

Министерство образования Республики Беларусь  
Гомельский государственный технический  
Университет им. П.О.Сухого

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

направление специальности 1-40 01 01-12 «Информационные системы и  
технологии (в игровой индустрии)»

Доклад по дипломной работе

Выполнил:  
студент гр. ИТИ-41  
Менделев Д.А.

[слайд 1] Приветствие

Я, студент группы ИТИ-41, Менделев Дмитрий Александрович, хочу представить вам дипломную работу на тему: «3D игра в жанре выживания с адаптивной сложностью боев и открытой экономикой «Буйство прерий».

[слайд 2] Актуальность темы

Из результатов аналитического обзора стало понятно, что игровая индустрия это в целом достаточно актуальная часть быстрорастущего рынка цифровых технологий. Проведя анализ жанров, стало понятно, что жанр выживания является востребованным и актуальным на данный момент. Поспособствовало этому его открытость для игроков, множество возможностей и растущее сообщество. Помимо этого, сам жанр относительно прост в разработке и имеет малое количество конкурентов с уникальными механиками, что позволит выделить проект, полученный в ходе дипломной работы на фоне остальных.

[слайд 3] Использование *Unity* для разработки

Проведя анализ средств разработки, было выявлено две наиболее популярных среды для создания игр в открытом доступе – *Unity* и *Unreal Engine*. Оба движка являются достаточно мощными и предлагают большой набор инструментов для разработчика. Однако, несмотря на возможности *Unreal Engine*, *Unity* оказался более предпочтительным для разработки игры в жанре выживания благодаря большому сообществу и поддержке, множеству готовых компонентов, которые упростят разработку, его гибкости и легкости в настройке. Все это в совокупности делает *Unity* наиболее подходящей средой разработки для игры в жанре выживания.

[слайд 4] Архитектура клиента проекта

Клиентская часть использует большинство готовых возможностей и компонентов *Unity*, представленных в *WorldModule* и *UIModule* для создания

игрового мира и пользовательского интерфейса. В эти модули включены статические объекты, которые по большей части реализованы средствами *Unity*. Модуль *ClientModule* является реализацией *TCP* клиента. Выбор *TCP* обоснован тем, что данный протокол гарантирует доставку данных в нужном порядке, что в данном случае, предпочтительнее чем скорость доставки данных, так как используется для игрового рынка.

[слайд 5] Архитектура сервера

Серверная часть приложения имеет 3 основных модуля. 2 из них это контексты баз данных, которые используются для получения и обработки данных пользователей и игрового рынка. Модуль *ServerModule* является реализацией *TCP* сервера и служит для обработки запросов клиента.

[слайд 7] Клиент серверное взаимодействие

Так же была спроектирована архитектура клиент-серверного взаимодействия. В проекте был использован многопоточный *TCP* сервер для обработки нескольких клиентов одновременно. Сервер передает данные из контекстов баз данных клиентам, а клиент в свою очередь передает серверу ввод игрока.

[слайд 8] Архитектура игрового мира

Большая часть компонентов реализована средствами *Unity*, а некоторые статические объекты имеют собственный функционал для реализаций функций игрового мира, таких как добыча ресурсов, обработка входа в локации и тд. Для этого функционал был дописан, опираясь на стандартные компоненты *Unity*.

[слайд 9] Алгоритм адаптивной сложности

Алгоритм отслеживания начала боя и имеет 3 состояния: начало, сбор данных и подсчет. Во время боя каждый удар противника и игрока подсчитывается и считается в конце для определения результатов и обновления значений у противников.

[слайд 10] Состояния противников

В зависимости от типа появления, противник может начать с двух состояний: движение к точке или простой. В любом случае, как только игрок попадет в область зрения, противник начнет преследование и уйдет вступит в бой либо же вернется в первоначальное состояние. В результате боя, если противник останется жив, он вернется в первоначальное состояние и обновит значения из алгоритма.

[слайд 11] Уникальные механики

К уникальным механикам проекта стоит отнести многоступенчатую систему создания предметов, где для создания одного предмета необходимо создать цепочку предыдущих. Так же уникальной механикой является открытая экономика, где нету ограничения по ценам со стороны разработчика и игроки могут сами регулировать цены на предметы. Еще одной отличительной особенностью является адаптивная сложность боев, которая позволит игрокам с любым уровнем навыка достаточно легко войти в игровой процесс.

[слайд 12] Адаптивная сложность

Система адаптивной сложности боев в игре регулирует поведение противников во время боя и дает игроку преимущество в случае, если он не имеет достаточного навыка игры. Это обусловлено тем, что игра имеет сетевое взаимодействие и нельзя сделать отдельный уровень сложности для каждого игрока, а с таким подходом каждый игрок затратит одинаковое количество усилий для добычи ресурсов. Алгоритм регулирует шанс блока, шанс использование способности и уменьшает частоту пробивания в блок и не влияет на основные характеристики, такие как здоровье и урон.

[слайд 13] Игровая экономика и социальное взаимодействие

Открытая экономика является уникальной механикой для игр в жанре выживания. Такой подход позволит игрокам продавать и покупать ресурсы без регулирования цен со стороны разработчика. Это позволит каждому игроку сфокусироваться на определённой части игрового процесса, будь то добыча ресурсов или бой с противниками. Помимо этого, игроки могут получать игровую валюту множеством различных способов.

[слайд 14] Игровой цикл

Игровой цикл является бесконечным и не имеет сюжетной концовки, что типично для жанра выживания. Игрок должен собирать ресурсы и вести бой с противниками, чтобы добыть ресурсы и получить деньги. После чего, игрок может заняться созданием предметов либо же торговать с другими игроками, чтобы получить ресурсы, необходимые для обновления таймера и дальнейшей игры.

[слайд 15] Оптимизация с помощью *DLSS*

Как одно из современных средств оптимизации, в игру была встроена система *Deep Learning Super Sampling*, которая помогает игрокам, имеющим графические процессоры старше *NVidia GeForce RTX 20+* увеличить частоту кадров в игре. Прирост частоты кадров оказывается достаточно значительным, особенно в моментах, где нагрузка оказывается на графический процессор.

[слайд 16] Заключение

В ходе дипломной работы была разработана игра для длительных множественных сессий, покрывающая широкий пласт аудитории благодаря низкому порогу вхождения. Игра имеет возможности масштабирования и может быть дополнена новыми предметами, противниками и локациями. Игра имеет множество уникальных механик и особенностей, что делает ее привлекательно на фоне остальных. Алгоритмы адаптивной сложности боев и оптимизации с помощью *DLSS* прошли апробацию в научных журналах «Интернаука» и «Студенческий».

[слайд 17] Конец

Спасибо за внимание.