

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ
Школа бакалавриата

ОТЧЕТ

По проекту
«Разработка игрового ПО для персонального компьютера в жанре аркада»
по дисциплине «Проектный практикум»

Заказчик: Фамилия И.О.

Исаков Дмитрий
Николаевич

Куратор: Фамилия И.О.

Исаков Дмитрий

ученая степень, ученое звание, должность

Николаевич

Студенты команды Headphones

Ким Кирилл

Фамилия И.О.

Владиславович

Фамилия И.О.

Гайнетдинов Сергей

Витальевич

Фамилия И.О.

Марченко Максим

Константинович

Фамилия И.О.

Хохрин Михаил

Сергеевич

Фамилия И.О.

Кислицин Лев

Владиславович

Екатеринбург, 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Основная часть	5
1.1 Информация о работе каждого участника	5
1.2 Требования заказчика и пользователей к программному продукту и составление плана действий (backlog)	6
1.3 Анализ и сопоставление аналогов разрабатываемого продукта.....	9
1.4 Обзор архитектуры программного продукта, описание основных компонентов и связей между ними, обоснование выбора архитектурного решения.....	11
1.5 Описание методологии разработки, информация о процессе разработки, отчет о результатах тестирования на промежуточных этапах, разбор выявленных ошибок.....	13
1.6 Информация о планировании деятельности в ходе разработки и распределении задач между участниками команды разработчиков	16
ЗАКЛЮЧЕНИЕ	18
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	20
ПРИЛОЖЕНИЕ А Аналитика целевой аудитории и UserFlow	21

ВВЕДЕНИЕ

Целью данного проекта является разработка игрового программного обеспечения (ПО) для персонального компьютера (ПК) в жанре аркада-раннер. Конкретнее, проект направлен на создание 2D-игры, где главный герой, находясь в офисном пространстве, спасается бегством от преследующей его толпы людей, стремясь добраться до столовой.

Основные задачи проекта включают:

- 1) Определение и анализ целевой аудитории проекта, выявление ее предпочтений и ожиданий;
- 2) Проведение анализа существующих на рынке игр-аналогов для идентификации успешных игровых механик, визуальных решений и потенциальных точек для улучшения или инноваций;
- 3) Разработка ключевых игровых механик: автоматический бег персонажа, прыжок, подкат (скатывание) под препятствия, механика замедления при столкновении с объектами и система отслеживания приближения толпы;
- 4) Проектирование и реализация интуитивно понятного управления, адаптированного для ПК (клавиатура, мышь, геймпад);
- 5) Создание уникального визуального стиля, который будет сочетать силуэтные элементы, эффект параллакс-скроллинга и минималистичный 2D-дизайн с использованием градиентов, мягкой цветовой палитры и эффектов освещения;
- 6) Разработка адаптивного звукового сопровождения, усиливающего атмосферу игры и реагирующего на игровые события.

Актуальность проекта обусловлена устойчивой популярностью аркадных игр и, в частности, жанра «раннер» среди широкой аудитории, особенно среди молодых игроков, ищущих динамичный, увлекательный и визуально привлекательный игровой опыт. Важность проекта заключается в практическом применении студентами знаний и навыков в области игрового

дизайна и разработки, анализе рыночных тенденций и создании конкурентоспособного продукта, отвечающего запросам целевой аудитории.

Областью применения разрабатываемого программного продукта является индустрия компьютерных игр, конкретно – сегмент казуальных и аркадных игр для платформы ПК.

Ожидаемые результаты по завершении проекта:

- 1) Функционирующий игровой прототип 2D-аркады-раннера, разработанный на игровом движке Unity;
- 2) Полностью реализованные и отлаженные ключевые игровые механики: автоматический бег, прыжок, подкат, система столкновений и замедления, механика преследования толпой;
- 3) Продукт с проработанным и узнаваемым визуальным стилем, а также качественным и атмосферным звуковым сопровождением;
- 4) Достижение сбалансированного игрового процесса, сочетающего элементы динамичного экшена и моменты для демонстрации ловкости, с целью охвата широкой игровой аудитории.

1 Основная часть

1.1 Информация о работе каждого участника

Ким Кирилл (тимлид, презентер):

- Распределение задач в yougile;
- Создание видео;
- Защита проекта;
- Создание презентации продукта;
- Создание связи в команде;
- Презентация изменений куратору, получение обратной связи;
- Тестирование.

Гайнетдинов Сергей (геймдизайнер):

- Создание userflow;
- Планирование структуры проекта;
- Визуализация взаимодействия классов (CRC-карта);
- Тестирование;
- Управление задачами в разработке.

Марченко Максим (разработчик):

- Работа с движком Unity (изучался с нуля);
- Создание ассетов;
- Реализация функционала;
- Организация кода;
- Предложил идею.

Хохрин Михаил (дизайнер):

- Создание моделей героев;
- Анимация героев;
- Отрисовка фона;
- Создание спрайтов объектов на уровне;
- Рендер кадров для заставки.

Кислицин Лев (аналитик):

- Определение целевой аудитории;
- Анализ аналогов и рыночной тенденции;
- Помощь в определении основных механик игрового ПО;
- Помощь в определении визуального стиля и музыкального сопровождения;
- Создание, структурирование и оформление проектной документации.

1.2 Требования заказчика и пользователей к программному продукту и составление плана действий (backlog)

Игровое ПО для персонального компьютера. пользователь управляет персонажем с помощью мышки \или клавиатуры выполняет действия в соответствии с игровой механикой.

Требования пользователей к программному продукту:

– Геймплей: динамичный и увлекательный, с возможностью выполнения маневров (прыжки, подкаты). Игровой процесс должен предлагать как напряженные моменты погони, так и участки, требующие от игрока ловкости и реакции. Основные механики должны быть интуитивно понятны;

– Управление: простое, отзывчивое и интуитивно понятное управление, адаптированное для стандартных устройств ввода ПК (клавиатура, мышь);

– Визуальный стиль: привлекательный, стильный и запоминающийся. Предпочтение отдается сочетанию силуэтного стиля, параллакс-скроллинга и минималистичного дизайна с использованием градиентов, мягких цветов и эффектов освещения;

– Звуковое сопровождение: качественное, атмосферное и адаптивное. Музыка должна соответствовать динамике игрового процесса (спокойные и напряженные моменты), а звуковые эффекты – усиливать погружение;

- Производительность: плавная работа игры на среднестатистических ПК без значительных технических проблем или падения частоты кадров;
- Сюжет/Атмосфера: игра должна эффективно передавать атмосферу офисного забега и напряжение от преследования толпой на пути к столовой.

План действий (основные элементы backlog):

- Подготовительный этап:
 - Окончательное утверждение концепции и игрового дизайна;
 - Настройка рабочего окружения и проекта в Unity.
- Разработка основного персонажа:
 - Реализация скрипта автоматического бега;
 - Программирование механики прыжка (с настройкой высоты и длительности);
 - Программирование механики подката/скатывания (с настройкой длительности и коллизий);
 - Создание и интеграция анимаций персонажа (бег, прыжок, подкат).
- Разработка игровой среды и уровней:
 - Проектирование и создание базовых элементов офисных локаций (платформы, препятствия);
 - Внедрение системы процедурной генерации или сборки уровней из префабов;
 - Реализация эффекта параллакс-скроллинга для задних планов.
- Реализация механики погони и столкновений:
 - Создание логики поведения "толпы" преследователей;
 - Реализация механики приближения толпы и ее влияния на игровой процесс (например, сужение экрана, изменение музыки);

- Программирование системы столкновений персонажа с препятствиями и реализация механики замедления;
 - Определение условий поражения (когда толпа догоняет игрока).
- Разработка пользовательского интерфейса (UI):
- Создание главного меню (кнопки: "Новая игра", "Обучение" (опционально), "Настройки", "Выход");
 - Разработка экрана загрузки;
 - Создание экрана поражения (с возможностью рестарта или выхода в меню);
 - Интерфейс настроек (громкость звука/музыки, возможно, управление).
- Визуальный стиль и графика:
- Создание графических ассетов (спрайты персонажа, элементов окружения, препятствий) в соответствии с выбранным визуальным стилем;
 - Настройка освещения, шейдеров (если необходимо) и цветовой палитры уровней.
- Звуковое сопровождение:
- Подбор или создание музыкальных треков для различных игровых ситуаций (меню, спокойные участки, напряженные моменты погони);
 - Создание и интеграция звуковых эффектов (прыжок, подкат, столкновение, шаги толпы, звуки интерфейса).
- Тестирование и отладка:
- Регулярное тестирование игровых механик на каждом этапе разработки;
 - Тестирование управления на различных устройствах ввода.
 - Сбор обратной связи и исправление выявленных ошибок и недочетов.

– Создание обучающего уровня (опционально, но рекомендуется):

- Разработка короткого уровня, знакомящего игрока с основными механиками и управлением.

1.3 Анализ и сопоставление аналогов разрабатываемого продукта

Для формирования концепции проекта, определения его уникальных особенностей и избегания распространенных ошибок был проведен анализ следующих игр-аналогов в жанре раннер и аркада:

- 1) Canabalt (2009) — Считается одним из основоположников жанра бесконечных раннеров. Отличается простым управлением (одна кнопка для прыжка) и стильным минималистичным 2D-дизайном;
- 2) Vector (2012) — Паркур-раннер с выразительным силуэтным стилем и акцентом на выполнение разнообразных трюков. Геймплей требует от игрока точности и хорошей реакции;
- 3) Alto's Adventure (2015) — Медитативный раннер с красивым минималистичным дизайном, использованием параллакс-скроллинга, динамической сменой времени суток и погодных условий, а также расслабляющей атмосферой;
- 4) Sky Dancer (2017) — Динамичный паркур-раннер, выполненный в ярком 3D-стиле, с акцентом на прыжки между парящими платформами и ощущение свободного падения;
- 5) Alto's Odyssey (2018) — Продолжение и развитие идей Alto's Adventure, предлагающее новые биомы, улучшенную графику и сохранившее медитативную и атмосферную механику;
- 6) Grand Mountain Adventure (2019) — Раннер с элементами открытого мира, посвященный зимним видам спорта (лыжи, сноуборд). Упор сделан на исследование больших локаций и выполнение трюков.

Сопоставление ключевых характеристик аналогов: Детальное сопоставление представлено в Приложении А. Основные выводы из анализа аналогов, повлиявшие на проектирование игры «Офисный забег»:

– Геймплей: было решено стремиться к сочетанию динамичных, напряженных участков, характерных для Vector, с более спокойными и атмосферными моментами, вдохновленными Alto's Adventure/Odyssey. Важным аспектом является интуитивность управления (как в Canabalt и Alto's Adventure), но с добавлением нескольких ключевых механик (прыжок, подкат) для большей вовлеченности;

– Визуальный стиль: принято решение комбинировать эффектный силуэтный стиль (Vector) с мягкостью градиентов, глубиной параллакс-скроллинга и минимализмом (Alto's Adventure/Odyssey). Планируется смена окружения (офисные коридоры, возможно, другие зоны на пути к столовой) для визуального разнообразия;

– Звуковое сопровождение: цель – создать (или взять из открытых источников) адаптивный саундтрек, где музыка будет меняться в зависимости от интенсивности игрового процесса (например, спокойный эмбиент или lo-fi для исследования и более динамичные треки для моментов погони). Качественные звуковые эффекты (шум толпы, шаги, взаимодействие с объектами) должны усиливать погружение;

– Управление: основной упор делается на простое и отзывчивое управление для ПК (клавиатура, мышь), избегая сложных комбинаций, которые могут отпугнуть казуальных игроков (как, например, в Grand Mountain Adventure на клавиатуре);

– Баланс: необходимо найти золотую середину, избегая монотонности (как в ранних версиях Canabalt) и чрезмерной сложности на начальных этапах (как в Vector для новичков), предлагая игрокам умеренные испытания и приятную атмосферу.

1.4 Обзор архитектуры программного продукта, описание основных компонентов и связей между ними, обоснование выбора архитектурного решения

Разработка игры «Офисный забег» ведется на игровом движке Unity. Этот выбор обоснован следующими факторами:

- Поддержка 2D-разработки: Unity предоставляет мощный набор инструментов для создания 2D-игр, включая систему спрайтов, 2D-физику, анимацию и Tilemap редактор;

- Компонентно-ориентированная архитектура: позволяет гибко конструировать игровые объекты из независимых компонентов (скриптов, коллайдеров, рендереров и т.д.), что способствует модульности и повторному использованию кода;

- Кроссплатформенность: хотя основной целью является ПК, Unity позволяет в будущем относительно легко портировать игру на другие платформы (например, мобильные устройства);

- Большое сообщество и доступность ресурсов: обширная документация, множество учебных материалов и активное сообщество разработчиков упрощают решение возникающих проблем и обучение;

- Asset Store: доступ к магазину ассетов, где можно найти готовые решения, графику, звуки и инструменты, ускоряющие разработку.

Основные компоненты (модули) программного продукта:

1) Модуль управления персонажем (PlayerController):

- а. Отвечает за логику автоматического бега персонажа;
- б. Обработывает ввод от игрока для выполнения прыжка и подката;
- в. Управляет анимациями персонажа (переключение состояний аниматора);
- г. Взаимодействует с физическим движком для перемещения и обнаружения столкновений.

2) Модуль игровой логики и управления уровнем (GameManager/LevelManager):

- a. Управляет состоянием игры (старт, игра, пауза, поражение);
- b. Отвечает за генерацию или загрузку уровня, расстановку препятствий и коллекционных предметов (если будут);
- c. Контролирует логику "толпы" преследователей (скорость, приближение);
- d. Определяет условия победы/поражения.

3) Модуль пользовательского интерфейса (UIManager):

- a. Отображает все элементы интерфейса: главное меню, игровой HUD (очки, дистанция и т.п. – если предусмотрено), экран паузы, экран поражения, настройки;
- b. Обработывает взаимодействие пользователя с элементами UI.

4) Модуль визуализации и рендеринга (RenderingEngine - встроен в Unity):

- a. Отвечает за отрисовку всех игровых объектов, спрайтов, фонов;
- b. Реализует параллакс-скроллинг;
- c. Применяет визуальные эффекты (например, при столкновении, эффекты освещения).

5) Модуль звукового сопровождения (AudioManager):

- a. Управляет воспроизведением фоновой музыки и ее сменой в зависимости от ситуации;
- b. Воспроизводит звуковые эффекты, связанные с действиями персонажа и событиями в игре.

6) Модуль обработки столкновений (PhysicsEngine - встроен в Unity):

- a. Обнаруживает столкновения между персонажем, препятствиями и другими игровыми объектами;

- в. Передает информацию о столкновениях другим модулям для соответствующей реакции (например, замедление персонажа).

Связи между компонентами:

- PlayerController получает команды от пользователя (через систему ввода Unity) и взаимодействует с PhysicsEngine для движения и GameManager для обновления состояния игры (например, при столкновении);
- GameManager управляет ходом игры, взаимодействует с PlayerController (например, для рестарта), UIManager (для отображения информации) и AudioManager (для смены музыки);
- UIManager получает данные от GameManager и PlayerController для отображения актуальной информации и реагирует на действия пользователя;
- AudioManager получает команды от GameManager и PlayerController для воспроизведения соответствующих звуков и музыки;
- Все визуальные компоненты используют RenderingEngine Unity для отображения на экране.

Выбранная архитектура на базе Unity обеспечивает хорошую структурированность проекта, позволяет параллельно разрабатывать различные аспекты игры и легко вносить изменения.

1.5 Описание методологии разработки, информация о процессе разработки, отчет о результатах тестирования на промежуточных этапах, разбор выявленных ошибок

Тимлид занимался доской с распределением задач для команды, налаживал связь со всеми членами команды.

В ходе работы занимался выступлением продукта, его презентацией, получением обратной связи от куратора, а также монтажом видео, катсцен.

Аналитиком была проведена работа по определению целевой аудитории игры. Аналитик выделил возрастную группу (15–35 лет), охарактеризовал ее интересы и игровые предпочтения, что позволило точнее нацелить разработку на ожидания потенциальных игроков. Было учтено, что продукт должен быть интересен как казуальным, так и умеренным геймерам. Важной частью работы стал детальный анализ существующих игр-аналогов в жанре раннер (Canabalt, Vector, Alto's Adventure/Odyssey и др.). На основе этого анализа были выявлены сильные и слабые стороны конкурентов, успешные игровые механики и визуальные решения. Это позволило сформулировать рекомендации по созданию уникального продукта, сочетающего динамичный и спокойный геймплей, а также привлекательный визуальный стиль (сочетание силуэтного стиля, параллакс-скроллинга и минимализма).

Дизайнер собрал модели объектов по заданному списку и в программе Blender отрендерил их силуэты.

Далее в программе Adobe Photoshop, основываясь на референсы, собранные из зарисовок людей и аналогичных игр, был отрисован город и офис, в котором проходит игра.

Дизайнер создал модель главного героя в программе Blender, настроил его шейдера, чтобы их стиль был похож на игру Vector.

С помощью сайта Mixamo сделал 5 анимаций для главного героя (idle, бег, прыжок, подкат, смерть) и отредактировал их.

Далее, дублировав бегущего главного героя без белой рубашки и изменив их анимации бега, дизайнер создал спрайт толпы.

В том же Blender, используя уже готовые модели и некоторые ассеты из открытых источников, дизайнер создал 5 рендеров, которые в будущем будут использованы в заставке.

Геймдизайнер составил userflow от которого отталкивался процесс разработки, следовал по нему как по плану. Вместе с разработчиком была спланирована карта взаимодействий.

В процессе разработки обновлял карту, и тестировал билды игры. Предлагал возможные решения багов и помогал в разработке.

Помогал в управлении отдавая и поясняя некоторые задачи дизайнеру или разработчику.

Разработчик - занимался большей частью разработки. В ускоренном темпе освоил основы разработки в юнити, после приступил к разработке.

Разработка шла по userflow ориентируясь на карту взаимодействий, после добавления крупного функционала создавался билд для тестирования.

Был создан функционал для интро, меню, настроек громкости, динамического фона обработки ввода, переключения анимаций, взаимодействия с объектами. Также была настроена физика, и были сделаны мелкие доработки в анимациях и материалах.

Классы были написаны, соблюдая единый стиль, сформулированный в начале разработки. При разработке (насколько это возможно) соблюдались паттерны ООП и Singleton (ориентация на единственный экземпляр класса).

Процесс разработки:

– Концептуализация и планирование:

- Формулирование основной идеи игры («Офисный забег»);
- Анализ целевой аудитории и ее предпочтений;
- Анализ игр-аналогов и определение ключевых особенностей проекта;
- Выбор технологической платформы (Unity);
- Составление первоначального списка задач (backlog).

– Проектирование:

- Детальная проработка игровых механик;
- Проектирование визуального стиля и создание концепт-артов (если применимо);

- Проектирование звукового оформления;
 - Разработка пользовательского сценария (Userflow);
 - Проектирование архитектуры основных игровых систем.
- Реализация (разработка):
- Итеративная разработка ключевых механик (прототипирование персонажа, управления, базовых препятствий);
 - Постепенное добавление контента (уровни, графика, звук);
 - Реализация пользовательского интерфейса.
- Итерации и доработка:
- На основе результатов внутреннего тестирования и обратной связи (например, от куратора) вносятся коррективы в геймплей, дизайн и техническую реализацию.

1.6 Информация о планировании деятельности в ходе разработки и распределении задач между участниками команды разработчиков

Планирование деятельности:

- Этап 1: Предпроектная подготовка и анализ
- Определение идеи и концепции игры;
 - Исследование целевой аудитории;
 - Анализ существующих аналогов на рынке;
 - Формулирование основных требований к продукту.
- Этап 2: Проектирование
- Детализация игровых механик;
 - Разработка визуального стиля и концепции арт-дизайна;
 - Проектирование звукового сопровождения;
 - Создание пользовательского сценария (Userflow);
 - Выбор инструментов разработки (Unity) и планирование архитектуры.
- Этап 3: Разработка прототипа

- Реализация базовых механик управления персонажем (бег, прыжок, подкат);
 - Создание простого тестового уровня с препятствиями;
 - Первичная реализация механики преследования.
- Этап 4: Полномасштабная разработка
- Создание игровых уровней или системы их генерации;
 - Разработка графических ассетов (персонаж, окружение, UI);
 - Интеграция звукового сопровождения;
 - Реализация всех запланированных игровых механик и систем (UI, настройки и т.д.).
- Этап 5: Тестирование и отладка
- Проведение альфа- и бета-тестирования (внутреннего);
 - Выявление и исправление ошибок (багов);
 - Балансировка сложности и игрового процесса.
- Этап 6: Финализация и подготовка к сдаче
- Полировка игры (улучшение графики, звука, анимаций);
 - Подготовка финальной сборки продукта;
 - Написание отчетной документации.

ЗАКЛЮЧЕНИЕ

По итогам проделанной работы команда успешно завершила разработку 2D-игры в жанре аркада-раннер «Офисный забег». Создан полнофункциональный игровой продукт, соответствующий первоначально поставленным целям и задачам. Игра демонстрирует увлекательный геймплей, основанный на динамичном перемещении персонажа, уклонении от препятствий и побеге от преследующей толпы, в сочетании с уникальным визуальным стилем и атмосферным звуковым сопровождением.

Оценка соответствия программного продукта поставленным требованиям заказчика и пользователя.

Разработанная игра «Офисный забег» полностью соответствует требованиям, определенным на начальных этапах проекта. Реализованы все ключевые игровые механики, включая автоматический бег, прыжок, подкат и систему взаимодействия с препятствиями и преследователями. Управление интуитивно понятно и отзывчиво, поддерживает различные устройства ввода ПК. Визуальный стиль, сочетающий силуэтные элементы и параллакс-скроллинг, создает запоминающуюся атмосферу. Звуковое сопровождение адекватно реагирует на игровые события, усиливая эффект погружения.

Оценка качества программного продукта на основе результатов тестирования.

В ходе всестороннего тестирования программный продукт показал стабильную работу и соответствие заявленным функциональным характеристикам. Были выявлены и устранены незначительные дефекты, что позволило достичь высокого уровня «отполированности» игры. Геймплей был положительно оценен тестовой группой как увлекательный и сбалансированный. Производительность игры находится на должном уровне для комфортной игры на среднестатистических ПК. Таким образом, можно констатировать высокое качество разработанного программного продукта.

В целом, команда успешно справилась с поставленной задачей, создав качественный и завершённый игровой продукт «Офисный забег», который готов к использованию и имеет хорошие перспективы для дальнейшего развития и распространения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. <https://hawk.live/ru/posts/populyarnosti-igr-sredi-raznyh-vozrastov-ssha>
2. <https://unity.com/ru/games>
3. <https://habr.com/ru/articles/828302/>
4. <https://portal.sacralium.game/article/igram-vse-vozrasty-pokorny-kakoy-sredniy-vozrast-mmorpg-igrokov>
5. <https://habr.com/ru/news/512930/>

ПРИЛОЖЕНИЕ А

Аналитика целевой аудитории и UserFlow

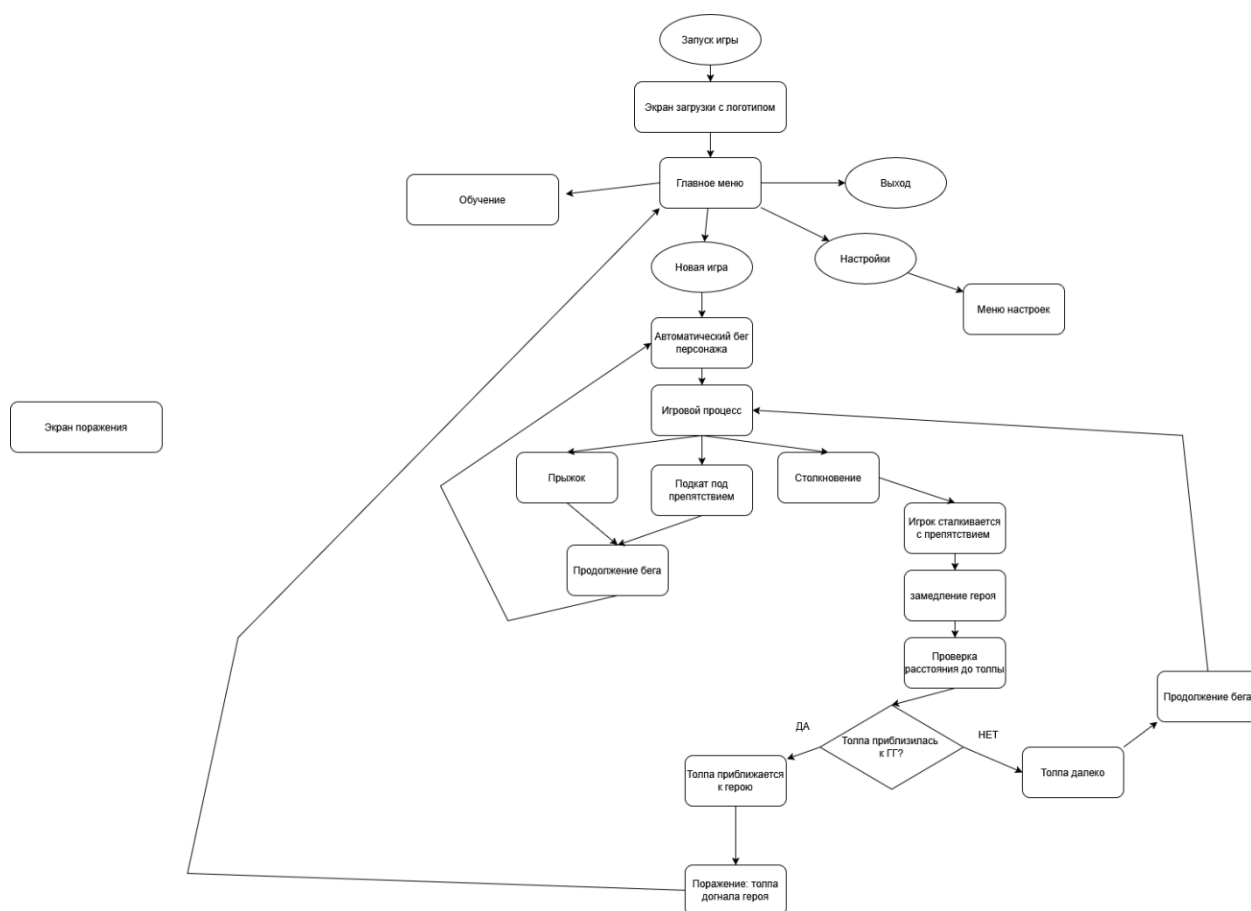


Рисунок 1 – Пример UserFlow используемый в нашем игровом ПО

Возрастная группа	Процент геймеров	Количество геймеров
Под 18 лет	24%	618 млн
18-34 года	36%	1.17 млрд
35-44 года	14%	433 млн

Рисунок 2 – Процентное соотношение возраста и количества игроков

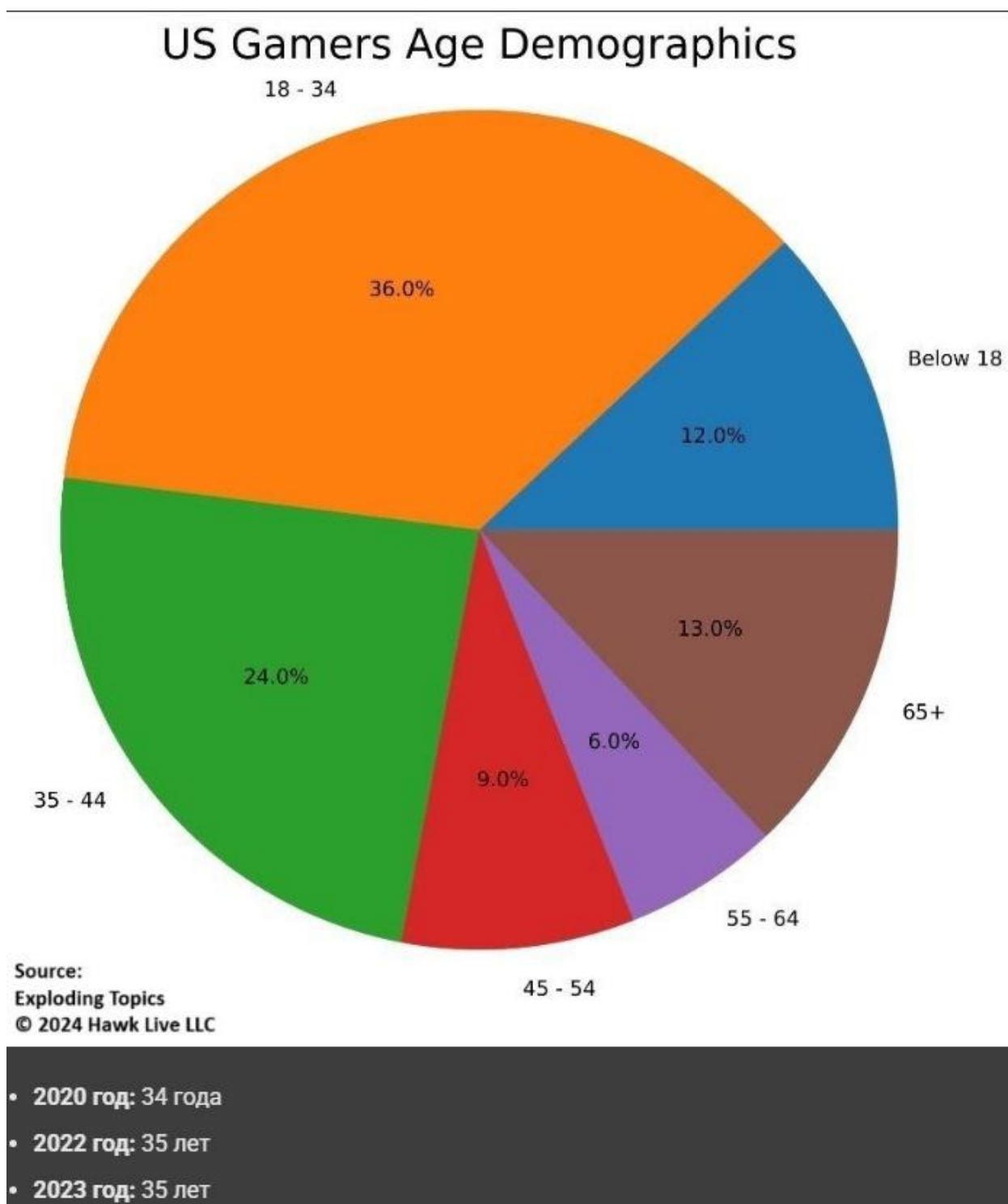


Рисунок 3 – Диаграмма с процентным соотношением возраста и количества игроков