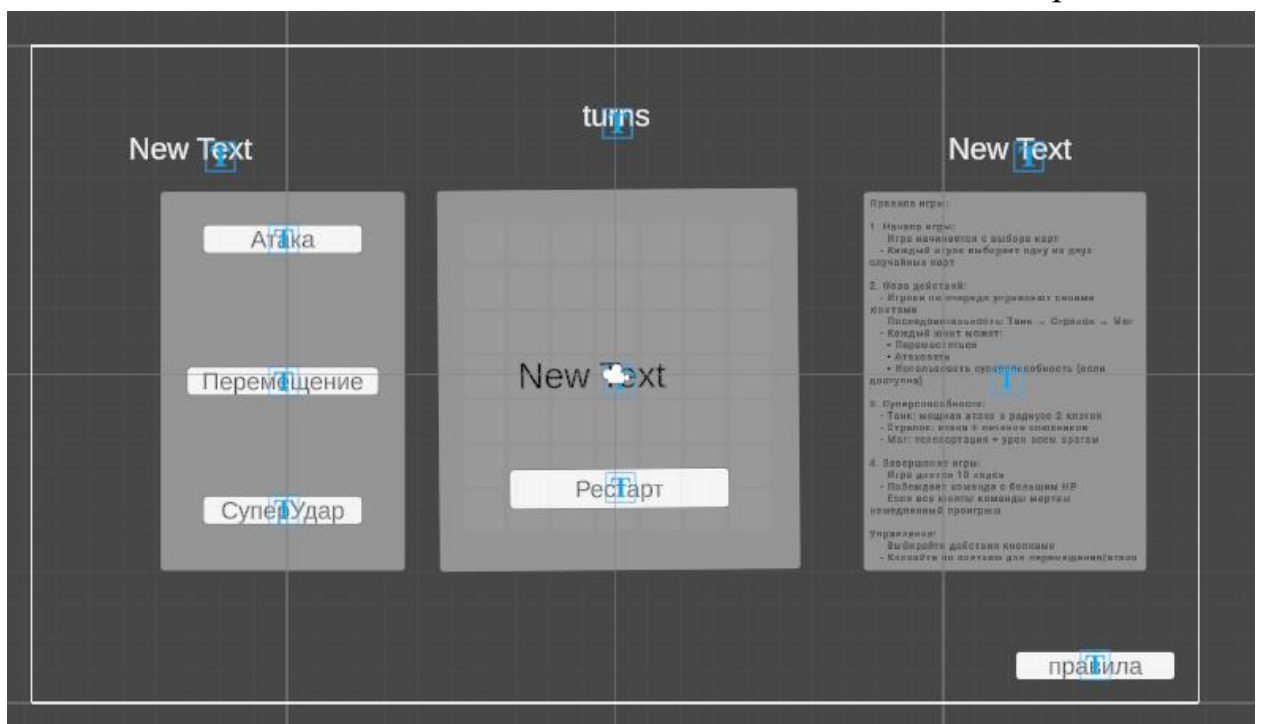
Приложение 2

Репозиторий проекта: <https://github.com/Mikhail-Skos/practice>

Приложение 3



Приложение 4



Приложение 5

