**Аннотация**

**Задание**

**Оглавление**

**Введение**

В современном мире компьютерные игры стали неотъемлемой частью развлечения миллионов людей по всему миру. Данное направление активно развивается и имеет множество перспективных направлений в медиаиндустрии. Времена, когда разработкой игр занимались только крупные компании и средства для их разработки были доступны только внутри их компаний ушли в прошлое. Технологии разработки доступны для всех и позволяют даже одному разработчику создать свою игру, не прибегая к чужой помощи. На данный момент большинство проектов разрабатываются при помощи игровых движков, которые включают в себя множество полезных функций, упрощающих разработку компьютерных игр. Однако, несмотря на большое количество многофункциональных инструментов, которые будут рассмотрены далее, от разработчика всё равно требуется очень большое количество труда и знаний, чтобы создать действительно хороший продукт.

Игровой движок («Game Engine») — это программное обеспечение, разработанное для упрощения и ускорения процесса создания компьютерных игр. Этот инструмент предоставляет разработчикам все необходимые компоненты и функции для проектирования, создания, редактирования и запуска игровых проектов.

Основные компоненты игрового движка включают в себя:

* графический движок («Graphics Engine»): отвечает за отображение графики, работу с «2D» и «3D» моделями, освещение, текстуры и другие визуальные аспекты игры;
* физический движок («Physics Engine»): обеспечивает моделирование физики в игре, такие как коллизии, гравитация, силы, движение объектов и взаимодействие между ними;
* искусственный интеллект («Artificial Intelligence»): позволяет создавать поведение неигровых персонажей, так называемых «Non-Player Character», управлять их действиями и решениями на основе заранее заданных алгоритмов;
* аудио-движок («Audio Engine»): отвечает за воспроизведение звуковых эффектов, музыки и звуковой атмосферы в игре;
* инструменты разработки («Development Tools»): включают в себя редакторы сцен, анимаций, скриптов, интерфейсов и другие инструменты для удобства работы разработчиков;
* система управления ресурсами («Resource Management System»): оптимизирует работу с ресурсами игры (текстуры, звуки, модели), управляя их загрузкой, выгрузкой и оптимизацией для улучшения производительности;
* скриптовые языки («Scripting Languages»): предоставляют средства для написания скриптов и программирования игровой логики.

Игровые движки бывают коммерческими и бесплатными с открытым исходным кодом. Они позволяют разработчикам сосредоточиться на креативном процессе создания игры, а не на написании с нуля всех необходимых компонентов. Кроме того, они часто предоставляют возможности многоплатформенной разработки, позволяя выпускать игры на различных устройствах, таких как компьютеры, консоли, мобильные устройства.

Также с помощью игрового движка разработчики могут значительно ускорить процесс создания игр, снизить затраты времени и ресурсов на разработку, а также получить доступ к мощным инструментам для реализации своих идей в интерактивной форме.

На сегодняшний день игровая индустрия развивается невероятно быстро, количество активных игроков и прибыль растет с каждым годом. Все это указывает на то, что разработка видеоигр является одним из перспективнейших направлений в сфере разработки программных продуктов.

Можно точно сказать, что разрабатываемая игра будет достаточно популярна среди выделенной целевой аудитории. Этому способствуют выбранные жанр и стилизация проекта. Жанр «бродилка» представляет собой простой и понятный, а также интересный игровой процесс. Обладая очень низким порогом вхождения, такие игры доступны очень широкому кругу людей, в том числе тем, кто играми как средством проведения собственно досуга мало интересуется.

Так же стоит помнить, что современное общество перенасыщено крупными проектами, требующие от игрока глубокого понимая механики игрового процесса. Для их прохождения требуется довольно большое количество времени. На их фоне игра с простым и понятным геймплеем будет смотреться более предпочтительно среди людей, которым сложно уделять достаточно много времени.

На основе вышеперечисленного легко выделяется цель проекта.

Цель курсового проекта заключается в создании компьютерной игры жанра «2D - бродилка» с помощью игрового движка «Unity».

Для достижения поставленной цели необходимо выполнить следующие задачи:

* проектирование игры: определение концепции игры, механик геймплея и создание игрового мира;
* реализация игровых механик: создание управления персонажем, взаимодействия с окружением и других игровых элементов;
* визуальное оформление: разработка графического стиля, создание анимаций и интерфейса пользователя;
* тестирование и оптимизация: проверка функционала, исправление ошибок и оптимизация производительности игры.

Практическая значимость полученных результатов заключается в том, что полученные результаты имеют важное практическое значение в контексте разработки игр. Созданная «бродилка» не только представляет собой полноценную игру, но и отражает ключевые аспекты работы над игровым проектом, начиная от идеи и заканчивая рабочей версией. Полученный опыт и навыки играют важную роль в дальнейшей разработке игры и программировании.

Для реализации проекта будут задействованы ресурсы, перечисленные ниже:

* игровой движок «Unity»: интегрированная среда разработки для создания игр.
* графические инструменты: «Photoshop» и «Aseprite»;
* инструменты программирования: «Visual Studio Code», для написания кода;

Перспективы расширения функционала приложения: полученный результат - лишь начальная точка для дальнейшего улучшения и расширения функционала приложения. Возможные направления развития включают в себя добавление новых уровней, введение дополнительных игровых механик, усовершенствование графики и адаптацию для различных платформ.

**Анализ предметной области**

Разработка компьютерных игр — процесс создания компьютерных игр (видеоигр). Разработкой видеоигр занимается разработчик, который может быть представлен как одним человеком, так и фирмой. Обычно крупномасштабные коммерческие игры разрабатываются командами разработчиков в пределах компании, специализирующейся на играх для персонального компьютера или консолей. Как правило, разработку финансирует другая, более крупная компания-издатель, которая по окончанию разработки занимается изданием игры и связанными с ним тратами. Реже компании-издатели могут содержать внутренние команды разработчиков, или же компания-разработчик может разрабатывать игры за свой счет и распространять их без участия издателей, например, средствами цифровой дистрибуции (инди-игры).

Разработка наиболее крупнобюджетных игр, таких как «AAA-игры», может стоить десятки миллионов долларов США, причем в течение последних десятилетий эти бюджеты непрерывно росли, как и численность команд разработчиков и сроки разработки.

Благодаря развитию рынка инди-игр, многие разработчики компьютерных игр получили возможность работать над своими игровыми проектами без финансовых и юридических обязательств перед компаниями-издателями.

Инди-игры (англ. «Indie games», от англ. «independent video games» — «независимые компьютерные игры») — это компьютерные игры, созданные отдельными разработчиками или небольшими коллективами без финансовой поддержки издателя компьютерных игр. Распространение осуществляется посредством каналов цифровой дистрибуции, то есть сейчас существует множество платформ, на которые независимый разработчик может выкладывать свои игры, самые известные из них: «Steam», «Epic Games Store», «Itch.io» и «VK Play». Масштаб явлений, связанных с инди-играми, ощутимо возрастает со второй половины 2000-х годов, в основном ввиду развития новых способов онлайн-дистрибуции и средств разработки.

Разработка инди игр на «Unity» представляет собой процесс создания игровых приложений в двухмерном или трехмерном формате при помощи программной среды «Unity». Развитие данной области происходит в контексте постоянного усовершенствования инструментов и технологий, направленных на облегчение и ускорение процесса разработки.

Сущности и явления данной области разработки перечислены ниже:

* интегрированная среда разработки («IDE»): «Unity» предоставляет разработчикам универсальную среду с графическим интерфейсом для создания, редактирования и программирования игровых элементов;
* графический движок и рендеринг: «Unity» обладает инструментами для создания и управления «2D» графикой, позволяющими реализовывать анимацию, работать с текстурами, спрайтами и анимированными объектами;
* программирование и скриптинг: использование языка программирования «C#» для создания игровой логики, взаимодействия объектов, создания и редактирования скриптов, определяющих поведение элементов игры;
* управление ресурсами: эффективное управление звуками, изображениями, анимациями, и другими ресурсами игры с помощью специальных инструментов «Unity».

Отношения и процессы, определяющие сферу деятельности данной области:

* цикл разработки игры: включает этапы планирования, дизайна, программирования, тестирования и выпуска игры, каждый из которых требует четкой организации и согласования усилий всей команды разработчиков;
* командная работа и коммуникация: в разработке игр важна эффективная командная работа, включая взаимодействие между программистами, дизайнерами, художниками, звукорежиссерами и другими специалистами;
* техническая поддержка и обновления: после выпуска игры важно обеспечивать техническую поддержку, исправлять ошибки, так называемые баги, и предоставлять обновления для улучшения игрового опыта игроков;
* монетизация и распространение: определение стратегий монетизации игры (платная/бесплатная с возможностью покупок внутри игры), выбор платформ для распространения («Steam», мобильные платформы и другие).

Разработка инди игр на «Unity» представляет собой многоаспектный процесс, требующий особый подход, глубоких знаний технических аспектов игровой разработки и умения эффективно управлять проектом от начала до выпуска готовой игры на рынок.

Разрабатываемый проект относится к инди-разработке и развивается только за счет средств разработчиков. Благодаря развитию программного обеспечения разработки компьютерных игр, команде разработчиков не требуется тратить несколько лет на разработку игрового движка. Это позволяет сразу же приступать к непосредственной работе над игровым проектом и значительно сокращает время его разработки.

**Проектирование приложения**

Проектируемое приложение «Wander 2D» – это игра в жанре «бродилка» и расположением камеры «top - down», в которой главный герой исследует игровой мир, взаимодействуя с различными предметами. Данная игра предназначена для широкого круга пользователей с простыми механиками и интуитивно понятным коротким игровым процессом.  
Для успешной реализации данного продукта, нужно сформулировать ряд требований.

Требования по геймплейной составляющей – игрок появляется на первом уровне, чтобы попасть на следующий, ему нужно найти некоторое количество монет. Чтобы получить те самые монеты, игрок должен изучать различные локации, присутствующие на карте. Изучая локации, игрок должен решать ряд простых задач такие как передвинуть предмет, принести предмет и так далее. Выполнение данных задач поощряется выдачей монет, которые необходимы игроку для успешного прохождения игры.

Игра должна иметь двумерную графику и вид сверху.

В игре должны присутствовать две локации: лес и замок. Каждая локация уникальна, то есть локации должны отличаться дизайном и загадками.

Загадки должны быть достаточно простыми, например требуется принести какую – либо вещь или переместить нужный объект в необходимое место для получения монеты.

Интерфейс игры состоит из главного меню и игровой панели. В главном меню игрок может выбрать уровень, сбросить свой прогресс или выйти из игры. Игровая панель находится в верхней части экрана и состоит из двух компонентов: изображение монетки, которые нужно собирать и непосредственно их количество.

Управление осуществляется при помощи клавиш, представленных ниже:

* «w» - вверх;
* «a» - влево;
* «s» - вправо;
* «d» - вниз.

В ходе анализа предметной области и на основе необходимых требований были определены функциональные и нефункциональные требования к игровому приложению и возможности, которые должны предоставляться пользователю приложения.

Функциональные требования определяют функциональность программного обеспечения, то есть описывают, какое поведение должна предоставлять разрабатываемая система. Разрабатываемая система должна удовлетворять следующим функциональным требованиям:

* в приложении должно присутствовать меню для более удобного начала игры;
* в меню должна присутствовать кнопка для выхода из игры;
* пользователь должен иметь возможность выбора уровня;
* пользователь должен иметь возможность сбрасывать свой прогресс;
* пользователь должен иметь возможность взаимодействовать с окружающими объектами;
* пользователь должен иметь возможность подбирать монеты;
* пользователь должен иметь возможность подбирать предметы;
* пользователь должен иметь возможность видеть текущее число подобранных монет.

Нефункциональные требования описывают свойства и ограничения, накладываемые на систему. Для реализации приложения были выявлены следующие требования:

* приложение должно быть создано на игровом движке «Unity» с использованием языка программирования «C#»;
* обеспечение стабильной работы приложения на настольных компьютерах с операционной системой семейства «Windows», версии не ниже «Windows 7»;
* создание удобного пользовательского интерфейса для облегчения взаимодействия с игрой;
* возможность дальнейшего расширения функционала и добавления новых элементов.

В ходе анализа требований была разработана «UML-диаграмма» вариантов использования компьютерной игры, с которой можно ознакомиться в «Приложение А».

**Разработка программного обеспечения**

**Описание технологического стека разработки**

Стек разработки — это набор технологий, инструментов, языков программирования, фреймворков и других компонентов, используемых при создании программного обеспечения или приложений. Этот стек определяет инфраструктуру, которая будет использоваться для разработки, тестирования, внедрения и поддержки проекта.

Основные компоненты стека разработки могут включать в себя:

* языки программирования;
* фреймворки и библиотеки;
* базы данных;
* среда разработки и инструменты управления версиями;
* серверные и клиентские технологии.

Стек разработки зависит от типа проекта, требований к производительности, предпочтений команды разработчиков и других факторов. Правильно выбранный стек может ускорить процесс разработки, повысить производительность и обеспечить лучшую масштабируемость и надежность приложения.

При разработке «2D» игры будет использован стек, представленный ниже:

* язык программирования «C#»;
* игровой движок «Unity Engine».

Язык программирования «C#» - является основным языком программирования в «Unity». Данный язык используется для написания скриптов — программы, написанные на языке программирования «C#», которые определяют поведение и действия объектов в игре. Они позволяют управлять движением, взаимодействием и внешним видом объектов. С помощью скриптов можно создавать сложные анимации, реагировать на пользовательский ввод и управлять игровой логикой.

«Unity Engine» — это кроссплатформенный игровой движок, который широко используется для разработки компьютерных игр, виртуальной и дополненной реальности, тренировочных симуляторов, а также для создания интерактивных «2D» и «3D» приложений.

Обоснование выбора данного стека заключается в том, что «Unity Engine» предоставляет мощные инструменты и гибкие возможности для создания «2D» игр, обеспечивая удобство работы с графикой, анимациями, физикой и аудио. Это позволяет сосредоточиться на самом процессе разработки игры, ускоряя время разработки. Также данный движок обладает богатым комьюнити, то есть Unity имеет огромное сообщество разработчиков, готовых поддержать и помочь в решении проблем. В свою очередь, язык программирования «C#» обладает мощными возможностями объектно-ориентированного программирования, и интеграция с «Unity API» обеспечивают высокую производительность и эффективность в разработке.

Выбранный стек разработки обладает рядом преимуществ, такими как продуктивность разработки, масштабируемость и портативность и поддержка сообщества.

Продуктивность разработки заключается в том, что «C#» обеспечивает удобство при программировании игровой логики, а «Unity Engine» предоставляет интуитивно понятный интерфейс и множество инструментов для ускорения разработки.

Масштабируемость «Unity» позволяет разрабатывать игры для различных платформ, таких как персональные компьютеры, мобильные устройства, консоли и веб. Это обеспечивает большую аудиторию и возможность масштабирования проекта.

Поддержка сообщества выражается большим сообществом разработчиков «Unity», которые активно обменивается опытом, предоставляет обучающие материалы и помощь, что существенно облегчает разработку и устранение возможных проблем.

Использование игрового движка «Unity» с языком программирования «C#» обеспечивает необходимый инструментарий для создания «2D» игр, упрощает процесс разработки и обеспечивает возможность достижения широкой аудитории игроков.

**Описание алгоритма работы**

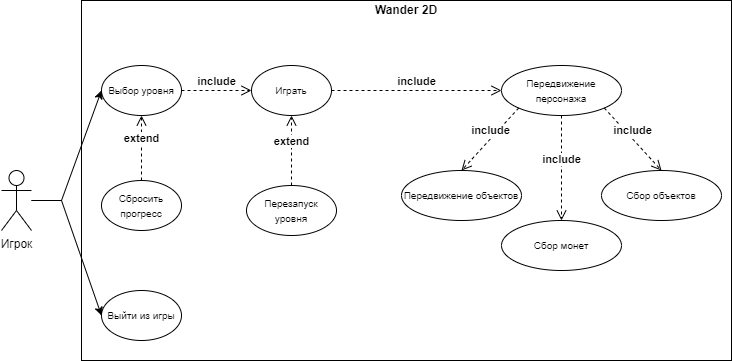
**Описание интерфейса пользователя**

**Тестирование приложения**

**Заключение**

**Список литературы**

**Диаграмма вариантов использования**

****

**Диаграмма деятельности**

**Диаграмма классов**