фМинистерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С. Галущака»

Разработка логичеческой головоломки с элементами платформера

Отчёт по производственной практике

ПМ. 02 Осуществление интеграции программных модулей

ПМ. 04 Сопровождение и обслуживание программного обеспечения компьютерных систем

НАТКиГ.421000.010.000

Разработал:

студент группы ПР-21.106

Классман Р.В.

2024

**Содержание**

[1 Описание предметной области, исходные данные 3](#_Toc163683301)

[2 Проектирование ИС 7](#_Toc163683302)

[3 Создание приложения 8](#_Toc163683303)

[4 Анализ аналогов и прототипов 12](#_Toc163683304)

[5 Подготовка тестовых сценариев 14](#_Toc163683305)

[Вывод 15](#_Toc163683306)

1 Описание предметной области, исходные данные

Логические игры, объединяющие элементы платформера и головоломок, представляют собой захватывающий жанр, который вызывает интеллектуальные вызовы и предлагает игрокам волнующее приключение. Этот уникальный поджанр в мире видеоигр сочетает в себе не только управление персонажем и платформенные элементы, но и взаимодействие с физическими объектами, такими как загадочные кубы.

В этих играх игроки вынуждены сочетать мастерство в управлении персонажем с разгадыванием головоломок, где физические объекты, такие как кубы, играют важную роль.

Игроки будут сталкиваться с задачами, где взаимодействие с физическими, их перемещение, изменение положения будут необходимы для продвижения вперед и решения головоломок. Этот аспект добавляет в игры головоломки-платформеры дополнительный уровень сложности и делает геймплей более увлекательным и многогранным. Так как «Square Cube» объединяет в себе головоломки и платформеры, по отдельности рассмотрим характерные черты для обоих жанров.

Платформеры – это жанр видеоигр, фокусирующийся на управлении персонажем, который перемещается по различным платформам и преодолевает препятствия. Этот жанр имеет следующие характеристики:

1. Активное управление персонажем: Основным элементом геймплея платформеров является управление персонажем, включающее в себя бег, прыжки, и другие действия. Игроки должны быть мастерами в управлении своим персонажем, чтобы успешно пройти уровни.

2. Платформы и препятствия. В платформерах игроки перемещаются по различным платформам и сталкиваются с разнообразными препятствиями, такими как ямы, пропасти и враги. Они должны использовать свои навыки управления, чтобы избегать опасностей и преодолевать преграды.

3. Рефлексы и координация. платформеры требуют быстрых рефлексов и хорошей координации движений, чтобы персонаж мог выполнять точные и сложные действия.

4. Прогрессивная сложность. Игры этого жанра часто увеличивают сложность по мере продвижения, предоставляя игрокам новые вызовы и возможности для развития навыков.

Головоломки – это жанр видеоигр, где главным элементом является разгадывание разнообразных головоломок и задач. Этот жанр характеризуется следующим:

1. Интеллектуальное развитие. Головоломки требуют от игроков логического мышления и креативных решений. Они сталкиваются с задачами, которые не всегда очевидны, и должны использовать свой ум, чтобы найти правильные ответы.

2. Разнообразие задач. В жанре головоломок представлено множество различных типов задач, включая логические головоломки, кроссворды, физические головоломки и многие другие. Это придает играм головоломок разнообразие и уникальность.

3. Удовлетворение от решения. Главная награда в головоломках – это удовлетворение от разгадывания задачи. Каждая правильная головоломка приносит чувство успеха и мотивирует игрока продолжать разгадывать следующие.

4. Тренировка мозга. Головоломки предоставляют отличную возможность для тренировки мозга и развития когнитивных навыков, таких как логическое мышление, планирование и решение проблем.

В жанре логических игр головоломок-платформеров, как «Braid», эти два элемента – активное управление персонажем и разгадывание головоломок, объединяются, создавая захватывающий и интеллектуальный игровой опыт, который увлечет игроков в мир креативных решений и приключений.

В современной игровой индустрии, где каждый день появляются новые тайтлы и разнообразные игровые жанры, можно заметить, что игры, объединяющие элементы платформера и головоломок, и при этом акцентирующие внимание на развитии навыков планирования, логического мышления и реакции, являются настоящей редкостью.

Даже при том, что платформеры и головоломки сами по себе весьма популярны и имеют свою аудиторию, игры, способные гармонично сочетать эти два элемента в одном проекте, остаются на рынке в ограниченном количестве. Этот жанровый нишевый дефицит оставляет множество игроков, желающих погрузиться в увлекательный и интеллектуально насыщенный игровой опыт, с ограниченным выбором.

Сегодня, когда игры становятся не только развлечением, но и средством развития критического мышления, способности к решению задач и аналитических навыков, есть острая необходимость в создании более многочисленных и разнообразных игр данного жанра. Это позволит удовлетворить запросы и ожидания игроков, а также предоставит им возможность развивать свои умственные способности во время игры в захватывающие и увлекательные приключения.

Этот дефицит в качественных логических играх-платформерах оставляет игроков, желающих развивать свои креативные и аналитические способности, в поиске новых и увлекательных игровых опытов. Таким образом, существует потребность в разработке и предоставлении более многочисленных и разнообразных игр в данном жанре, чтобы удовлетворить запросы широкой аудитории игроков.

«Square Cube» будет предоставлять собой захватывающий платформер предоставляя игрокам возможность развивать свои умственные способности в интересном игровом контексте. В «Square Cube» цель игры будет заключатся в том, чтобы с ограниченным количеством действий добраться до выхода из закрытой комнаты, решая сложные головоломки и используя уникальную механику телепортации.

Игра будет постепенно усложняется с добавлением новых механик и элементов, что будет увеличивать количество возможных комбинаций решения головоломок и так же будет повышать сложность уровней. «Square Cube» ориентирована на аудиторию от 14 до 30 лет, предоставляя им интеллектуальное развлечение и возможность развивать навыки планирования, логического мышления и реакции.

Эта игра будет доступна на ПК с операционными системами Windows, Linux и macOS, обеспечивая широкую доступность. В игре, будет минималистичная графика, напоминающая игру «Thomas Was Alone», «Square Cube» предоставляет игрокам стильную визуальную эстетику и увлекательное погружение в мир логических головоломок и платформеров. Таким образом, «Square Cube» решает проблему недостатка качественных логических игр, предлагая игрокам уникальный игровой опыт и способствуя развитию их интеллектуальных навыков.

2 Проектирование ИС

Анализ требований: Изучение требований к игре, включая функциональные и нефункциональные требования, а также предпочтения целевой аудитории.

Определение концепции игры: Определение основной концепции, темы, цели и механики игры, которые будут определять ее структуру и функциональность.

Проектирование игрового движка: Разработка игрового движка или выбор существующего, который будет обеспечивать основные функциональные возможности игры, такие как графика, физика, звук и управление персонажами.

Разработка игровых механик и интерфейса: Создание игровых механик (правил игры) и интерфейса пользователя, который позволит игрокам взаимодействовать с игровым миром.

Разработка игровых контента: Создание игровых уровней, персонажей, объектов, сценариев и других элементов контента, которые составляют игровой опыт.

Тестирование и отладка: Проведение тестирования игры на различных этапах разработки для выявления ошибок, ошибок и проблем с производительностью, а также их устранения.

Внедрение и выпуск: Запуск игры на выбранных платформах (например, компьютеры, консоли, мобильные устройства) и распространение среди пользователей.

Поддержка и обновление: Поддержка и обновление игры после выпуска, включая патчи, исправления ошибок, добавление нового контента и т. д.

3 Создание приложения

Класс, описанный в Листинге 1, который наследуется от MonoBehaviourPunCallbacks, что предполагает использование Photon Unity Networking (PUN) для соединения с удаленным сервером. В методе Start() устанавливается никнейм игрока, включается автоматическая синхронизация сцены и запускается процесс подключения к серверу с помощью вызова PhotonNetwork.ConnectUsingSettings().

Листинг 1 – соединение с сервером

**public** **class** **ConnectionToServer** : MonoBehaviourPunCallbacks

{

[SerializeField] **private** UIServer \_ui;

**private** **void** **Start**()

{

PhotonNetwork.NickName = "Player";

PhotonNetwork.AutomaticallySyncScene = **true**;

PhotonNetwork.ConnectUsingSettings();

}

**public** IEnumerator **Connection**()

{

**yield** **return** **new** **WaitUntil**(GetConnectionToServer);

print("IsConnetcion");

}

**private** **bool** **GetConnectionToServer**()

{

**return** PhotonNetwork.IsConnected;

}

**public** IEnumerator **Disconection**(System.Action actionEvent)

{

PhotonNetwork.Disconnect();

**while** (PhotonNetwork.IsConnected)

{

**yield** **return** **null**;

}

PhotonNetwork.OfflineMode = **true**;

actionEvent?.Invoke();

}

**public** **override** **void** **OnConnectedToMaster**()

{

\_ui.ActiveCoopButton();

}

}

Метод Connection() представляет корутин, которая ждет, пока произойдет подключение к серверу, с помощью WaitUntil, и после этого выводит сообщение "IsConnetcion".

Метод Disconection() также представляет корутин, который отключает пользователя от сервера, ожидая, пока подключение не разорвется. После этого устанавливается оффлайн режим, и вызывается указанное событие.

В методе OnConnectedToMaster() выполняется действие активации кнопки кооперативного режима, предположительно, используя связанный с классом объект \_ui.

Класс WindowConnectORCreateRoom описанный в Листинге 2 предназначен для управления процессом подключения игроков к существующим комнатам или создания новых комнат в многопользовательском приложении на с использованием Photon Unity Networking (PUN).

Листинг 2 – код для создания или подключения к комнате

[SerializeField] **private** TMP\_InputField \_idRoom;

[SerializeField] **private** WindowError \_windowError;

[SerializeField] **private** GameObject \_elementsWindows;

**public** **void** **JoinRoom**()

{

PhotonNetwork.JoinRoom(\_idRoom.text);

}

**public** **void** **CreateRoom**()

{

PhotonNetwork.CreateRoom(CreateIdRoom(), **new** Photon.Realtime.RoomOptions{ MaxPlayers = **2**});

}

**private** **string** **CreateIdRoom**()

{

**string** id = "";

**for** (**int** i = **0**; i < **4**; i++)

{

id += Random.Range(**0**, **9**);

}

**return** id;

}

**public** **override** **void** **OnJoinRoomFailed**(**short** returnCode, **string** message)

{

\_windowError.Active("комнаты с таким id не существует");

}

**public** **override** **void** **OnJoinedRoom**()

{

PhotonNetwork.LoadLevel(**1**);

}

**public** **void** **CloseWindows**()

{

\_elementsWindows.SetActive(**false**);

}

Методы JoinRoom() и CreateRoom() обеспечивают функциональность присоединения к существующей комнате или создания новой комнаты соответственно.

Методы OnJoinRoomFailed() и OnJoinedRoom() отвечают за обработку событий неудачного присоединения к комнате и успешного присоединения к комнате. В случае неудачи выводится сообщение об ошибке, а при успешном присоединении происходит переход на другой уровень игры.

Метод CreateIdRoom() генерирует случайный идентификатор для новой комнаты.

Метод CloseWindows() используется для закрытия окна после завершения действий.

Класс GameUISeriver, показанный в Листинге 3 является наследником класса GameUi и предназначен для управления пользовательским интерфейсом (UI) и загрузки сцен в многопользовательской игре с использованием Photon Unity Networking (PUN)..

Листинг 3 – модуль mixer

**public** **class** **GameUISeriver** : GameUi

{

**private** PhotonView \_photonView;

**public** **void** **InitPhoton**(PhotonView view)

{

\_photonView = view;

}

**public** **override** **void** **Restart**()

{

RestartPhoton();

}

**public** **void** **RestartPhoton**()

{

PhotonNetwork.LoadLevel(GetIndexCurrentScene());

\_photonView.RPC(nameof(LoadSceneRPC), RpcTarget.Others, GetIndexCurrentScene());

print("Server restart");

}

**public** **override** **void** **NextLevel**()

{

PhotonNetwork.LoadLevel(GetIndexCurrentScene()+**1**);

}

[PunRPC]

**public** **void** **LoadSceneRPC**(**int** idScene)

{

// Загрузка новой сцены на всех клиентах

PhotonNetwork.LoadLevel(idScene);

}

}

Метод InitPhoton() принимает объект PhotonView, который инициализирует переменную \_photonView, предположительно для взаимодействия с сетевыми объектами.

Метод Restart() переопределяет абстрактный метод из родительского класса GameUi и вызывает метод RestartPhoton() для перезапуска игры на сервере и на всех клиентах.

Метод RestartPhoton() загружает текущую сцену с использованием PhotonNetwork.LoadLevel(), а также отправляет RPC (Remote Procedure Call) вызов сцены другим клиентам с помощью \_photonView.RPC().

Метод NextLevel() загружает следующую сцену в игре.

Атрибут [PunRPC] используется для определения метода LoadSceneRPC(), который загружает указанную сцену на всех клиентах через PhotonNetwork.LoadLevel().

4 Анализ аналогов и прототипов

На основе выделенного функционала было рассмотрено 2 аналога Braid и Portal 2.

Portal 2 - это культовая головоломка-платформер, разработанная Valve Corporation. В игре игроки используют порталы для перемещения и решения головоломок, а также наслаждаются увлекательным сюжетом и харизматичными персонажами. В игре есть как одиночный так и кооперативный режимы.

Одиночный режим в "Portal 2" предлагает игроку использовать основную механику игры - порталы - для перемещения по уровням и решения головоломок. Создавая два портала - входной и выходной - игрок может преодолевать препятствия и находить пути к выходу. Головоломки в одиночном режиме разнообразны и требуют от игрока логического мышления, а также использования механики порталов в сочетании с различными объектами, переключателями, лазерами и гелями, что добавляет разнообразие в игровой процесс. Кроме того, в одиночном режиме игрок следует за сюжетом игры, взаимодействует с различными персонажами и исследует разнообразные локации, что добавляет в игру элементы приключения и насыщенности.

Кооперативный режим в "Portal 2" предоставляет уникальный опыт совместного прохождения уровней и решения головоломок. Игроки совместно решают головоломки, используя механику порталов и требуя не только сотрудничества, но и хорошей коммуникации, чтобы успешно преодолеть препятствия. Каждый игрок в кооперативном режиме имеет свою портальную пушку, что добавляет дополнительные сложности и возможности для совместного решения головоломок. Кроме того, кооперативный режим предлагает уникальные моменты веселья и взаимодействия между игроками, что делает процесс решения головоломок более интересным и разнообразным.

Braid - это уникальная головоломка-платформер. Игра предлагает увлекательное путешествие по миру, где игроку предстоит решать сложные головоломки и следовать за захватывающим сюжетом. Основной механикой "Braid" является перемотка времени, позволяющая игроку управлять временем вперед и назад. Это позволяет игроку исправлять ошибки, решать сложные задачи и достигать ранее недоступных мест на уровнях. Кроме того, игрок сталкивается с разнообразными головоломками, которые требуют от него не только логического мышления, но и мастерства в применении механики перемотки времени. Все это создает уникальный и захватывающий игровой опыт, который привлекает игроков своей глубиной, креативностью и интеллектуальными вызовами.

5 Подготовка тестовых сценариев

В процессе разработки логической головоломки с элементами платформера было необходимо провести ряд тестовых сценариев, чтобы убедиться в корректной работе функциональности и соответствии требованиям игры. Ниже, в таблице 1, приведены примеры тестовых сценариев для различных компонентов программного продукта:

Таблица 1 – Тестовые сценарии

|  |  |  |
| --- | --- | --- |
| **Ввод** | **Ожидаемый результат** | **Фактический результат** |
| Подключение к комнате по id существующей комнаты | Загрузка соответствующей комнаты и синхронизация данных | Загрузка соответствующей комнаты и синхронизация данных |
| Создание новой комнаты | Запуск стартовой сцены ожидания игрока | Запуск стартовой сцены ожидания игрока |
| Выход хоста из комнаты | Передача роли хоста другому клиенту в комнате | Передача роли хоста другому клиенту в комнате |
| Движения персонажа | Синхронизация персонажа между клиентами | Синхронизация персонажей между клиентами |
| Перезапуск сцены | Перезапуск сцены у всех клиентов в комнате | Перезапуск сцены у всех клиентов в комнате |
| Ввод неверного id комнаты при подключении | Окно с предупреждением о том, что комнаты с таким id не существует | Окно с предупреждением о том, что комнаты с таким id не существует |

Вывод

Развитие кооперативных игр в жанре головоломок-платформеров открывает перед игровой индустрией новые возможности для создания захватывающих и интеллектуально насыщенных игровых опытов. Этот жанр предоставляет игрокам возможность не только развивать логическое мышление, но и улучшать коммуникационные и командные навыки, работая вместе с друзьями или другими игроками для решения сложных головоломок и преодоления препятствий.

Кооперативные игры в жанре головоломок-платформеров стимулируют воображение и творческое мышление, предоставляя игрокам возможность искать нестандартные решения и использовать фантазию для преодоления вызовов игрового мира. Благодаря разнообразию задач и уникальным игровым механикам игроки получают широкие возможности для тренировки ума и развития навыков в различных областях.

В современном мире, где игры становятся не только средством развлечения, но и способом развития критического мышления и аналитических навыков, кооперативные игры в жанре головоломок-платформеров играют важную роль. Они предоставляют игрокам возможность не только провести время с удовольствием, но и улучшить свои умственные способности во время игры в захватывающие и увлекательные приключения.

Таким образом, кооперативные игры в жанре головоломок-платформеров представляют собой уникальный игровой опыт, способствующий развитию различных навыков и стимулирующий интеллектуальное развитие игроков.