

1 ОБЗОР ЛИТЕРАТУРЫ

1.1 Обзор существующих аналогов игр

В данном разделе рассмотрим ярких представителей жанра ритм игр и игр использующих алгоритмы для анализа аудио сигналов.

1.1.1 Guitar Hero

Одним из известнейших примеров жанра ритм игр является серия игр Guitar Hero (рисунок 1.1).



Рисунок 1.1 – Ритм игра «Guitar Hero»

Первая игра серии вышла в 2005 году и известна тем, что управление в игре осуществлялось посредством специального контроллера в виде уменьшенной копии гитары. В качестве примера, первая часть серии включала в себя 47 композиций. Данная серия игр стала основой для множества подобного рода игр, в последующем занявших крупную долю рынка мобильных игр.

Основной целью игры являлось своевременное нажатие необходимых кнопок на специализированном контроллере (рисунок 1.2). Все кнопки контроллера были окрашены 5 различных цветов, означающих определённый лад. Кроме того на корпусе имела педаль, отвечающая за активацию струн. Успешным считалось нажатие на данную педаль, с одновременно нажатыми кнопками ладов, отображаемых на экране игрока. Нажатие было необходимо производить именно в тот момент, когда окружность некоторого цвета, доходила до находящихся зон внизу экрана. По итогу успешных нажатий, игроку начислялись очки, влияющие на рейтинг данной попытки прохождения.

Основной целью игры являлось своевременное нажатие необходимых кнопок на специализированном контроллере (рисунок 1.2). Все кнопки контроллера были окрашены 5 различных цветов, означающих определённый лад. Кроме того на корпусе имелась педаль, отвечающая за активацию струн. Успешным считалось нажатие на данную педаль, с одновременно нажатыми кнопками ладов, отображаемых на экране игрока. Нажатие было необходимо производить именно в тот момент, когда окружность некоторого цвета, доходила до находящихся зон внизу экрана. По итогу успешных нажатий, игроку начислялись очки, влияющие на рейтинг данной попытки прохождения.



Рисунок 1.2 – Контроллер для Guitar Hero

При разработке данного продукта планировалось использовать лишь определенное количество лицензированных треков. В связи с этим, при создании набора проигрываемых нот, разработчики не использовали алгоритмы для анализа аудиофайлов. Вместо этого акцент был сделан на ручное воссоздание гитарных аккордов под специализированный контроллер, для создания ощущения игры на настоящей гитаре. Такое решение могло быть так же обосновано финансовыми возможностями компании и ограничениями технологий того времени.

Данный подход является более выгодным и точным по сравнению с использованием алгоритмов, при условии, что количество аудиофайлов не будет превышать некоторый порог. В своё время порог определяется штатом и финансовыми возможностями компании.

Минусом данного подхода можно выделить сложность к масштабируемости продукта в будущем. Алгоритмы для частотного анализа аудиофайлов позволили бы в дальнейшем сэкономить на разработке данного проекта, а так же удешевить выпуск дополнительного контента.

1.1.2 Osu!

«Osu!» – это бесплатная ритм игра, в которой имеется четыре официальных игровых режима, отличающихся игровыми механиками (рисунок 1.3). Управление в игре осуществляется посредством различных

периферийных устройств: компьютерной мыши, клавиатуры, графического планшета, сенсорного устройства.

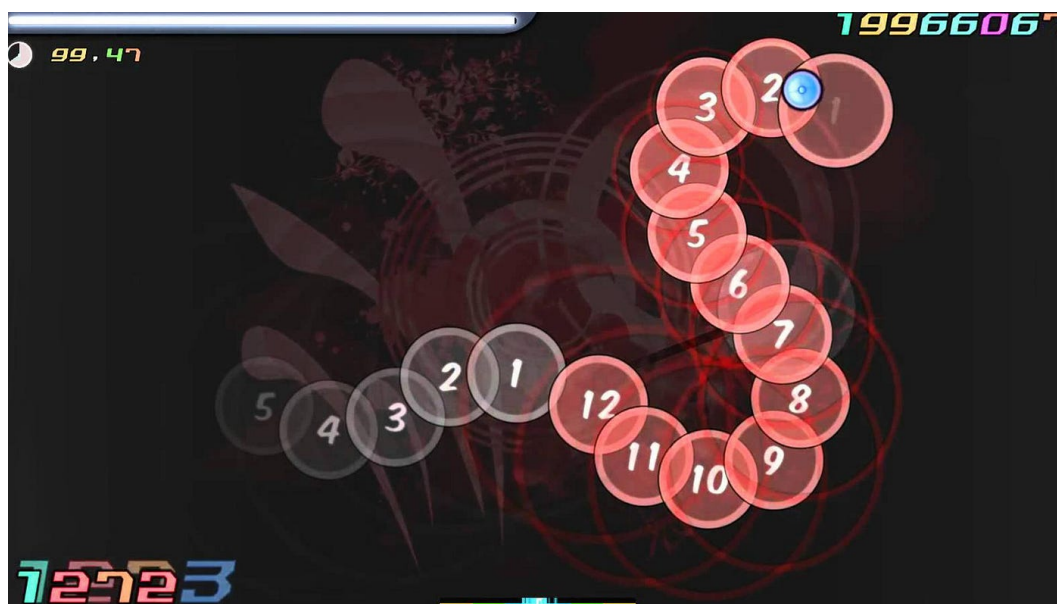


Рисунок 1.3 – Ритм игра «Osu!»

Отличительной особенностью данного проекта является то, что «Osu!» является платформой, позволяющей игрокам добавлять свои композиции, предоставляя игрокам инструменты для добавления собственного фона, композиций, настройки времени появления, времени задержек и позиций интерактивных объектов.

При установке данного продукта, пользователь обнаружит лишь пару тестовых композиций. Это связано с тем, что библиотека доступных треков хранится на официальном сайте продукта и обладает открытым доступом для добавления и скачивания, готовых к игре композиций (рисунок 1.4). Данные композиции хранятся в специальных файлах формата, хранящих в себе саму композицию, настройки, а так же порядок и положение интерактивных элементов.

Цель игры заключается в своевременном нажатии на появляющиеся в такт музыки ноты, удержание курсора в пределах подвижных слайдеров и раскручивание спиннеров на максимально возможное значение. За все вышеперечисленные действия игроку начисляются очки рейтинга. Чем точнее игрок попадает в такт или удерживает курсор в центре указанных зон, тем больше очков рейтинга ему начисляется. Полученные очки влияют на позицию игрока в мировом рейтинге игроков.

В данной игре частично используются алгоритмы для частотного анализа композиций, но остаётся необходимость вручную определять положение нот на экране, а так же других интерактивных элементов. Данный аспект является основным минусом такого подхода, потому как конечному пользователю хочется как можно быстрее получить желаемый результат в

виде готовой интерактивной композиции, а не задумываться о расположении элементов по мере игры.

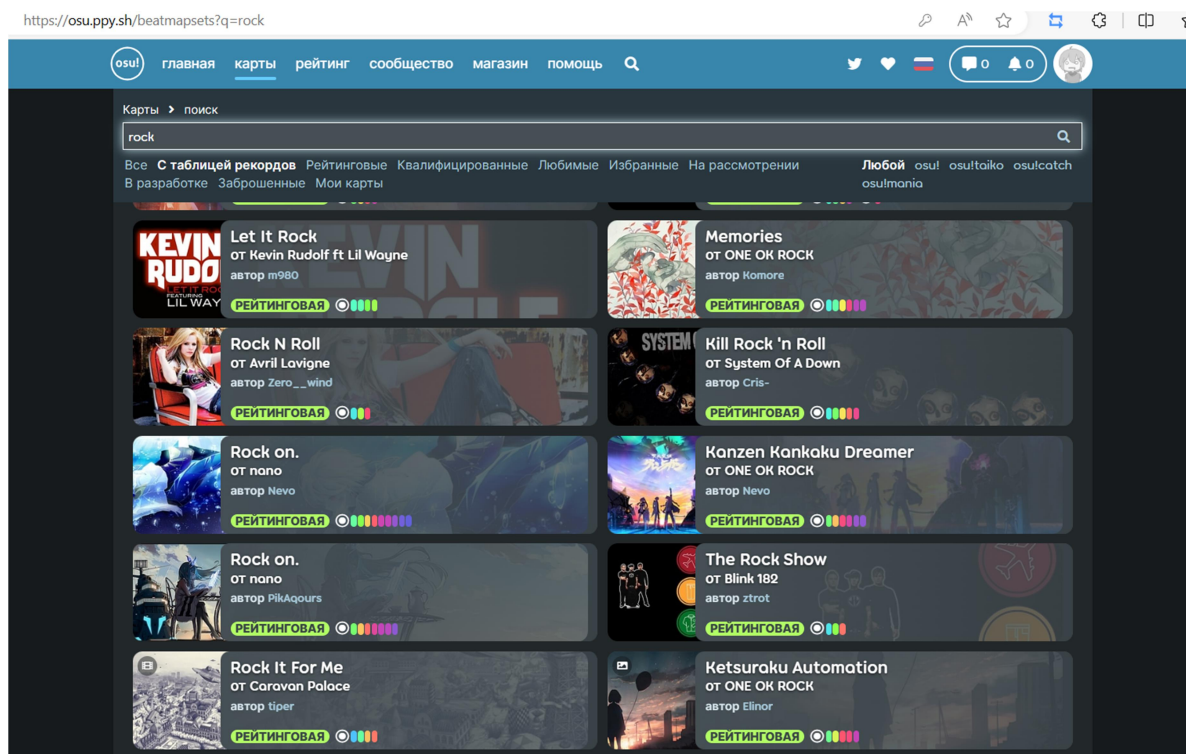


Рисунок 1.4 – Библиотека игры «Osu!»

В связи со своей доступностью и наличием большой библиотеки композиций, данный проект стал крайне популярен как среди фанатов жанра игры и определённых прослоек пользователей, увлекающихся восточной музыкой, так и среди пользователей, использующих данную платформу для тренировки реакции и точности нажатия клавиш.

1.1.3 One Hand Clapping

One Hand Clapping – это вокальный 2D платформер от небольшой инди студии (рисунок 1.5). Несмотря на небольшую известность проекта, хочется отметить нестандартный подход разработчиков к реализации взаимодействия игрока с внутри игровым миром.

В отличие от вышеперечисленных примеров, в данной игре отсутствует фактор борьбы с другими игроками за позицию в локальном или мировом рейтинге. Так же в данной игре не делается фокус на точности нажатия кнопок в соответствии с музыкальным сопровождением. One Hand Clapping предлагает игрокам использовать свой голос как инструмент для преодоления препятствий и решения головоломок. Игра погружает в фантастический мир, где правильные звуковые частоты помогают освещать путь, создавать платформы и управлять элементами окружения. Это не

только новый способ взаимодействия с игровым миром, но и возможность для игроков тренировать свои вокальные навыки и чувство ритма.



Рисунок 1.5 – Игра One Hand Clapping

Управление в игре осуществляется посредством клавиатуры и микрофона. Все головоломки в игре построены так, что для их решения необходимо пропевать ноты определенной тональности. За счёт встроенных алгоритмов анализа аудио сигнала, поступающий в микрофон сигнал, преобразуется в частотно временной график, кривая которого, как пример, может послужить мостом для прохождения препятствий (рисунок 1.6). Все взаимодействия с персонажами и объектами в игре, так же зачастую осуществляют посредством голосового ввода.



Рисунок 1.6 – Пример использования голоса в One Hand Clapping

1.2 Обзор инструментов и технологий

Данный подраздел включает в себя краткий обзор инструментов и технологий, которые использовались при разработке дипломного проекта.

1.2.1 Unreal Engine 4

Unreal Engine 4 – является программной платформой, разработанной компанией Epic Games. Имеет в наличие широкий спектр инструментов, для создания визуальных эффектов, 3D графики, игр и симуляций. Четвертая версия данной платформы была разработана в 2012 году. Именно с четвертой версии данное программное обеспечение перешло на бесплатную форму распространения. С момента выпуска Unreal Engine 4 стала одной из самых популярных платформ для разработки игр по всему миру (рисунок 1.7).

Основные аспекты, выделяющие Unreal Engine 4 на фоне других платформ для разработки игр:

1. Большое сообщество. Unreal Engine 4 обладает активным сообществом разработчиков. Существует форум разработчиков Epic, где обсуждаются различные темы, связанные с разработкой на Unreal Engine 4. Специалисты компании часто отвечают на вопросы по разработке, а так же осуществляют консультацию и помощь с разработкой для компаний, работающих с продуктом компании. Кроме того, Unreal Engine предлагает магазин, где пользователи могут приобрести и продать различные плагины и контент, созданный другими пользователями.

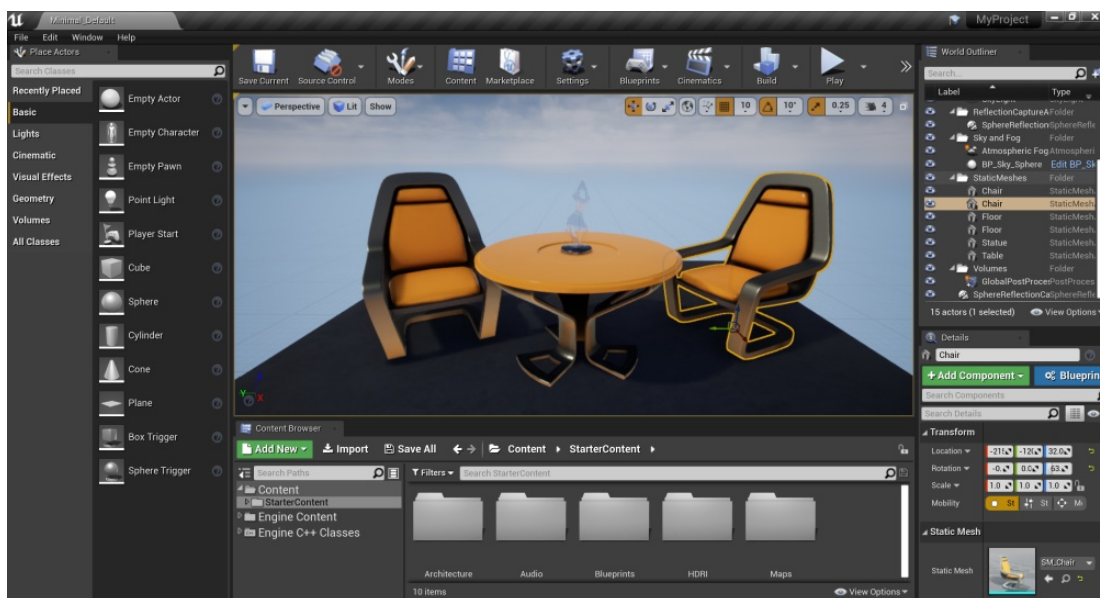


Рисунок 1.7 – Рабочая среда Unreal Engine 4

2. Языки программирования. Unreal Engine 4 поддерживает C++ и проприетарного Blueprint Visual Scripting для создания игровых элементов. Blueprint Visual Scripting – это визуальная система скриптов, которая

позволяет разработчикам значительно увеличить скорость прототипирования и создания полноценных игровых элементов, без написания кода.

3Качественная документация. Документация Unreal Engine 4 доступна на многих языках и содержит подробные сведения по установке и работе с программным обеспечением. Она включает в себя введение в программирование на C++, обзоры предустановленных функций и технологий, руководства по работе с определёнными аспектами платформы, будь то графика или физические симуляции, и многое другое. Также имеется поддержка следующих инструментов программирования на C++:

- C++ Class Wizard: Инструмент для быстрого создания новых классов и компонентов;
- Console Variables: Инструмент для управления различными параметрами с использованием консоли;
- Low-Level Memory Tracker: Инструмент для детального отслеживания используемой создаваемым продуктом памяти устройства.

4Поддержка современных технологий компьютерной графики. Unreal Engine 4 поддерживает множество современных технологий компьютерной графики. Он включает в себя системы освещения и теней, материалы и текстуры, визуальные эффекты и постобработку. Кроме того, Unreal Engine 4 поддерживает технологии RTX, Dolby Atmos, VR, ARCore, Niagara, Houdini и многие другие.

5Поддержка всех известных платформ. Unreal Engine 4 поддерживает разработку для множества платформ, включая Windows, macOS, Linux, iOS, Android, PlayStation, Xbox, Nintendo Switch и другие.

6Интеграция со средой разработки. Для разработки на C++ в Unreal Engine 4 используется Microsoft Visual Studio или Apple Xcode. Для корректной работы данных сред разработки, Epic Games были разработаны специализированные плагины, которые находятся в открытом доступе.

7Модульная архитектура. Unreal Engine 4 построен на модульной архитектуре. Имеется возможность добавления собственных модулей и расширений функциональности платформы.

1.2.2 Microsoft Visual Studio

Microsoft Visual Studio – это интегрированная среда разработки (IDE), разработанная компанией Microsoft (рисунок 1.8). Является одним из важнейших инструментов, используемых при разработке данного дипломного проекта и для разработки программного обеспечения в целом.

Данная интегрированная среда разработки поддерживает широкий спектр языков программирования, таких как C++, Node.js, Python и R, что позволяет разработчикам создавать приложения для различных платформ, включая Windows, Azure и многие другие. Одной из выдающихся особенностей Visual Studio является её способность улучшать продуктивность разработчиков за счёт таких функций, как IntelliSense,

которая предлагает автодополнения кода, и CodeLens, которая предоставляет полезную информацию прямо в редакторе кода.

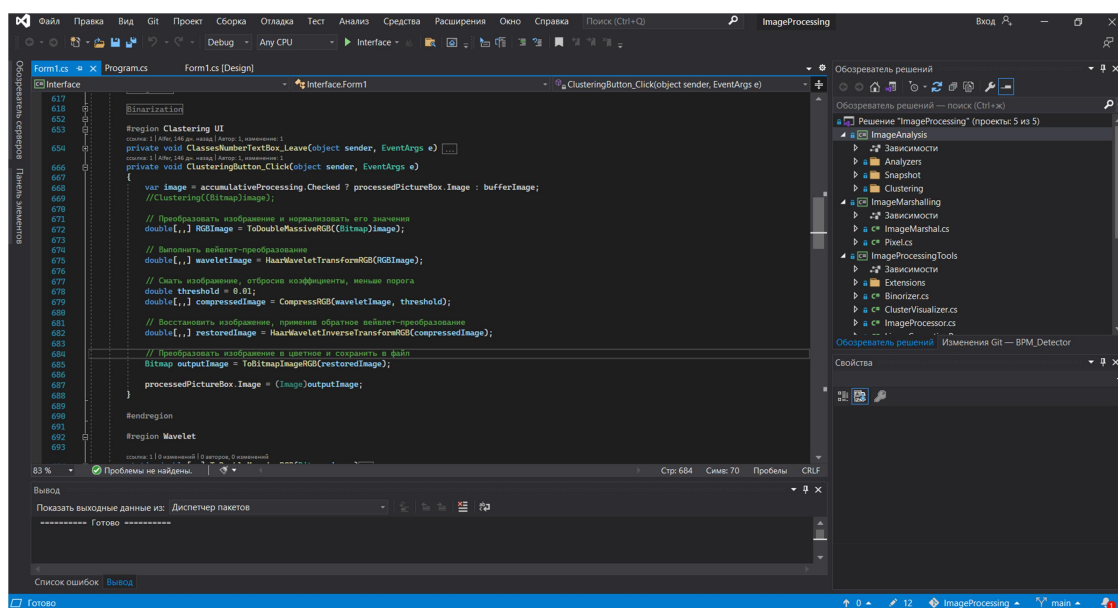


Рисунок 1.8 – Интегрированная среда разработки Microsoft Visual Studio

Visual Studio интегрируется с GitHub и Unreal Engine 4, что упрощает управление версиями и работу над проектом. Кроме этого данное ПО обладает детальными настройками, как самой среды разработки, так и разрабатываемого проекта. Для более тонкой настройки можно использовать доступные к установке плагины и дополнительные инструменты от других пользователей. Эти инструменты и возможности интеграции делают Visual Studio идеальным выбором для командных и одиночных проектов.

Visual Studio также выделяется на фоне конкурентов благодаря своим возможностям для отладки и диагностики, предоставляя разработчикам инструменты для эффективного поиска и устранения ошибок в коде. Это включает в себя визуализации для асинхронных операций и автоматические анализаторы, которые помогают оптимизировать производительность и качество кода. Такие возможности делают Visual Studio не только инструментом для написания кода, но и платформой для обеспечения высокого качества разработки программного обеспечения.

Использование Visual Studio в дипломном проекте позволяет значительно ускорить процесс разработки и повысить его качество, благодаря широкому спектру инструментов и функций, которые она предлагает.