

СОДЕРЖАНИЕ

| | |
|--|----|
| Введение..... | 5 |
| 1 Информационные системы мониторинга действий пользователей | 6 |
| 2 Эргономическое проектирование программного комплекса | 7 |
| 2.1 Разработка алгоритмов работы пользователя | 7 |
| 2.2 Разработка эргономических требований и сценария информационного взаимодействия | 7 |
| 2.3 Эргономическая оценка проектируемой системы и выводы | 11 |
| 3 Разработка программного комплекса для отслеживания и визуализации действий пользователей в игравельной рекламе | 17 |
| 3.1 Структура программного средства | 17 |
| 3.2 Алгоритм работы программы | 18 |
| 3.3 Структура базы данных программного средства | 20 |
| 3.4. Выводы и оценка результатов разработки | 22 |
| 4 Тестирование программного средства | 24 |
| 5 Техничко-экономическое обоснование себестоимости и цены программного комплекса для отслеживания и визуализации действий пользователей в игравельной рекламе..... | 41 |
| 5.1 Характеристика программного средства..... | 41 |
| 5.2 Расчет сметы затрат и отпускной цены программного средства..... | 42 |
| 5.3 Расчет экономического эффекта от реализации программного средства..... | 44 |
| 6 Охрана труда. Обоснование выбора системы освещения в производственном помещении. | 50 |
| Заключение | 56 |
| Список используемых источников..... | 58 |
| Приложение | 59 |

ВВЕДЕНИЕ

Развитие цифровой рекламы привело к появлению такого формата, как игральная реклама (*Playable Ads*), который активно используется для привлечения внимания аудитории благодаря её интерактивности. Очень важной для такой рекламы становится возможность точного отслеживания действий пользователей: это обеспечивает понимание поведенческих паттернов, что необходимо для повышения эффективности кампаний и оптимизации пользовательского опыта.

В рамках данного дипломного проекта разрабатывается специализированный программный комплекс для эффективной визуализации и аналитики взаимодействия с игровой рекламой. Этот комплекс будет бесшовно интегрирован в дашборд – единую платформу управления рекламой, включающую такой функционал, как сборка билдов, замена ассетов, создание локализаций, изменение настроек геймплея.

В отличие от многих существующих инструментов, наш проект ориентирован исключительно на специфические потребности игровых рекламных кампаний, что позволяет сосредоточиться на высоком качестве функциональности под данную узкую сферу. Особое внимание уделяется эргономике: интерфейс разработан для упрощения анализа пользовательских данных и оптимизации рабочего процесса аналитики.

Комплекс будет включать в себя продвинутые аналитические инструменты с применением современных методик обработки больших данных и машинного обучения для точной визуализации ключевых показателей взаимодействия пользователей со рекламой. Это обеспечивает не только глубокое понимание поведенческих паттернов, но также быстрое реагирование на нужды целевой аудитории при управлении маркетинговыми кампаниями.

Целевая аудитория проекта – разработчики рекламы, маркетологи и *UX/UI* дизайнеры, нуждающиеся в продвинутых аналитических инструментах для отслеживания вовлеченности пользователей.

В контексте современного интерактивного маркетинга предлагаемый комплексный инструмент не только улучшит понимание аудитории, но также послужит фундаментом для будущих инноваций в сфере эргономики рекламных технологий. Его специализация на игровой рекламе обеспечивает высокую функциональность и качество пользовательского опыта – критически важный фактор современных рекламных стратегий, а бесшовность интеграции в дашборд – простоту и удобство настройки.

1 ИНФОРМАЦИОННЫЕ СИСТЕМЫ МОНИТОРИНГА ДЕЙСТВИЙ ПОЛЬЗОВАТЕЛЕЙ

1.1 Анализ предметной области

В современном мире реклама, особенно игральная – это мощный способ вовлечения аудитории в взаимодействие с брендом или продуктом через интерактивные элементы. Пользователи взаимодействуют не только стандартными кликами и переходами на другие страницы, но также могут взаимодействовать со сложными игровыми элементами внутри рекламного контента. Эти действия важны для понимания поведенческих шаблонов пользователей, их предпочтений и времени взаимодействия – все это критически важно для оптимизации маркетинговых кампаний и улучшения пользовательского опыта.

Стоит отметить, что обилие возможных действий свойственно именно игровой рекламе: в то время, как для неинтерактивной рекламы обычно важны факты просмотра и перехода, в игровой рекламе в зависимости от конкретного объявления может произойти множество специфических событий (например, выбор уровня, успешность прохождения мини-игры, включение или отключение звука, просмотр видео до определенного момента).

Такие специфичные события важно отслеживать для определения, к примеру, относительной степени заинтересованности тем или иным продуктом, сложности мини-игр, интересности геймплея (по количеству перезапусков игры или ее продолжительности).

Игровая реклама может представлять из себя не только игру. Частым неигровым способом использования этого вида рекламы являются опросы, причем гибкий формат рекламы позволяет проекту включить в себя не только опрос, но и любую дополнительную информацию. Примером такого вида объявлений может послужить проект, в котором после просмотра видео (трейлера фильма) пользователю задается несколько вопросов, ответы на которые отслеживаются и позволяют оценить степень заинтересованности аудитории видеоматериалом.

Отслеживание выбора из нескольких продуктов также является типовой целью для подобного рода рекламных кампаний. Для того, чтобы неявно узнать у пользователя, какой продукт из перечня ему больше нравится, ему дается выбор среди них для игровых целей. Например, несколько игрушечных машинок могут быть даны на выбор пользователю для прохождения гонки.

Трекинг в игровой рекламе также позволяет понять, насколько хорошо удалась задумка проекта, оценив такие показатели, как количество бросивших игру пользователей, количество «заблудившихся» в геймплее

пользователей, количество дошедших до конца игры и количество запустивших игру заново.

1.2 Аналоги информационной системы

На рынке представлено много программных средств для выполнения задачи анализа данных, собранных о взаимодействии с рекламой. Рассмотрим некоторые из них

Google Analytics – популярный инструмент для анализа веб-трафика и поведения пользователей. Он предлагает широкий спектр функций для отслеживания действий пользователей, включая клики, время на странице и конверсии. Положительные стороны данного инструмента – это его мощные аналитические возможности и интеграция с различными платформами. Отрицательные стороны – это сложность настройки и интерфейса, который может быть неинтуитивным для новых пользователей.

Mixpanel – это аналитическая платформа, которая позволяет отслеживать действия пользователей в реальном времени и анализировать их поведение. Основные функции включают создание воронок конверсии, сегментацию пользователей и отслеживание событий. Положительные стороны – это высокая гибкость и возможность глубокого анализа данных. Отрицательные стороны – это высокая стоимость использования и необходимость технических навыков для настройки.

Amplitude – еще один инструмент для анализа поведения пользователей, который предлагает детальные отчеты и визуализацию данных. Основные функции включают отслеживание событий, создание когорт и анализ воронок конверсии. Положительные стороны – это удобный интерфейс и мощные аналитические возможности. Отрицательные стороны – это ограниченные возможности бесплатной версии и сложность интеграции с некоторыми платформами.

Обобщая, рынок полон инструментов для выполнения схожих задач, но все эти инструменты предназначены для более широкого круга задач, чем только работа с игравельной рекламой, за счет чего их целевая аудитория больше, но функционал также в большей степени обобщен и не может быть заточен под специфичные для этой сферы нужды.

1.3 Выводы и постановка заданий на дипломное проектирование

Учет уникальных требований игравельной рекламы для отслеживания действий пользователей показал необходимость специализированного программного комплекса. Традиционные инструменты не всегда способны

удовлетворить специфические нужды этой области и зачастую перегружены несущественными элементами или требуют насыщенного технического понимания их настройки.

Рынок предлагает разнообразные решения, такие как *Google Analytics*, *Mixpanel*, *Amplitude* и прочие, однако эти платформы слишком обобщенны, для адаптации к игравельной рекламе требуются дополнительные усилия по настройке и интеграции. Их использование требует не только технических знаний но также дополнительных ресурсов на визуализацию специфических действий пользователя внутри рекламного контента.

Следует сосредоточиться на разработке инструмента с фокусом на эргономику пользовательского опыта, который будет легко интегрировать в существующую инфраструктуру дашборда. Это решение должно быть более компактным и сфокусированным именно на игравельной рекламе.

Учитывая эту специфику, задача дипломного проекта заключается в разработке специализированного модуля отслеживания взаимодействий пользователя. Комплекс должен поддерживать широкий спектр действий от перезапусков мини-игр до изменения настроек звука или сложности геймплея в реальном времени. Важна также разработка эргономично продуманного интерфейса для простых и мощных аналитических инструментов, обеспечивающих интуитивное понимание поведенческих шаблонов пользователей.

Необходимо обеспечить гибкость при интеграции с маркетинговыми платформами, включая простые средства кастомизации отчетов под требования конкретных кампаний или брендов. Внедрение алгоритмов обработки больших данных и машинного обучения для прогнозирования поведения пользователей должно стать ключевым элементом этого комплекса. Визуализация должна предоставлять четкую картину взаимодействия с возможностью скорой корректировки рекламных стратегий.

Важной составляющей будет также тестирование удобства использования разработанных функций у опытных пользователей маркетинговых платформ, чтобы на основе их обратной связи улучшать интерфейс и функциональность программного решения в процессе его развития до финальной версии.

Это позволит обеспечить комплексный подход к оптимизации пользовательского взаимодействия на современном уровне интерактивности и вовлеченности в рекламе.

2 ЭРГОНОМИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА

2.1 Разработка алгоритмов работы пользователя

В данном разделе описаны алгоритмы работы человека для подсистем «пользователь – ПК – среда».

Алгоритм работы пользователя является важным элементом проектирования системы, поскольку он позволяет определить последовательность действий, необходимых для достижения целей взаимодействия с системой. Это особенно важно в случае комплексных приложений, где пользователю предоставляется доступ к широкому спектру функций [7].

Общий алгоритм действий выглядит так: пользователь авторизуется, выбирает интересующую его версию определенного проекта, меняет при необходимости настройки трекинга для этой версии, просматривает визуализированные данные в виде диаграмм и скачивает отчет в виде файла для дальнейшей работы с ним вне проектируемого программного обеспечения, например, анализа сырых данных или отправка клиенту суммированных данных по дням.

В таблице 2.1 представлен алгоритм работы пользователя проектируемого приложения в виде последовательности операций с описанием обращений к системе обработки информации и органам управления.

Таблица 2.1 – Алгоритм работы человека в процессе взаимодействия с программным комплексом

| Содержание операции | Обращение к системе обработки информации | Обращение к органам управления |
|--|--|---|
| 1 Перейти на сайт сервиса | Ссылка в браузере на экране | Манипуляции с мышью |
| 2 Ввести логин и пароль для входа в учетную запись | Поле «Логин» и поле «Пароль» на экране | Манипуляции с мышью и клавиатурой |
| 3 Войти в учетную запись | Кнопка «Войти» на экране | Наведение курсора на кнопку и нажатие левой кнопки мыши |
| 4 Выбрать проект | Иконки проектов в меню | Наведение курсора на иконку и нажатие левой кнопки мыши |
| 5 Выбрать версию | Названия версий в меню | Наведение курсора на название версии и |

| | | |
|--------------------------------|---|---|
| | | нажатие левой кнопки мыши |
| 6 Открыть настройки версии | Иконка меню настроек | Наведение курсора на иконку и нажатие левой кнопки мыши |
| 7 Настроить трекинг | Поля ввода и переключатели в меню настроек | Манипуляции с мышью и клавиатурой |
| 8 Выбрать фильтры | Списки отслеживаемых значений и их варианты (например, операционная система: Android, IOS, Windows, Linux, MacOS) | Манипуляции с мышью |
| 9 Просмотр визуализаций данных | Диаграммы | Манипуляции с мышью (скроллинг) |
| 10 Скачать отчёт | Иконка скачивания отчёта | Манипуляции с мышью |

Данный алгоритм является важным элементом проектирования системы, так как он определяет порядок действий, необходимых для достижения целей. Его разработка обеспечивает четкую структуру взаимодействия, что особенно важно для сложных приложений с большим количеством функций.

Основное назначение алгоритма – оптимизация пользовательского опыта. Четко выстроенная последовательность шагов позволяет минимизировать ошибки, сократить время на освоение системы и снизить когнитивную нагрузку на пользователя. Это способствует комфортной и быстрой работе, даже для новых пользователей, впервые взаимодействующих с системой.

Кроме того, алгоритм играет важную роль в обеспечении эргономики интерфейса. На его основе создается удобная навигация, интуитивно понятный дизайн и логическая структура интерфейса. Такой подход делает взаимодействие пользователя с системой простым и эффективным, помогая сосредоточиться на выполнении задач, а не на поиске нужных функций.

Алгоритм также ускоряет процесс работы, так как направлен на минимизацию лишних шагов. Определенная последовательность действий позволяет пользователю быстрее достигать своих целей, будь то настройка трекинга, анализ данных или формирование отчетов.

Еще одним важным аспектом алгоритма является унификация работы системы. Он задает единые правила взаимодействия, что особенно актуально для командного использования приложения. Унифицированные процессы помогают избежать путаницы и обеспечивают согласованность работы различных пользователей в рамках одного проекта.

Алгоритм работы пользователя в виде диаграммы представлен в приложении А.

2.2 Разработка эргономических требований и сценария информационного взаимодействия

Эргономические требования к системе человеко-машинного взаимодействия (СЧКС) включают стандарты для системы в целом, ее отдельных компонентов, оборудования и рабочей среды. Эти требования учитывают характеристики человека и направлены на обеспечение его эффективной и безопасной работы.

Для повышения эффективности работы оператора и улучшения эргономических качеств системы необходимо сформулировать конкретные эргономические требования. Они определяются характеристиками оператора и подразделяются на следующие группы:

- антропометрические требования – основаны на анатомических, морфологических и биомеханических особенностях человека;
- физиологические требования – учитывают энергетические и скоростные возможности организма;
- психофизиологические требования – обусловлены возможностями и особенностями сенсорных систем;
- психологические требования – обеспечивают соответствие системы психологическим особенностям человека;
- гигиенические требования – задают безопасные условия для выполнения работы;
- социально-психологические требования – регулируют соответствие конструкции оборудования и организации рабочих мест характеру группового взаимодействия [8].

Дальнейший анализ представляет собой выявление конкретных эргономических требований, составляющих каждую из названных групп показателей. Эргономические требования проектируемой системы представлены в таблице 2.1.

Таблица 2.2 – Эргономические требования проектируемой СЧКС

| Требования | Группа требований |
|--|--------------------|
| Управляемость | |
| Элементы интерфейса (кнопки, поля ввода, графики) должны быть достаточно крупными для точного взаимодействия (в том числе на сенсорном экране) | Антропометрический |

| | |
|---|---|
| Пользователь должен иметь возможность настраивать масштаб отображаемой информации | |
| Цветовая схема и шрифты должны минимизировать нагрузку на зрение пользователя | Физиологический |
| Элементы управления должны подходить под тип вводимых данных (например, для чисел – слайдеры вместо текстового поля; для выбора из небольшого количества вариантов – радиокнопки, а для выбора из большого – выпадающий список) | Психофизиологический |
| Цветовая схема приложения должна учитывать восприятие пользователем контрастности и избегать цветовых сочетаний, вызывающих дискомфорт | Психофизиологический Психологический |
| Соответствие сложности инструкций времени, отводимому на их усвоение | |
| Соответствие объемов информации, требующей запоминания, возможностям памяти человека | Психологический |
| Минимальное количество шагов для доступа к основной информации | |
| Соответствие компоновки органов управления и средств отображения информации стереотипам восприятия | |
| Соответствие индикации срабатывания ОУ сформированным навыкам, наличие индикации хода выполнения функции | |
| Наличие подсказок о следующих шагах работы в системе | |
| Информация должна быть представлена в легко воспринимаемой форме (например, графики и диаграммы вместо сырых чисел) | |
| Соответствие количества одновременно предъявляемых сигналов возможностям внимания человека | |
| Наличие указаний на проблемы, возникающие в процессе обслуживания системы | |
| Наличие предупреждений о нежелательных последствиях некоторых действий соответствие сложности инструкций, времени, отводимому на их усвоение | |
| Отсутствие неоднозначного толкования требований, инструкций и команд | |
| Имеется возможность работы над одним проектом в команде, например, изменения версий проекта, | Социально-психологический |

| | |
|---|----------------------|
| внесенные одним участником отображаются у других | |
| Освояемость | |
| Отсутствие нестандартных сложных для освоения элементов управления | Психологический |
| Иконки соответствуют сформированным у пользователя ассоциациями, например, шестерня открывает настройки | |
| Цвета соответствуют сформированным у пользователя ассоциациям, например, зеленый обозначает успех | |
| Возможность быстрого обучения работе с приложением без необходимости изучения инструкции | |
| Умеренное количество информации на экране, соответствующее возможностям восприятия пользователя | Психофизиологический |
| Отображение информации в течение промежутка времени достаточного для восприятия человеком | |

Опишем конкретные значения для тех требований, которые в них нуждаются:

Элементы интерфейса должны быть достаточно крупными для точного взаимодействия: минимальный размер интерактивного элемента (например, кнопки) — 44x44 пикселя для мобильных устройств, чтобы обеспечить удобное нажатие пальцем.

Пользователь должен иметь возможность настраивать масштаб отображаемой информации: возможность увеличения/уменьшения масштаба интерфейса в диапазоне от 50% до 200% без потери читаемости.

Цветовая схема и шрифты должны минимизировать нагрузку на зрение пользователя: минимальный контраст между текстом и фоном — коэффициент 4,5:1 для обычного текста и 3:1 для крупного. Используемый шрифт должен быть не менее 12 pt (желательно масштабируемый).

Высокая скорость отклика системы: время отклика на любое пользовательское действие (например, фильтрация данных) должно быть не более 300 мс для предотвращения ощущения "задержки".

Минимальное количество шагов для доступа к основной информации: не более 4 кликов (или касаний) от главного экрана до ключевой статистики или нужных данных.

Информация должна быть представлена в легко воспринимаемой форме: графики должны содержать не более 4-5 цветов, четко обозначенные легенды, а текстовые данные сопровождаться иконками или визуальными маркерами для акцента.

2.3 Эргономическая оценка проектируемой системы и выводы

Разработка системы требует учета множества факторов, среди которых одним из ключевых является соответствие эргономическим требованиям. Это необходимо для обеспечения комфортного и эффективного взаимодействия пользователя с приложением. Учет эргономики позволяет создать систему, которая минимизирует нагрузку на пользователя, делает интерфейс интуитивно понятным, а также снижает вероятность ошибок при работе.

Для достижения этой цели важно не только определить базовые эргономические требования, но и разработать конкретные показатели, которые позволят оценить степень их выполнения. Это обеспечивает возможность количественного анализа и последующего улучшения системы. Каждый из показателей отражает ключевые аспекты удобства использования, такие как простота управления, визуальная доступность информации, быстрота освоения функционала и соответствие системы когнитивным особенностям пользователей [9].

В таблице 7.1 представлена спецификация, в которой каждая группа эргономических требований соотнесена с единичными показателями, характеризующими её выполнение. Такой подход позволяет формализовать процесс оценки эргономики и использовать его как инструмент для совершенствования системы.

Таблица 2.3 – Эргономические требования к проектируемой системе
и соответствующие им единичные эргономические показатели

| Эргономические требования | Единичный эргономический показатель | Группа ЭТ |
|--|---|--------------------|
| Управляемость | | |
| А-1 Элементы интерфейса должны быть достаточно крупными для точного взаимодействия (в том числе на сенсорном экране) | Размер элементов интерфейса достаточен для однозначного попадания по ним на любых целевых устройствах | Антропометрический |
| А-2 Пользователь должен иметь возможность | При изменении масштаба в браузере скейлинг | |

| | | |
|---|---|---|
| настраивать масштаб отображаемой информации | происходит корректно, элементы не перекрывают друг друга | |
| Ф-1 Цветовая схема и шрифты должны минимизировать нагрузку на зрение пользователя | Используется простой для восприятия шрифт без засечек (гротеск), текст и элементы управления различимы | Физиологический |
| ПФ-1 Элементы управления должны подходить под тип вводимых данных | Для чисел используются слайдеры вместо текстового поля; для выбора из небольшого количества вариантов – радиокнопки, а для выбора из большого – выпадающий список | Психофизиологический |
| ПФ-2 Цветовая схема приложения должна учитывать восприятие пользователем контрастности и избегать цветовых сочетаний, вызывающих дискомфорт | Цветовая схема приложения обеспечивает достаточный контраст и исключает использование цветовых сочетаний, вызывающих дискомфорт у пользователей | Психофизиологический Психологический |
| П-1 Соответствие объемов информации, требующей запоминания, возможностям памяти человека | Объем информации для запоминания ограничен до 5-7 элементов в одной функции, что соответствует средним возможностям памяти человека | |
| П-2 Минимальное количество шагов для доступа к основной информации | Доступ к основной информации осуществляется за 4 шага или меньше | Психологический |
| П-3 Соответствие компоновки органов управления и средств отображения информации стереотипам восприятия | Органы управления скомпонованы по принципу схожего назначения, например – все настройки трекинга вместе | |

| | | |
|--|--|-----------------|
| П-4 Соответствие индикации срабатывания органов управления сформированным навыкам, наличие индикации хода выполнения функции | При взаимодействии с элементами управления визуально понятно, что изменение требуемого значения учтено | Психологический |
| П-5 Информация должна быть представлена в легко воспринимаемой форме (например, графики и диаграммы вместо сырых чисел) | Статистика изображена в виде диаграмм | |
| П-6 Соответствие количества одновременно предъявляемых сигналов возможностям внимания человека | Отсутствие таких ситуаций, как, например, несколько всплывающих окон | |
| П-7 Наличие указаний на проблемы, возникающие в процессе обслуживания системы | Отображение ошибок в понятном для пользователя виде | |
| П-8 Наличие предупреждений о нежелательных последствиях некоторых действий соответствие сложности инструкций, времени, отводимому на их усвоение | Предупреждение о неотвратимости, например, удаления проекта из приложения | |
| П-9 Отсутствие неоднозначного толкования | Отсутствие нечетких формулировок в интерфейсе, | |

| | | |
|--|---|---------------------------|
| требований, инструкций и команд | дополнительные пояснения в случае необходимости | |
| СП-1 Имеется возможность работы над одним проектом в команде | Изменения версий проекта, внесенные одним участником отображаются у других | Социально-психологический |
| Освояемость | | |
| П-10 Отсутствие нестандартных сложных для освоения элементов управления | В интерфейсе имеют место только знакомые пользователю элементы управления, такие, как текстовые поля, слайдеры, выпадающие списки и тому подобные | Психологический |
| П-11 Иконки соответствуют сформированным у пользователя ассоциациями | Подбор подходящих иконок, например, шестерня соответствует настройкам, а схематичное изображение диаграммы – статистике | |
| П-12 Цвета соответствуют сформированным у пользователя ассоциациям | Правильный подбор цветов, например, красный сообщает о неудаче | |
| П-13 Возможность быстрого обучения работе с приложением без необходимости изучения инструкции | Интерфейс приложения сам содержит в себе все необходимые объяснения по работе с ним | |
| ПФ-3 Умеренное количество информации на экране, соответствующее возможностям восприятия пользователя | Отсутствие таких ситуаций, как появление нескольких диалоговых окон или предупреждений | Психофизиологический |
| ПФ-4 Отображение информации в | Информация, которая отображается в течение | |

| | | |
|--|--|--|
| течение промежутка времени достаточного для восприятия человеком | ограниченного времени может быть прочтена пользователем за это время | |
|--|--|--|

Оценим единичные эргономические показатели.

Единичные эргономические показатели оцениваются по бинарной шкале: 1, если значение соответствует рекомендациям, и 0, если нет.

Оценка значений единичных и групповых эргономических показателей приведена в таблице 7.2.

Таблица 7.2 – Оценка значений единичных и групповых эргономических показателей

| Группа ЭП | Значение единичных ЭП | Значение групповых ЭП |
|---------------------------|----------------------------------|------------------------|
| Управляемость | | |
| Антропометрический | А-1 = 1 А-2 = 0 | $1 \cdot 1 / 2 = 0,5$ |
| Физиологический | Ф-1 = 1 | $1 \cdot 1 / 1 = 1$ |
| Психофизиологический | ПФ-1, ПФ-2 = 1 | $2 \cdot 1 / 2 = 1$ |
| Психологические | П-1, П-2, ..., П-9 = 1 | $9 \cdot 1 / 9 = 1$ |
| Социально-психологические | СП-1 = 1 | $1 \cdot 1 / 1 = 1$ |
| Освояемость | | |
| Психологические | П-10, П-11, П-13 = 1 П-12 = 0 | $3 \cdot 1 / 4 = 0,75$ |
| Психофизиологический | ПФ-3, ПФ-4 = 1 | $2 \cdot 1 / 2 = 1$ |

Оценим эргономические свойства, важные для проектируемой системы, так как они будут определять её общую эргономичность.

Эргономические свойства СЧМ представляют собой совокупность групповых эргономических показателей. Каждому групповому показателю присваивается весовой коэффициент, поскольку разные показатели играют различную роль в реализации конкретного эргономического свойства. Сумма весовых коэффициентов для каждого эргономического свойства должна равняться единице.

Распределение значений весовых коэффициентов групповых эргономических показателей представлен в таблице 7.3.

Таблица 2.4 – Распределение значений весовых коэффициентов групповых эргономических показателей

| Групповой ЭП | Значение весового коэффициента |
|---|--------------------------------|
| Эргономическое свойство «Управляемость» | |
| Антропометрический | 0,15 |
| Физиологический | 0,15 |
| Психофизиологический | 0,15 |
| Психологические | 0,4 |
| Социально-психологические | 0,15 |
| Эргономическое свойство «Освояемость» | |
| Психологические | 0,6 |
| Психофизиологический | 0,4 |

С учетом данных таблиц 7.1 и 7.2 определим количественное значение эргономических свойств «Управляемость» и «Освояемость»:

$$\text{ЭС}_{\text{Вуправляемость}} = (0,15 \cdot 1) \cdot 3 + (0,15 \cdot 0,75) \cdot 1 + (0,4 \cdot 1) \cdot 1 = 0,9625$$

$$\text{ЭС}_{\text{Восвояемость}} = (0,6 \cdot 0,75) + (0,4 \cdot 1) = 0,85$$

Разные эргономические свойства влияют на эргономичность интерфейса проектируемой системы по-разному, поэтому им присваиваются весовые коэффициенты, отражающие их вклад в общую эргономичность. Свойствам управляемости и освояемости присваиваются равные коэффициенты, так как они одинаково важны для разрабатываемой системы.

Вычислим эргономичность:

$$\text{Э} = (0,5 \cdot 0,9625) + (0,5 \cdot 0,85) = 0,90625$$

Таким образом, эргономичность проектируемого приложения равна 0,90625.

Рассмотрим способы для увеличения эргономичности в таблице 7.4.

Таблица 2.5 – Рекомендации по улучшению пользовательского интерфейса эргономичности проектируемой системы

| Невыполненное эргономическое требование | Предложение по улучшению эргономичности |
|---|---|
| А-2 Пользователь должен иметь возможность настраивать масштаб отображаемой информации | Обеспечить корректность отображения блоков приложения при изменении масштабов |

| | |
|---|---|
| <p>П-12 Цвета соответствуют сформированным у пользователя ассоциациям</p> | <p>В местах, где красный и зеленый цвета использованы не для ассоциируемых с ними действий (ошибка или деструктивное действие, такое, как удаление, для красного и успех или положительный ответ для зеленого) использовать более нейтральные цвета, например, для текста – белый или его оттенки</p> |
|---|---|

Выполнение приведенных рекомендаций позволит увеличить эргономичность приложения и улучшить управляемость и освояемость системы.

3 РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ ОТСЛЕЖИВАНИЯ И ВИЗУАЛИЗАЦИИ ДЕЙСТВИЙ ПОЛЬЗОВАТЕЛЕЙ В ИГРАБЕЛЬНОЙ РЕКЛАМЕ

3.1 Структура программного комплекса

Структура программного комплекса организована с учетом принципов надежности, масштабируемости и модульности, что позволяет эффективно отслеживать действия пользователей в играбельной рекламе и визуализировать их в реальном времени.

Ключевым элементом является легковесный сервис, который отвечает исключительно за прием трекинг-запросов. Его проектирование ориентировано на максимальную стабильность и отказоустойчивость, поэтому он не содержит ни дополнительного функционала, ни избыточных зависимостей, что гарантирует возможность фиксировать каждый входящий запрос в любой момент времени без риска потери данных.

Код, реализующий функциональность трекинг-сервиса, интегрируется непосредственно в монолитный дашборд, что упрощает архитектуру и обеспечивает тесную связь с другими компонентами системы. Архитектура дашборда организована как единое серверное приложение, код которого состоит из множества отдельных сервисов, каждый из которых отвечает за обслуживание эндпоинтов, обработку событий, поступающих с клиентской стороны, выполнение специализированного кода при запуске системы и выполнение регулярных задач, необходимых для поддержания актуальности и целостности данных. Такая структура позволяет изолировать бизнес-логику, связанную с различными функциональными возможностями, а также централизовать управление процессами внутри комплекса.

Взаимодействие между серверной и клиентской частями реализовано посредством постоянного *websocket*-соединения, что обеспечивает двустороннюю связь и мгновенное обновление информации. Это позволяет клиентской части получать данные в режиме реального времени, что является критически важным для оперативного мониторинга и анализа действий пользователей. Соединение по протоколу *websocket* обеспечивает минимальную задержку и высокую производительность при обмене данными, что особенно актуально в условиях интенсивной работы системы трекинга [16].

Клиентская часть дашборда оснащена функционалом для настройки параметров трекинга, визуализации собранных данных и формирования отчетов. Здесь пользователю предоставляется возможность не только наблюдать за поступающей информацией, но и самостоятельно управлять параметрами, влияющими на сбор данных, а также загружать подробные отчеты для последующего анализа. Такой подход позволяет обеспечить

гибкость и адаптивность системы под различные требования и сценарии использования.

Кроме того, сервер дашборда выполняет задачу по сборке финальных билдов игральной рекламы. В процессе сборки, если включена опция трекинга, в билд автоматически интегрируется специальный код, который отвечает за отправку трекинг-запросов непосредственно на легковесный сервис. Это позволяет обеспечить непрерывную цепочку обработки данных от момента взаимодействия пользователя с рекламой до отображения информации в дашборде, гарантируя, что все действия фиксируются и могут быть проанализированы в последующем.

Структура программного комплекса представлена на рисунке 3.1

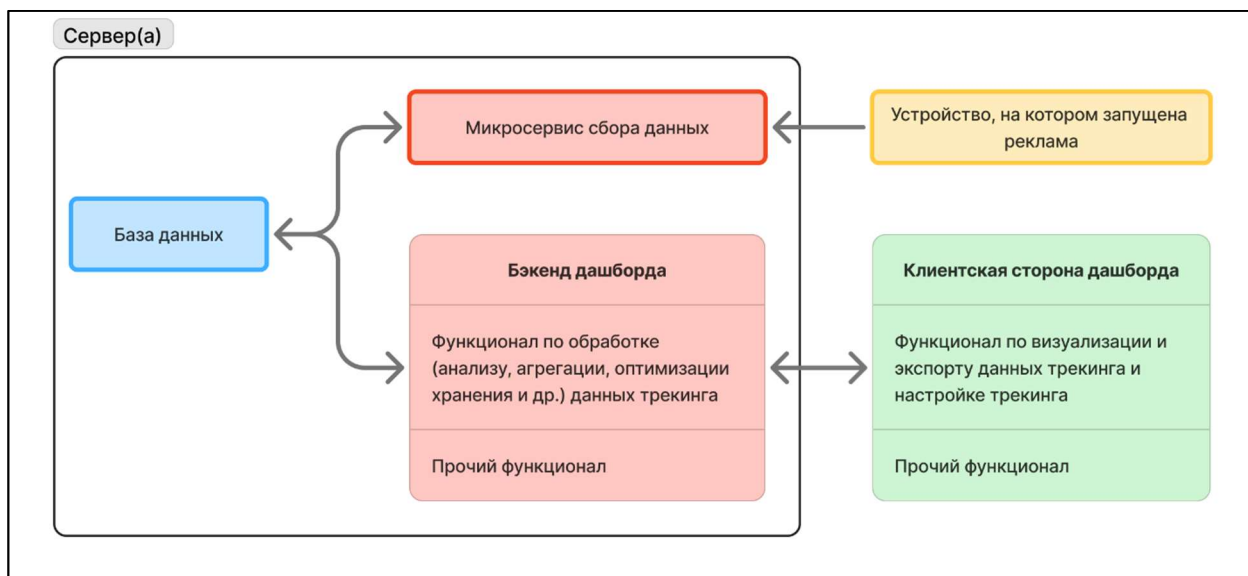


Рисунок 3.1 – Структурная схема программного комплекса

Таким образом, комплексная структура программного обеспечения объединяет высоко оптимизированный легковесный сервис, монолитный дашборд с модульной серверной архитектурой, эффективное *websocket*-соединение для обмена данными в реальном времени и клиентскую часть с широкими возможностями настройки и анализа. Это обеспечивает надежный, масштабируемый и гибкий инструмент для отслеживания и визуализации действий пользователей в игровой рекламе, удовлетворяя как технические, так и аналитические требования конечных пользователей системы.

3.2 Алгоритм работы программного комплекса

Работа программного комплекса включает в себя реализацию ряда алгоритмов, рассмотрим наиболее значимые из них.

Начнем с описания алгоритма клиентского кода, встраиваемого в билд. Триггером к запуску этого алгоритма служит произошедшее отслеживаемое событие. Его инициатором может выступать как пользователь, так и код рекламного приложения сам по себе. Примерами таких событий могут служить загрузка плеера, взаимодействие пользователя с приложением, таймаут, ошибка.

1. Произошло отслеживаемое событие (загрузка, взаимодействие, таймаут, ошибка и др.)
2. Проверяется, инициализирован ли модуль трекинга
3. Если ещё не инициализирован, генерируются и сохраняются уникальные идентификаторы сессии и времени старта
4. При необходимости создаётся и сохраняется уникальный идентификатор пользователя
5. Определяется, требуется ли получение геолокации
6. Если требуется, отправляется запрос за геолокацией и приём событий ставится на паузу
7. После получения ответа (успешного или с ошибкой) помечается статус «геолокация получена» и возобновляется приём событий
8. Устанавливается ожидание появления элемента в поле зрения пользователя (если это задано настройками)
9. После появления элемента в поле зрения возобновляется передача накопленных событий
10. Подключаются внешние библиотеки аналитики через динамическую загрузку и конфигурацию скриптов
11. Подписываются обработчики на все типы отслеживаемых событий в окружении (MRAID, внутренняя шина)
12. При каждом событии формируется объект с параметрами: имя, данные, временные метки, счётчики
13. Для каждого целевого URL проверяются условия: отправка первого события или всех подряд
14. Если элемент не видим — событие помещается в буфер ожидания
15. Если элемент видим или видимость не требуется — событие отправляется немедленно
16. Для глобального трекинга проверяются исключения и ограничения по числу отправок
17. Каждое событие отправляется к провайдеру (XHR, загрузка изображения, динамический скрипт)
18. Нативные события накапливаются в пакет и отправляются по расписанию интервалов
19. При возникновении ошибки приложения регистрируется и добавляется в пакет нативного трекинга

20. После отправки пакет очищается; в случае неудачи данные сохраняются для повторной попытки

21. Перед отправкой каждого запроса URL обогащается актуальными параметрами (время, имя события и т.п.)

Отправленный запрос приходит на сервер. Рассмотрим алгоритм работы сервиса, принимающего и сохраняющего запросы.

1 При старте приложения регистрируются обработчики для приёма одиночных и пакетных событий

2 При поступлении запроса извлекаются параметры контекста (идентификатор проекта и версии) и полезная нагрузка

3 Выполняется валидация обязательных параметров; при некорректных данных формируется ответ об ошибке

4 Для каждого события подготавливается запись: объединяются контекст и данные события

5 Записи событий сохраняются в хранилище асинхронно

6 При необходимости для каждого события также формируется и сохраняется запись информации о пользователе

7 Ошибки при сохранении логируются и влияют на итоговый код ответа

8 Ожидается завершение всех операций записи (событий и, при наличии, пользователей)

9 По результатам всех операций формируется и отправляется клиенту итоговый статус выполнения

Для быстрого предоставления суммированных данных на серверной стороне дашборда работает сервис, который постоянно обрабатывает новые запросы. Опишем алгоритм его работы.

1 Запускается процедура агрегации

2 Выбираются временные интервалы (час, сутки, месяц) и определяется их длительность

3 Для каждого интервала определяется точка последней обработки

4 Последовательно запрашиваются порции неподготовленных данных с учётом лимита и смещения

5 При отсутствии новых записей переход к следующему интервалу

6 Для каждой записи вычисляется начало её интервала и обновляется соответствующий счётчик в агрегированной таблице

7 После обработки всех записей интервала переход к следующему

8 По завершении обработки всех интервалов процедура заканчивается

Далее рассмотрим алгоритм, который обрабатывает данные при скачивании отчета. Для этого алгоритма особенно важно обеспечить стабильную работу с любым объемом данных, так как он работает со всеми записями сразу, а их может быть очень много.

1 Запускается обработка запроса на скачивание отчёта

- 2 Проверяются права доступа пользователя и корректность запрошенной версии
- 3 Извлекаются параметры диапазона дат и формат отчёта
- 4 В зависимости от формата выбирается путь обработки (сырые данные или агрегированные по дням)
- 5 Для сырого формата устанавливаются заголовки и начинается поэтапная выборка порций данных
- 6 Каждая порция преобразуется в строку подходящего вида и сразу отправляется клиенту
- 7 После обработки всех порций соединение корректно закрывается
- 8 Для формата «дни» выполняется единоразовый запрос всех агрегированных записей за период
- 9 Собирается структура таблицы (заголовки, даты, счётчики) в памяти
- 10 Генерируется итоговый файл в нужном формате и отправляется целиком
- 11 По завершении операции обеспечивается корректное завершение передачи данных клиенту

Постоянная работа приложения трекинга очень важна, потому что при его выключении теряются ценные данные пользователей. Необходимо как можно раньше узнавать о неполадках в его работе и логировать проблемы, чтобы потерять как можно меньше данных и знать о времени потенциального сбоя.

- 1 Запускается периодическая проверка доступности сервиса
- 2 Выполняется запрос к тестовому эндпоинту трекинга
- 3 Анализируется результат: успешный ответ или ошибка подключения
- 4 При обнаружении недоступности генерируется запись об ошибке для мониторинга
- 5 Сравнивается текущее состояние с предыдущим зафиксированным
- 6 При изменении статуса запускается рассылка уведомлений администраторам
- 7 Для каждого активного соединения с клиентом отправляется обновлённый статус
- 8 Повтор проверок продолжается через заданный интервал времени

3.3 Структура базы данных программного комплекса

Структура базы данных должна обеспечивать логичное хранение данных трекинга и доступ к ним, а также возможности по оптимизации хранения данных и мониторинг ошибок.

Структурная схема базы данных программного комплекса представлена на рисунке 3.2.



Рисунок 3.2 – Структурная схема базы данных программного комплекса

Опишем все таблицы и их поля.

Таблица *tracking_users* содержит поля *id*, *platform*, *os*, *screen_width*, *screen_height*, *common* и *timestamp*. Поле *id* — уникальный идентификатор пользователя. Поле *platform* указывает на используемую платформу. Поле *os* описывают операционную систему пользовательского устройства. Поля *screen_width* и *screen_height* содержат информацию о ширине и высоте экрана в пикселях соответственно. Поле *common* содержит дополнительные общие

данные по пользователю в свободной форме. Поле *timestamp* указывает на время записи данных.

Таблица *tracking_requests* содержит поля *id*, *version_id*, *user_id*, *event*, *common* и *timestamp*. Поля *id* и *user_id* являются идентификатором запроса и уникальным *ID* пользователя соответственно, связанными через внешний ключ с таблицей *tracking_users*. Поле *version_id* указывает на версию проекта или приложения. Поле *event* описывает тип события (например, открытие страницы или клик по кнопке). Поле *common* содержат дополнительные данные о запросе в свободной форме, а поле *timestamp* фиксирует время создания запроса.

Таблица *tracking_requests_hours* содержит поля *time*, *version_id*, *event*, *last_aggregated_request_id* и *count*. Поле *time* обозначает интервал для агрегации данных по часу. Поле *version_id* связано с версией проекта, а *event* описывает тип события как в таблице запросов. Поле *last_aggregated_request_id* указывает на идентификатор последнего запроса в данной категории, включенного в эту агрегацию. Поле *count* содержит количество событий за определенный интервал времени.

Таблицы *tracking_requests_days* и *tracking_requests_months* аналогично предыдущей представляет данные по дню с полями *time*, *version_id*, *event*, *last_aggregated_request_id*, и *count*.

Таблица *tracking_unavailability* содержит поле *id* как уникальный идентификатор записи и *timestamp*, указывая, когда именно запись была сделана для отслеживания периодов недоступности или ошибок в системе.

В таблице *projects* представлены данные о проектах: *id* (идентификатор), *codename* (кодовое имя проекта), *publisher_id* (*ID* издателя, связанный с таблицей *publishers*), *title*, путь и опции проекта (*path* и *options*) а также *create_timestamp* для времени создания проекта.

Таблица *project_versions* включает в себя идентификатор версии проекта (*id*), связанная версия проекта (*project_id* - ссылка на таблицу *projects*), кодовое имя, название версии, дополнительные настройки, и время создания версии (*create_timestamp*).

Таблица *project_version_settings* хранит настройки для версий проектов с полями: уникальный *ID* записи (*id*), связанный идентификатор версии (*version_id*, связан с таблицей *project_versions*), кодовая информация или наименование параметра (*codename*) и значение параметра в свободной форме (*value*).

Структура используемых в базе данных приложения сущностей представлена в таблицах 3.1-3.10.

Таблица 3.1. Структура сущности *tracking_unavailability*

| Название поля | Тип данных | Описание |
|---------------|------------|----------|
|---------------|------------|----------|

| | | |
|-----------|----------|----------------------|
| id (PK) | INT | Идентификатор записи |
| timestamp | DATETIME | Время записи |

Таблица 3.2. Структура сущности publishers

| Название поля | Тип данных | Описание |
|-------------------------|--------------|--------------------------------------|
| id (PK) | INT | Идентификатор паблишера |
| codename | VARCHAR(255) | Кодовое имя паблишера |
| title | VARCHAR(255) | Отображаемое имя паблишера |
| create_timestamp | DATETIME | Время создания паблишера |
| last_activity_timestamp | DATETIME | Время последней активности паблишера |
| options | MEDIUMTEXT | Опции паблишера |
| active | TINYINT | Статус активности паблишера |

Таблица 3.3. Структура сущности tracking_users

| Название поля | Тип данных | Описание |
|---------------|-------------|-------------------------------|
| id (PK) | BIGINT | Идентификатор пользователя |
| platform | VARCHAR(20) | Название платформы (релиза) |
| os | TEXT | Название операционной системы |
| screen_width | INT | Ширина экрана |
| screen_height | INT | Высота экрана |
| common | TEXT | Дополнительная информация |
| timestamp | DATETIME | Время создания пользователя |

Таблица 3.4. Структура сущности tracking_requests

| Название поля | Тип данных | Описание |
|-----------------|--------------|-----------------------------------|
| id (PK) | BIGINT | Идентификатор запроса |
| version_id (FK) | INT | Идентификатор версии |
| user_id (FK) | BIGINT | Идентификатор пользователя |
| event | VARCHAR(256) | Название события |
| common | TEXT | Дополнительная информация |
| timestamp | DATETIME | Время запроса |
| time_delta | INT | Время, прошедшее с запуска плеера |

Таблица 3.5. Структура сущности projects

| Название поля | Тип данных | Описание |
|---------------|------------|----------|
|---------------|------------|----------|

| | | |
|-------------------|--------------|-----------------------------------|
| id (PK) | INT | Идентификатор проекта |
| codename | VARCHAR(100) | Кодовое имя проекта |
| publisher_id (FK) | INT | Идентификатор паблишера |
| title | VARCHAR(255) | Отображаемое имя проекта |
| path | MEDIUMTEXT | Путь к папке с файлами проекта |
| options | MEDIUMTEXT | Опции проекта |
| create_timestamp | DATETIME | Время создания проекта |

Таблица 3.6. Структура сущности tracking_requests_hours

| Название поля | Тип данных | Описание |
|-------------------------|--------------|---|
| time (PK) | DATETIME | Время начала интервала |
| version_id (FK) | INT | Идентификатор версии |
| event | VARCHAR(256) | Название события |
| last_aggregated_id (FK) | BIGINT | Идентификатор последнего подсчитанного события |
| count | INT | Количество сосчитанных событий |

Таблица 3.7. Структура сущности tracking_requests_days

| Название поля | Тип данных | Описание |
|-------------------------|--------------|---|
| time (PK) | DATETIME | Время начала интервала |
| version_id (FK) | INT | Идентификатор версии |
| event | VARCHAR(256) | Название события |
| last_aggregated_id (FK) | BIGINT | Идентификатор последнего подсчитанного события |
| count | INT | Количество сосчитанных событий |

Таблица 3.8. Структура сущности tracking_requests_months

| Название поля | Тип данных | Описание |
|-------------------------|--------------|---|
| time (PK) | DATETIME | Время начала интервала |
| version_id (FK) | INT | Идентификатор версии |
| event | VARCHAR(256) | Название события |
| last_aggregated_id (FK) | BIGINT | Идентификатор последнего подсчитанного события |
| count | INT | Количество сосчитанных событий |

Таблица 3.9. Структура сущности project_versions

| Название поля | Тип данных | Описание |
|------------------|--------------|------------------------------|
| id (PK) | INT | Идентификатор версии проекта |
| project_id (FK) | INT | Идентификатор проекта |
| codename | VARCHAR(100) | Кодовое название версии |
| title | VARCHAR(255) | Отображаемое название версии |
| options | MEDIUMTEXT | Опции |
| create_timestamp | DATETIME | Время создания версии |

Таблица 3.10. Структура сущности project_version_settings

| Название поля | Тип данных | Описание |
|-----------------|--------------|------------------------------|
| id (PK) | INT | Идентификатор настройки |
| version_id (FK) | INT | Идентификатор версии проекта |
| codename | VARCHAR(100) | Кодовое название настройки |
| value | MEDIUMTEXT | Значение настройки |

3.4. Выводы и оценка результатов разработки

Структура программного комплекса для отслеживания действий пользователей в игровой рекламе разработана с учетом ключевых принципов надежности и масштабируемости. Легковесный сервис трекинга настроен на максимально эффективный прием данных о действиях пользователя без дополнительного функционала или зависимостей, что делает его устойчивым к сбоям и способным обрабатывать запросы круглосуточно без потери информации. Код этого сервиса интегрируется в монолитном дашборде для тесной связи со всеми компонентами системы.

Взаимодействие серверных компонентов организовано через постоянное *WebSocket*-соединение, которое позволяет клиенту получать данные в режиме реального времени с минимальными задержками и высокой производительностью при интенсивном обмене информацией. Клиентская часть оснащена расширенным функционалом настройки параметров трекинга, визуализации данных и формирования детальных отчетов для последующего анализа.

Структура баз данных также играет ключевую роль: она спроектирована так, чтобы эффективно обрабатывать данные о пользовательских действиях с возможностью гибкого агрегирования информации по проектам или категориям событий. В таблицах фиксируется уникальная информация об этапах взаимодействия пользователя со рекламой, включая временные метки и дополнительные атрибуты для детализации анализа.

Система регулярно обновляет дашборд в режиме реального времени через *WebSocket*-соединение, что обеспечивает оперативное отображение данных о действиях пользователей с высокой точностью и гибкостью настроек трекинга.

В заключение, разработка такого комплексного программного комплекса предоставляет надежный инструмент отслеживания действий пользователя в игравельной рекламе с возможностью детальной аналитики. Это обеспечивает не только удовлетворенность технических требований, но также гибкую адаптацию под различные сценарии использования и непрерывное улучшение на основе данных пользовательского поведения.

4 ТЕСТИРОВАНИЕ ПРОГРАММНОГО СРЕДСТВА

Тестирование программного обеспечения — процесс проверки соответствия заявленных к продукту требований и реально реализованной функциональности, осуществляемый путем наблюдения за его работой в искусственно созданных ситуациях и на ограниченном наборе тестов, выбранных определенным образом.

В рамках проведения тестирования программный комплекс был разделен на следующие модули:

- главный экран (подмодули меню пользователя, меню проектов, хедер, рабочая область, меню настроек, панель вида);
- экран профиля (подмодули настройки профиля, история действий);
- экран статистики (подмодули информационная панель, панель настроек отображения).

Тестирование рекламного программного комплекса осуществлялось сотрудниками отдела контроля качества. Работы проводились вручную и с применением автоматизированных тестов. Основным этапом тестирования проходил в период с 10 по 21 марта 2025 года. Все действия фиксировались в системе отслеживания ошибок, где также документировались выявленные дефекты и передавались в разработку для последующей доработки.

Тестирование выполнялось в специально подготовленной среде, максимально приближенной к реальным условиям эксплуатации. Использовались различные конфигурации браузеров (Chrome, Firefox, Edge), операционных систем (Windows 10, macOS Ventura), а также мобильные устройства на базе Android и iOS. В окружении применялись тестовые аккаунты, симулирующие различные пользовательские роли и права доступа. Также проводилась проверка работы рекламных блоков при нестабильном соединении и высокой нагрузке на серверную часть.

По результатам тестирования рекламное приложение получило положительную оценку. Система в целом продемонстрировала стабильную работу, корректную обработку пользовательских сценариев и корректное отображение рекламного контента на всех проверенных устройствах. Большинство выявленных ошибок носили не критичный характер и были оперативно устранены до завершения тестового этапа.

Положительная оценка основана на высоком уровне соответствия фактической работы программного продукта заявленным требованиям, успешном прохождении ключевых пользовательских сценариев, стабильной работе интерфейса и корректной интеграции с системами аналитики и сбора статистики. Также учитывался низкий процент дефектов, не влияющих на основной функционал, а также своевременное устранение всех критических

замечаний. Проведённое тестирование подтверждает готовность рекламного программного комплекса к эксплуатации в рабочей среде.

Test Case представлены в таблице 4.1

Таблица 4.1 – Список *Test Case*

| Но- мер тест- кей- са | При- ори- тет | Тре- бо- ва- ние | Модуль | Подмодуль | Описание тест-кейса | Ожидаемые результаты |
|-----------------------------------|---------------------|---------------------------|------------------|--|--|--|
| 1 | В | А-1 | Главный экран | Меню пользовател я | 1. Открыть главный экран приложения на устройстве с сенсорным экраном. 2. Открыть меню пользователя 3. Нажать на элемент меню. | 1. Меню открылось 2. Целевой элемент меню нажался |
| 2 | С | А-2 | Главный экран | Меню пользовател я, меню проектов, хедер, рабочая область, меню настроек, панель вида | 1. Открыть главное окно приложения или страницу сайта. 2. Применить изменение масштаба отображения при помощи комбинации клавиш (например, CTRL +/-). 3. Перезагрузит | 1. Масштаб отображаемой информации изменяется корректно в соответствии с действиями пользователя. 2. Текст и элементы интерфейса остаются читаемыми и кликабельным и / интерактивны |

| | | | | | | |
|---|---|-----|---------------|--|--|---|
| | | | | | е страницу и повторно примените изменение масштаба. | ми на всех доступных уровнях увеличения / уменьшения. 3. Визуальные компоненты (например, изображения) адаптируются к новому масштабу без потери качества или искажения. 4. Функциональность приложения или сайта сохраняется при изменении масштаба: кнопки нажимаются корректно, поля ввода работают ожидаемым образом. |
| 3 | А | Ф-1 | Главный экран | Меню пользователя, меню проектов, хедер, рабочая область, меню | 1. Открыть главный экран приложения. 2. Визуально оценить цветовую схему фона, текста и | 1. Цветовая схема не вызывает зрительного дискомфорта (отсутствие слишком ярких или ядовитых цветов). |

| | | | | | | |
|---|---|------|---------------|-----------------------|---|---|
| | | | | настроек, панель вида | интерфейсных элементов. 3. Проверить читаемость текста при стандартных условиях освещения. 4. Перейти на экран с длинным текстом или списком. 5. Оценить контрастность шрифта и цветовую гармонию. | 2. Шрифты читаемы при обычной нагрузке на глаза (достаточный размер, межстрочный интервал, контраст). 3. Контраст между текстом и фоном соответствует требованиям доступности. 4. Глаза не устают при длительном чтении текста. |
| 4 | В | ПФ-1 | Главный экран | Меню настроек | 1. Перейти на экран ввода данных. 2. Открыть меню настроек 3. Найти элемент выбора из небольшого количества вариантов (например, 2–5). 4. Найти элемент выбора из большого количества вариантов | 1. Для ввода числовых значений используется слайдер, а не текстовое поле. 2. Для выбора из небольшого количества вариантов используются радиокнопки. 3. Для выбора из большого количества вариантов используется выпадающий |

| | | | | | | |
|---|---|------|---------------|--|---|---|
| | | | | | (например, 10+). | список (dropdown). 4. Все элементы управления соответствуют типу данных, обеспечивая удобство и снижение риска ошибок при вводе. |
| 5 | В | ПФ-2 | Главный экран | Меню пользователя, меню проектов, хедер, рабочая область, меню настроек, панель вида | 1. Открыть разные экраны приложения. 2. Оценить цвет текста, фона и элементов управления. 3. Проверить интерфейс при нарушении цветового восприятия | 1. Контраст между текстом и фоном достаточный. 2. Нет цветовых сочетаний, вызывающих дискомфорт. 3. Все элементы легко различимы и читаемы. 4. Интерфейс понятен при различных типах восприятия цвета. |
| 6 | В | П-1 | Главный экран | Меню настроек | 1. Пройти сценарий с несколькими шагами (например, настройка параметров). | 1. Подсказка есть на каждом этапе. 2. Подсказка краткая, ясная и помогает без лишней |

| | | | | | | |
|---|---|-----|---------------|--|--|--|
| | | | | | 2. Найти подсказку или вспомогательный текст. 3. Оценить ее содержание. | нагрузки на память. |
| 7 | В | П-2 | Главный экран | Меню настроек | 1. Попробовать получить доступ к важной функции (например, включению трекинга). 2. Подсчитать количество действий от главного экрана до нужной информации. | 1. Доступ к функции осуществляется за минимальное количество шагов (желательно 1–2). 2. Навигация простая и интуитивно понятная. |
| 8 | В | П-3 | Главный экран | Меню пользователя, меню проектов, хедер, рабочая область, меню настроек, панель вида | 1. Открыть разные экраны приложения. 2. Оценить расположение кнопок, меню, заголовков и информации. 3. Сравнить компоновку с привычными паттернами интерфейсов (например, кнопка | 1. Элементы управления расположены ожидаемо и логично. 2. Пользователь быстро ориентируется в интерфейсе без обучения. 3. Компоновка соответствует привычным сценариям восприятия. |

| | | | | | | |
|----|---|-----|------------------|-----------------------|---|--|
| | | | | | "назад" слева, "далее" — справа). | |
| 9 | В | П-4 | Главный экран | Меню настроек | <p>1. Нажать на элементы управления (кнопки, переключатели и т.д.).</p> <p>2. Запустить действия, требующие времени (например, загрузку, отправку формы).</p> <p>3. Оценить визуальную обратную связь после действия.</p> | <p>1. Есть четкая индикация нажатия (подсветка, анимация, изменение состояния).</p> <p>2. Отображается ход выполнения действия (прогресс-бар, индикатор загрузки).</p> <p>3. Пользователь понимает, что действие выполнено или находится в процессе.</p> |
| 10 | А | П-5 | Экран статистики | Информационная панель | <p>1. Открыть экраны с данными (например, статистика, отчеты).</p> <p>2. Проверить, как представлена информация: текст, числа, графики или диаграммы.</p> <p>3. Оценить, насколько</p> | <p>1. Информация представлена в удобной для восприятия форме (графики, диаграммы, иконки).</p> <p>2. Тексты и цифры сопровождаются визуальными элементами</p> |

| | | | | | | |
|----|---|-----|---------------|---------------|--|--|
| | | | | | легко воспринять информацию в текущем виде. | для лучшего понимания. 3. Пользователь может быстро понять ключевые данные без лишних усилий. |
| 11 | В | П-6 | Главный экран | Меню настроек | 1. Открыть меню настроек приложения. 2. Оценить количество и типы доступных опций в меню. 3. Проверить, легко ли воспринимать все доступные настройки. | 1. Меню настроек содержит разумное количество опций, не перегружая пользователя. 2. Важные настройки выделяются или сгруппированы для удобства поиска. 3. Пользователь может легко найти нужную настройку без излишней перегрузки вниманием. |
| 12 | В | П-7 | Главный экран | Меню настроек | 1. Перейти в меню настроек приложения. 2. Открыть раздел | 1. При попытке загрузки неподдерживаемого файла появляется ясное |

| | | | | | | |
|----|---|-----|---------------|---------------|---|--|
| | | | | | <p>замены ассетов.</p> <p>3. Попробовать загрузить файл неправильно о формата (например, изображение вместо документа).</p> <p>4. Оценить сообщение о том, что файл не поддерживается.</p> | <p>сообщение об ошибке.</p> <p>2. Сообщение указывает, какой формат файла требуется для загрузки (например, «Поддерживаются только .png и .jpg»).</p> <p>3. Пользователь получает понятные указания, как исправить ошибку.</p> |
| 13 | В | П-8 | Главный экран | Меню проектов | <p>1. Выбрать и открыть нужный проект.</p> <p>2. Перейти в меню проекта.</p> <p>3. Нажать на кнопку "Удалить проект".</p> <p>4. Оценить наличие предупреждения о возможных последствиях удаления.</p> | <p>1. Появляется предупреждение о том, что удаление проекта приведет к потере данных.</p> <p>2. Предупреждение содержит четкие инструкции по последствиям действия и возможным способам восстановления.</p> <p>3. Пользователь</p> |

| | | | | | | |
|----|---|------|---------------|---------------|--|--|
| | | | | | | может отменить операцию, если передумал, и имеет достаточно времени для принятия решения. |
| 14 | А | СП-1 | Главный экран | Меню настроек | 1. Один пользователь открывает проект и изменяет настройки проекта 2. Второй пользователь проверяет проект, чтобы убедиться, что изменения были сохранены и отображаются корректно. | 1. Изменения, внесенные первым пользователем, сохраняются и отображаются во втором пользователе. 2. Второй пользователь видит актуализированные настройки проекта. 3. Все изменения синхронизируются корректно между пользователями. |
| 15 | С | П-10 | Главный экран | Меню настроек | 1. Открыть несколько экранов приложения с элементами управления (например, | 1. Все элементы управления интуитивно понятны и соответствуют |

| | | | | | | |
|----|---|------|---------------|---|--|---|
| | | | | | <p>кнопки, переключатели, меню).</p> <p>2. Попробовать взаимодействовать с элементами управления (например, нажать кнопку, переключить переключатель, раскрыть меню).</p> <p>3. Проверить наличие сложных для освоения элементов управления (например, жестов, нестандартных иконок или анимаций).</p> | <p>стандартам платформы.</p> <p>2. Нет нестандартных или сложных для восприятия элементов управления, которые требуют обучения или дополнительного объяснения.</p> <p>3. Все элементы управления доступны и легко воспринимаются пользователем без лишних усилий.</p> |
| 16 | В | П-11 | Главный экран | <p>Меню пользователя, меню проектов, хедер, рабочая область, меню настроек, панель вида</p> | <p>1. Открыть главный экран.</p> <p>2. Оценить, соответствуют ли иконки общим ассоциациям (например, шестеренка для настроек).</p> | <p>1. Иконки интуитивно понятны и соответствуют привычным ассоциациям.</p> <p>2. Пользователь легко понимает, что обозначает каждая иконка.</p> |

| | | | | | | |
|----|---|------|---------------|--|--|--|
| 17 | В | П-12 | Главный экран | Меню пользователя, меню проектов, хедер, рабочая область, меню настроек, панель вида | 1. Открыть экраны с элементами, использующими цветовые схемы (например, кнопки, уведомления) 2. Оценить, соответствуют ли цвета общим ассоциациям (например, красный для ошибок, зеленый для успеха). | 1. Цвета соответствуют привычным ассоциациям (красный — ошибка, зеленый — успех). 2. Пользователь легко воспринимает цвета и ассоциирует их с правильными действиями или состояниями. |
| 18 | В | ПФ-3 | Главный экран | Меню пользователя, меню проектов, хедер, рабочая область, меню настроек, панель вида | 1. Открыть главный экран 2. Оценить количество информации на экране, включая текст, изображения и элементы управления. | 1. Информация на экране представлена в умеренном количестве, не перегружая пользователя. 2. Все элементы на экране легко воспринимаются и не вызывают визуальной перегрузки. |
| 19 | В | ПФ-4 | Главный экран | Меню пользователя, меню проектов, хедер, | 1. Открыть экран с информацией, которая автоматически | 1. Информация остаётся на экране достаточно |

| | | | | | | |
|--|--|--|--|--|---|--|
| | | | | <p>рабочая область, меню настроек, панель вида</p> | <p>ки исчезает (например, уведомления или всплывающие сообщения). 2. Оценить, сколько времени информация остается на экране, прежде чем исчезнет.</p> | <p>долго, чтобы пользователь мог её воспринять. 2. Время отображения соответствует типу информации (например, уведомления — несколько секунд, важные сообщения — до тех пор, пока пользователь не закроет их).</p> |
|--|--|--|--|--|---|--|

Также проведем ряд проверок графического пользовательского интерфейса с целью выявления дефектов, снижающих эффективность взаимодействия пользователя с системой.

Таблица 4.2 – Перечень проведенных *GUI*-проверок для всего приложения

| Название проверки | Описание проверки |
|--|--|
| Правописание | Лексические, грамматические и пунктуационные ошибки |
| Расположение и выравнивание | Выравнивание по левому или правому краю (в зависимости от требований приложения), отступы, идентичность расстояний между названием и полем. Корректное расположение текста, длинный текст не выходит за границы поля при вводе |
| Длинные названия | Длинные названия корректно обрезаются с помощью многоточия в конце, при наведении возникают хинты с полнотекстовым вариантом |
| Соответствие названий форм/элементов GUI их назначению | Проверка названий форм/элементов GUI с точки зрения их смысловой нагрузки |
| Унификация (стиля, цвета, шрифта, названий) | Единообразие цвета, шрифта, размеров (высоты/ширины), выравнивания полей, названий полей, категорий меню и др. в рамках всего приложения |
| Эффект «нажатия» | Изменение вида ссылок, кнопок, позиций меню и др. при наведении курсора. Изменение вида курсора при наведении на ссылки, кнопки, позиции меню и др. |
| Хинты | Проверка всплывающих подсказок с точки зрения правописания, выравнивания, соответствия назначению |

На основании проведенного тестирования был выявлен ряд дефектов, представленных в таблице 4.3.

Таблица 4.3 – Перечень обнаруженных дефектов

| № | Название дефекта | Важность | Алгоритм воспроизведения | Фактический результат | Ожидаемый результат | Приложение |
|---|--|----------|--|---|---|--------------|
| 1 | Цвет несет в себе неверное мета-сообщение | В | 1. Открыть главный экран 2. Выбрать проект | Зеленый и красный цвета вызывают ассоциации с положительным и отрицательно окрашенной информацией там, где этого быть не должно | Информация окрашена в соответствующие цвета | Рисунок 4.1. |
| 2 | После замены ассета отсутствует какая-либо реакция | В | 1. Открыть главный экран 2. Выбрать проект 3. Открыть настройки проекта 4. Открыть раздел ассетов | После попытки замены ассета невозможно понять, как идет процесс его замены и когда он завершился успешно или ошибочно | Виден прогрессбар или, как минимум, уведомление о том, что процесс идет | Рисунок 4.2. |
| 3 | Из визуального контекста неясно | В | 1. Открыть главный экран | Рядом с надписью о размерах ассетов | Предназначение кнопки ясно из контекста | Рисунок 4.3. |

| | | | | | | |
|--|-----------------------|--|--|---|--|--|
| | предназначение кнопки | | 2. Выбрать проект 3. Открыть настройки проекта 4. Открыть раздел ассетов | располагается неинтуитивная кнопка пересборки ассетов | | |
|--|-----------------------|--|--|---|--|--|

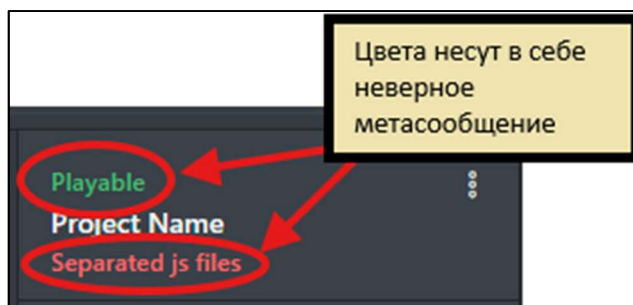


Рисунок 4.1 – Цвет несет в себе неверное метасообщение

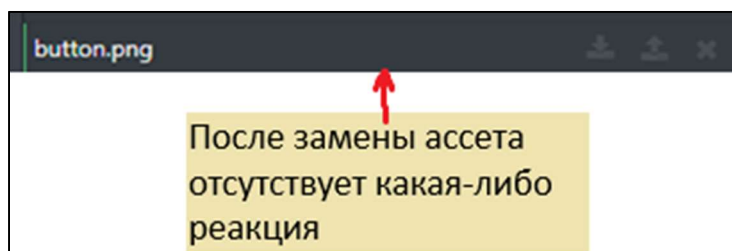


Рисунок 4.2 – После замены ассета отсутствует какая-либо реакция



Рисунок 4.3 – Из контекста неясно предназначение кнопки

5 ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ СЕБЕСТОИМОСТИ И ЦЕНЫ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ОПТИМИЗАЦИИ КОММУНИКАЦИИ С ДЕТЬМИ С АУТИЗМОМ И ЕГО ЭРГОНОМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

5.1 Характеристика разрабатываемого программного обеспечения

Целью данного дипломного проекта является разработка программного комплекса для отслеживания и визуализации действий пользователей в игравельной рекламе. Комплекс ориентирован на автоматизацию процессов сбора, хранения и анализа пользовательской активности, а также на обеспечение наглядного представления этих данных в реальном времени. Такой подход позволяет не только повысить прозрачность пользовательского взаимодействия с рекламными материалами, но и значительно упростить работу специалистов, занимающихся аналитикой и оптимизацией рекламных кампаний.

Ключевой функционал программного комплекса включает прием и обработку трекинг-запросов, визуализацию действий пользователей в дашборде, настройку параметров трекинга, генерацию отчетов, а также сборку финальных билдов рекламных материалов с возможностью автоматического добавления трекинг-кода. Это позволяет обеспечить полный цикл работы: от регистрации событий до их анализа.

Разработка осуществляется с использованием современных веб-технологий, включая *Node.js* для серверной части, *React.js* для клиентского интерфейса дашборда, а также *WebSocket*-протокол для обеспечения обмена данными в реальном времени. Такой технологический стек обеспечивает высокую производительность, быструю реакцию интерфейса и надежность передачи данных, что особенно важно при работе с большим объемом пользовательских событий.

Для успешной реализации проекта необходимо подготовить технико-экономическое обоснование, включающее анализ трудозатрат, расчет стоимости разработки, оценку потенциальной экономической эффективности и целесообразности внедрения комплекса в рабочие процессы. Это позволит обоснованно спланировать проектный бюджет, спрогнозировать ожидаемую отдачу от использования системы и принять рациональные решения о дальнейшем развитии продукта.

5.2 Определение объема и трудоемкости ПО

Объем программного средства определяется исходя из количества функций, реализуемых программой, которые представлены в таблице 5.1, и рассчитываются по формуле (5.1):

$$V_O = \sum_{i=1}^n V_i, \quad (5.1)$$

где V_O – общий объем ПС;

V_i – объем функции ПС;

n – общее число функций.

Среда разработки программного обеспечения – *Visual Studio Code*.

Таблица 5.1 – Функции, реализуемые программой

| Номер | Наименование (содержание) функции | Объем функции V_i | Уточненный объем функции V_y |
|--------|---|---------------------|--------------------------------|
| 101 | Организация ввода информации | 150 | 55 |
| 102 | Контроль, предварительная обработка и ввод информации | 450 | 295 |
| 203 | Формирование баз данных | 2180 | 924 |
| 309 | Формирование файла | 1020 | 625 |
| 706 | Предварительная обработка и печать файлов | 480 | 250 |
| Итого: | | 4150 | 2149 |

По формуле (5.1) находим общий объем ПО $V_O = 2149$.

На основании общего объема ПО определяется нормативная трудоемкость (T_H) с учетом сложности ПО. Так как разрабатываемый продукт относится к 1-ой группе сложности, то нормативная трудоемкость составляет 73 человеко-дней.

$$T_H = 73 \text{ чел.-дн.}$$

Общая трудоемкость ПО рассчитывается на основе нормативной трудоемкости с учетом дополнительного коэффициента сложности $K_{СЛ}$ (5.2), дополнительный коэффициент сложности равен

$$T_O = T_H \times K_{СЛ}, \quad (5.2)$$

где T_O – общая трудоемкость ПО;

T_H – нормативная трудоемкость ПО;

$K_{СЛ}$ – дополнительный коэффициент сложности ПО.

$$T_O = 73 \times (1 + 0,07) = 78,11 \text{ чел.-дн.}$$

Таким образом, общая трудоемкость с учетом округления составила 79 человеко-дней.

На основании общей трудоемкости разработки ПО и установленного периода разработки устанавливается общая плановая численность разработчиков:

$$Ч_p = \frac{T_O}{T_{рд} \times \Phi_{эф}}, \quad (5.3)$$

где $Ч_p$ – плановая численность разработчиков (чел.);

$\Phi_{эр}$ – годовой эффективный фонд времени работы одного работника в течение года (дней в год);

$T_{рд}$ – плановая продолжительность разработки ПС (лет). Устанавливается календарным графиком разработки, равна 0,5 года.

Эффективный фонд времени одного работника ($\Phi_{эф}$) рассчитывается по формуле:

$$\Phi_{эф} = D_{г} - D_{п} - D_{в} - D_{о}, \quad (5.4)$$

где $D_{г}$ – количество дней в году (365 дней);

$D_{п}$ – количество праздничных дней в году (15 дней);

$D_{в}$ – количество выходных дней в году (98 дней);

$D_{о}$ – количество дней отпуска (27 дней).

Таким образом, получаем $\Phi_{эф} = 365 - 15 - 98 - 27 = 225$ дней.

Рассчитаем общую плановую численность работников:

$$Ч_p = \frac{79}{0,5 \times 225} = 0,702.$$

Общая плановая численность разработчиков и трудоемкость служат базой для расчета основной заработной платы.

5.3 Расчет себестоимости и цены программного обеспечения

Исходные данные для расчета основной заработной платы исполнителя приведены в таблице 5.2.

Таблица 5.2 – Разработчик ПО

| Разработчик | Плановый фонд рабочего времени, дней | Месячный оклад ¹ , руб. |
|--------------------------|--------------------------------------|------------------------------------|
| Программист II категории | 79 | 7500 |
| Итого: | 79 | |

¹ на основе данных сайта [<https://www.belstat.gov.by/>] на январь 2025 г.

Часовая тарифная ставка рассчитывается путем деления месячного оклада (T_{Mi}) на установленный месячный фонд рабочего времени.

$$T_{чi} = \frac{T_{Mi}}{МЧФ}, \quad (5.5)$$

где $T_{чi}$ – часовая тарифная ставка руб.;

МЧФ – месячный часовой фонд (168 ч).

Таким образом, часовая тарифная ставка сотрудников равна:

$$T_{ч1} = 7500/168 = 44,64 \text{ руб.}$$

Основная заработная плата разработчиков рассчитывается по формуле:

$$З_o = \sum_{i=1}^n T_{чi} \times T_{ч} \times \Phi_{\text{Э}} \times K, \quad (5.6)$$

где n – количество разработчиков, занятых разработкой ПО;

$T_{чi}$ – тарифная ставка i -го исполнителя в руб.;

$\Phi_{\text{Э}}$ – эффективный фонд рабочего времени i -го исполнителя (дней);

$T_{ч}$ – количество часов работы в день (ч);

K – коэффициент премирования (1,5).

Так как количество разработчиков данной ПО равно 1, то можно следующим образом рассчитать основную заработную плату разработчиков:

$$З_o = 44,64 \times 8 \times 79 \times 1,5 = 42318,72 \text{ руб.}$$

Дополнительная заработная плата на конкретное ПО ($З_{\text{д}}$) включает выплаты, предусмотренные законодательством о труде (оплата отпусков, льготных часов, времени выполнения государственных обязанностей и других выплат, не связанных с основной деятельностью исполнителей), и определяется по нормативу в процентах к основной заработной плате

$$З_{\text{д}} = \frac{З_{\text{о}} \times Н_{\text{д}}}{100}, \quad (5.7)$$

где $З_{\text{д}}$ – дополнительная заработная плата исполнителей на конкретное ПО (руб.);

$Н_{\text{д}}$ – норматив дополнительной заработной платы в целом по организации (20 %).

$$З_{\text{д}} = 42318,72 \times \frac{20}{100} = 8463,74 \text{ руб.}$$

Следующей статьей сметы затрат являются отчисления на социальные нужды. Отчисления на социальные нужды включают в предусмотренные законодательством отчисления в фонд социальной защиты и фонд обязательного страхования (34,6 %) в процентах от основной и дополнительной заработной платы. Отчисления в фонд социальной защиты населения и обязательного страхования от несчастных случаев на производстве и профзаболеваний ($P_{\text{соц}}$) рассчитываются следующим образом:

$$P_{\text{соц}} = \frac{З_{\text{о}} + З_{\text{д}}}{100} \times Н_{\text{соц}}, \quad (5.8)$$

где $Н_{\text{соц}}$ – норматив отчислений на социальные нужды.

$$P_{\text{соц}} = (42318,72 + 8463,74) \times \frac{34,6}{100} = 17570,73 \text{ руб.}$$

Норма расхода материалов в суммарном выражении $Н_{\text{м}}$ определяется либо в расчете на 100 строк исходного кода, либо в процентах к основной заработной плате разработчиков (3 – 5 %). Возьмем 4 % от основной зарплаты. Следовательно, расходы на материалы рассчитываем следующим образом:

$$P_{\text{м}} = \frac{З_{\text{о}} \times Н_{\text{м}}}{100}, \quad (5.9)$$

где H_M – норматив отчислений на материалы.

$$P_M = 42318,72 \times \frac{4}{100} = 1692,75 \text{ руб.}$$

Расходы на спецоборудование (дополнительное ПО) отсутствуют.

Расходы на оплату машинного времени (P_{MB}) определяются по формуле:

$$P_{MB} = \Pi_{MB} \times \frac{V_O}{100} \times H_{MB}, \quad (5.10)$$

где Π_{MB} – цена одного машино-часа (0,05 руб.);

V_O – общий объем ПО (строк исходного кода);

H_{MB} – норматив расхода машинного времени на отладку 100 строк исходного кода (машино-часов).

Средний расход машинного времени 12 часов на 100 строк кода для ПО.

$$P_{MB} = 0,05 \times 2149 \times \frac{12}{100} = 12,89 \text{ руб.}$$

Расходы по статье «Научные командировки» $P_{НК}$ определяются по нормативу, разрабатываемому в целом по организации, в процентах к основной заработной плате:

$$P_{НК} = \frac{3_O \times H_{РНК}}{100}, \quad (5.11)$$

где $H_{РНК}$ – норматив расходов на научные командировки в целом по организации (10 %).

$$P_{НК} = 42318,72 \times \frac{10}{100} = 4231,87 \text{ руб.}$$

Расходы по статье «Прочие затраты» определяются по формуле 5.12:

$$\Pi_3 = \frac{3_O \times H_{ПЗ}}{100}, \quad (5.12)$$

где $H_{ПЗ}$ – норматив прочих затрат (20 %).

$$\Pi_3 = 42318,72 \times \frac{20}{100} = 8463,74 \text{ руб.}$$

Расчет накладных расходов P_H ведется по формуле 5.13:

$$P_H = \frac{3_O \times H_{PH}}{100}, \quad (5.13)$$

где P_H – накладные расходы на конкретную ПС (руб.);

H_{PH} – норматив накладных расходов в целом по организации (50 %).

$$P_H = 42318,72 \times \frac{50}{100} = 21159,36 \text{ руб.}$$

Общая сумма расходов по всем статьям сметы C_{Π} рассчитывается по формуле 5.14:

$$C_{\Pi} = 3_O + 3_D + P_{COЦ} + P_M + P_{MB} + P_{HK} + \Pi_3 + P_H \quad (5.14)$$

$$C_{\Pi} = 42318,72 + 8463,74 + 17570,73 + 1692,75 + 12,89 + 4231,87 + \\ + 8463,74 + 21159,36 = 103913,8 \text{ руб.}$$

Организация-разработчик участвует в освоении ПО и несет соответствующие затраты, на которые составляется смета, оплачиваемая заказчиком по договору. Сметой предусматриваются не только затраты (основная и дополнительная зарплата, начисления на зарплату и т.д.), но и налоги, предусмотренные законодательством, и прибыль организации-разработчика Π_O (5.16).

$$\Pi_O = \frac{C_{\Pi} \times Y_{RP}}{100}, \quad (5.15)$$

где C_{Π} – себестоимость;

Y_{RP} – уровень рентабельности (20 %).

$$\Pi_O = 103913,8 \times \frac{20}{100} = 20782,76 \text{ руб.}$$

Прогнозируемая цена ПО без налогов Π_{Π} :

$$\Pi_{\Pi} = C_{\Pi} + \Pi_O, \quad (5.16)$$

$$\Pi_{\Pi} = 103913,8 + 20782,76 = 124696,56 \text{ руб.}$$

Отпускная цена ПС включает в себя налог на добавленную стоимость:

$$\Pi_o = \Pi_{\Pi} + \text{НДС} = \Pi_{\Pi} + \frac{\Pi_{\Pi} \times \text{Н}_{\text{ндс}}}{100}, \quad (5.17)$$

где $\text{Н}_{\text{ндс}}$ – ставка налога на добавленную стоимость (20 %).

$$\text{НДС} = 124696,56 \times \frac{20}{100} = 24939,31 \text{ руб.}$$

$$\Pi_o = 124696,56 + 24939,31 = 149635,87 \text{ руб.}$$

Прогнозируемый экономический эффект у организации-разработчика (прибыль чистая) проекта составляет 80 % (равен прибыли за вычетом налога на прибыль, равного 20% на январь 2025 г.):

$$\Pi_{\text{ч}} = \Pi_o \times 0,8 = 20782,76 \times 0,8 = 16626,21 \text{ руб.}$$

Для упрощения расчетов до составления сметы затраты на освоение определяются по нормативу ($\text{Н}_o = 10\%$) от себестоимости ПО в расчет на 3 месяца и рассчитываются по формуле 5.18:

$$P_o = \frac{C_{\Pi} \times \text{Н}_o}{100}, \quad (5.18)$$

где Н_o – норматив расходов на освоение (10 %).

$$P_o = 103913,8 \times \frac{10}{100} = 10391,38 \text{ руб.}$$

Организация-разработчик осуществляет сопровождение ПО и несет затраты в соответствии с договором и сметой на сопровождение. Смета составляется по аналогии со сметой на освоение ПО [18]. Для упрощения расчетов для составления сметы затраты на сопровождение определяются по установленному нормативу ($\text{Н}_c = 20\%$) от себестоимости ПО (в расчете на год) и рассчитываются по формуле 5.19:

$$P_C = \frac{C_{\Pi} \times H_C}{100}, \quad (5.19)$$

где H_C – норматив расходов на сопровождение (20 %).

$$P_C = 103913,8 \times \frac{20}{100} = 20782,76 \text{ руб.}$$

Смета затрат на разработку ПО сведена в таблицу 5.3.

Таблица 5.3 – Смета затрат на разработку ПО

| Наименование статей затрат | Условное обозначение | Значение, руб. |
|---------------------------------|----------------------|----------------|
| Основная заработная плата | $З_О$ | 42318,72 |
| Дополнительная заработная плата | $З_Д$ | 8463,74 |

Продолжение таблицы 5.3

| Наименование статей затрат | Условное обозначение | Значение, руб. |
|---|----------------------|----------------|
| Отчисления в фонд социальной защиты населения и обязательного страхования от несчастных случаев на производстве и профзаболеваний | $P_{\text{соц}}$ | 17570,73 |
| Материалы | P_M | 1692,75 |
| Машинное время | $P_{\text{мв}}$ | 12,89 |
| Расходы на научные командировки | $P_{\text{нк}}$ | 4231,87 |
| Прочие затраты | $P_з$ | 8463,74 |
| Накладные расходы | P_n | 21159,36 |
| Полная себестоимость | C_{Π} | 103913,8 |
| Прибыль чистая организации-разработчика | P_o | 20782,76 |
| Прогнозируемая цена ПО | C_{Π} | 124696,56 |
| Затраты на освоение ПО | P_o | 10391,38 |
| Затраты на сопровождение ПО | P_C | 20782,76 |

6 ОХРАНА ТРУДА. НОРМАЛИЗАЦИЯ ГАЗОВОГО И ИОННОГО СОСТАВА ВОЗДУХА В ПРОИЗВОДСТВЕННОМ ПОМЕЩЕНИИ

Обеспечение безопасности труда является одним из важнейших аспектов функционирования производственных помещений. Нормализация газового и ионного состава воздуха играет решающую роль в создании благоприятного микроклимата, повышении эффективности работы сотрудников и снижении риска возникновения профессиональных заболеваний. Современные производственные процессы, интенсивная работа оборудования и технологические особенности различных цехов приводят к тому, что качество воздуха может существенно отклоняться от оптимальных значений. Такие отклонения негативно влияют как на физическое самочувствие работников, так и на надежность и долговечность технологического оборудования, что делает комплексное решение этой задачи крайне актуальным для предприятий, стремящихся к высоким стандартам охраны труда.

На современных производствах основными источниками загрязнения воздуха являются технологические процессы, связанные с использованием химических реагентов, выделением паров растворителей, сгоранием топлива, а также механическими процессами, сопровождающимися образованием пыли и мелкодисперсных частиц. Повышенные концентрации углекислого газа, оксидов азота, угарного газа и других токсичных веществ приводят к ухудшению качества воздуха, что в свою очередь способствует развитию респираторных заболеваний, хронической усталости и снижению концентрации внимания у сотрудников. Наряду с этим, присутствие большого количества положительно заряженных ионов, возникающее в результате недостаточной вентиляции и скопления статического электричества, оказывает негативное воздействие как на здоровье работников, так и на работу электронных систем, что может приводить к сбоям в работе оборудования и даже к его повреждениям.

Для нормализации газового состава воздуха широко применяются современные системы вентиляции, способные обеспечить непрерывное поступление свежего воздуха и эффективное удаление загрязнителей. Приточно-вытяжные установки, оборудованные многоступенчатыми фильтрами, позволяют не только задерживать твердые частицы и аэрозоли, но и эффективно улавливать летучие органические соединения и химически активные газы. Использование активированного угля в качестве абсорбента обеспечивает высокую адсорбционную способность, что позволяет снижать концентрацию растворителей и других вредных веществ до допустимых норм. Каталитические фильтры, предназначенные для разложения угарного газа и окислов азота, дополняют комплекс мер по очистке воздуха. Регулярное техническое обслуживание этих систем, своевременная замена фильтров и

модернизация оборудования являются неотъемлемой частью стратегии обеспечения безопасности труда на предприятии.

Особое внимание уделяется нормализации ионного состава воздуха, особенно в помещениях с высоким уровнем электроники и автоматизированных систем. Статическое электричество, возникающее в результате накопления положительных ионов, может вызывать не только дискомфорт у сотрудников, но и приводить к сбоям в работе сложной техники, что в свою очередь негативно сказывается на производственных процессах. Для решения этой проблемы используются ионизаторы, которые насыщают воздух отрицательно заряженными ионами, нейтрализуя избыток положительных зарядов. Это позволяет снизить уровень статического электричества, улучшить условия работы электронного оборудования и повысить общий комфорт в производственных помещениях. Применение таких устройств особенно актуально в лабораториях, цехах с чувствительной электроникой и других зонах, где стабильность микроклимата имеет критическое значение.

Организационные меры также играют важную роль в обеспечении качества воздуха на производстве. Регулярный мониторинг состояния воздушной среды с помощью специализированных датчиков и газоанализаторов позволяет оперативно выявлять отклонения от установленных норм и принимать необходимые меры. Постоянное наблюдение за показателями концентрации вредных веществ в сочетании с анализом динамики изменений дает возможность прогнозировать потенциальные риски и своевременно корректировать работу систем вентиляции и кондиционирования. Помимо технического контроля, значительную роль играет обучение персонала правилам безопасной работы в условиях, где качество воздуха может резко меняться. Сотрудники должны быть проинформированы о мерах предосторожности, знать порядок действий в случае аварийных ситуаций и уметь применять средства индивидуальной защиты, такие как респираторы и специальные маски.

Правильная организация рабочего пространства также способствует улучшению качества воздуха. Расположение оборудования, источников тепла и потенциальных загрязнителей должно быть тщательно продумано с учетом принципов естественной и искусственной циркуляции воздуха. Даже незначительные изменения в планировке могут привести к улучшению вентиляционных потоков и снижению концентрации вредных веществ в определенных зонах. Оптимизация расстановки оборудования, создание свободных зон для нормального воздухообмена и внедрение систем автоматического управления климатом помогают обеспечить стабильные показатели микроклимата, что напрямую влияет на работоспособность сотрудников и надежность технологических процессов.

Соблюдение нормативных требований и стандартов охраны труда является обязательным условием для всех производственных предприятий. Государственные и отраслевые нормы устанавливают предельно допустимые

концентрации различных загрязнителей, а также требования к устройству и эксплуатации систем вентиляции и кондиционирования. Проведение регулярных аудитов и независимых экспертиз позволяет оценить соответствие условий работы установленным стандартам, выявить слабые места и разработать меры по их устранению. Интеграция данных мониторинга в единую систему управления производственным процессом позволяет своевременно реагировать на изменения в состоянии микроклимата и корректировать работу оборудования, что в конечном итоге способствует повышению общей безопасности производства.

Таким образом, нормализация газового и ионного состава воздуха в производственных помещениях представляет собой многогранную задачу, требующую комплексного подхода. Совокупность технических мер, таких как установка современных вентиляционных систем, использование фильтров с активированным углем и каталитическими элементами, а также применение ионизаторов, в сочетании с организационными мерами, включая регулярный мониторинг, обучение персонала и оптимизацию рабочего пространства, позволяет создавать безопасные условия труда. Соблюдение нормативных требований, проведение регулярных аудитов и интеграция автоматизированных систем управления микроклиматом обеспечивают стабильность и надежность производственного процесса, способствуют снижению риска аварийных ситуаций и улучшают общее состояние здоровья сотрудников.

ПРИЛОЖЕНИЕ А **(обязательное)** **Алгоритм работы пользователя**

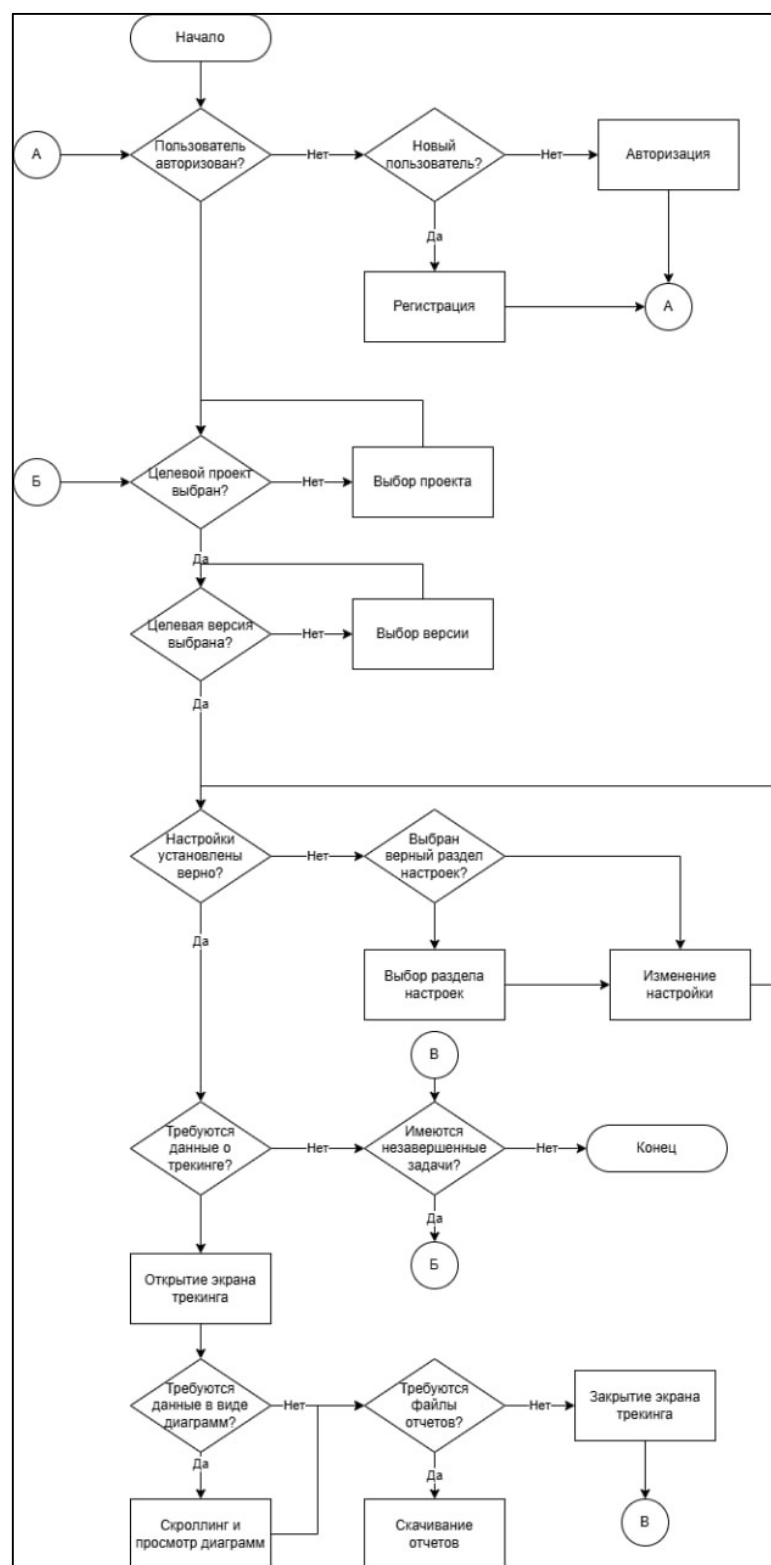


Рисунок А.1 – Алгоритм работы пользователя

ПРИЛОЖЕНИЕ Б

(обязательное)

Эскизы рабочих окон программного комплекса

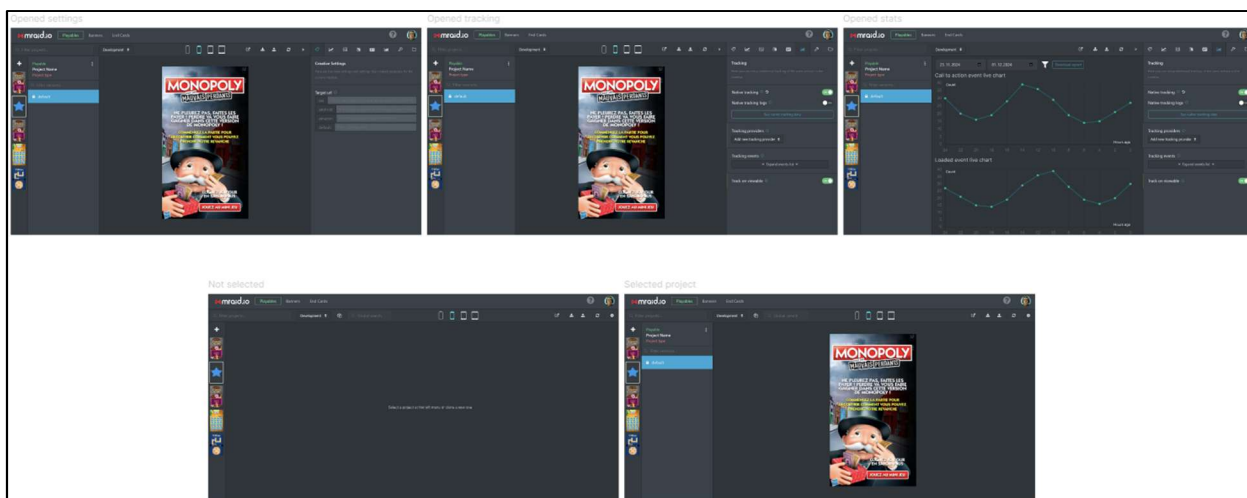


Рисунок Б.1 – Эскизы рабочих окон программного комплекса

ЗАКЛЮЧЕНИЕ

В процессе выполнения дипломного проекта была проведена комплексная работа, направленная на разработку программного комплекса для отслеживания и визуализации действий пользователей в игравельной рекламе. Актуальность выбранной темы обусловлена активным развитием цифровых рекламных технологий, растущими требованиями к точному анализу пользовательской активности и необходимостью предоставления маркетологам и специалистам в сфере рекламы современных инструментов обработки и визуального представления информации.

Разработка велась с учетом современных требований к эргономичности пользовательского интерфейса, производительности системы и гибкости архитектурных решений. Были изучены передовые подходы к проектированию человеко-машинных интерфейсов, тщательно проработаны алгоритмы взаимодействия пользователя с системой, а также реализованы механизмы настройки и визуализации ключевых аналитических данных в режиме реального времени.

Проект охватывает все этапы жизненного цикла программного обеспечения — от предварительного анализа требований и эргономического проектирования до реализации программного комплекса, его тестирования и оценки технико-экономических показателей. Проведенная оценка эргономических свойств системы подтвердила высокий уровень соответствия разработанного интерфейса современным требованиям, что способствует повышению удобства и эффективности работы пользователей.

Кроме того, в рамках дипломной работы была осуществлена детальная проверка работоспособности программного обеспечения, результаты которой показали высокую стабильность, надежность и корректность функционирования всех модулей системы. Программный комплекс продемонстрировал устойчивую работу в различных условиях эксплуатации, а выявленные недочеты были устранены в ходе доработки, что позволило довести продукт до состояния полной готовности к внедрению.

Особое внимание в работе уделено технико-экономическому обоснованию проекта, что позволило не только рассчитать затраты на разработку и сопровождение программного комплекса, но и спрогнозировать его экономическую эффективность. Полученные результаты подтвердили, что использование разработанного программного продукта способно не только оптимизировать процессы анализа взаимодействия с рекламным контентом, но и обеспечить экономическую выгоду при его внедрении и эксплуатации.

Таким образом, можно сделать вывод о том, что поставленные в начале проекта цели и задачи были успешно достигнуты. Разработанный программный комплекс представляет собой современное, функционально завершенное и экономически целесообразное решение, способное повысить

качество и эффективность аналитических процессов в сфере игровой рекламы. Полученные в ходе работы результаты могут быть использованы в дальнейшем при расширении функционала системы, интеграции дополнительных аналитических инструментов и адаптации комплекса под новые направления в области цифрового маркетинга.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Открытые инструменты визуализации данных [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/companies/mws/articles/820959/>
- [2] Как сделать полезный дашборд: советы и идеи [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/companies/tbank/articles/890658/>
- [3] Как правильно визуализировать данные, чтобы принимать эффективные решения? [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/articles/816191/>
- [4] Мониторинговые исследования рекламы в СМИ [Электронный ресурс]. – Режим доступа : <https://cyberleninka.ru/article/n/monitoringovyie-issledovaniya-reklamy-v-smi>
- [5] Все о Playable Ads: от теории к практике с экспертами Мобьо [Электронный ресурс]. – Режим доступа : <https://mobio.ru/blog/vse-o-playable-ads-ot-teorii-k-praktike-s-ehkspertami-mobio/>
- [6] Эволюция Playable Ads: как с помощью игр бренды повышают конверсию и вовлеченность [Электронный ресурс]. – Режим доступа: <https://adindex.ru/publication/opinion/marketing/2023/08/8/315019.phtml>
- [7] Что, Как и Почему в Playable Ads [Электронный ресурс]. – Режим доступа : <https://habr.com/ru/articles/415761/>.
- [8] Как эффективно работать с рекламным форматом Playable ads: пять типичных ошибок [Электронный ресурс]. – Режим доступа : <https://habr.com/ru/companies/appodeal/articles/339106/>
- [9] Google Analytics [Электронный ресурс]. – Режим доступа : <https://marketingplatform.google.com/about/analytics/>
- [10] Mixpanel [Электронный ресурс]. – Режим доступа : <https://mixpanel.com/>
- [11] Amplitude [Электронный ресурс]. – Режим доступа : <https://amplitude.com/>
- [12] Взаимодействие человека и компьютера: Тенденции, исследования, будущее [Электронный ресурс]. – Режим доступа : <https://cyberleninka.ru/article/n/vzaimodeystvie-cheloveka-i-kompyutera-tendentsii-issledovaniya-budushee/viewer>
- [13] Алгоритмы взаимодействия пользователя и интерактивной системы поиска информации [Электронный ресурс]. – Режим доступа : <https://cyberleninka.ru/article/n/algoritmy-vzaimodeystviya-polzovatelya-i-interaktivnoy-sistemy-poiska-informatsii/viewer>
- [14] Требования к разделу по эргономике в дипломных проектах специальности ИПОИТ и рекомендации по его выполнению [Электронный ресурс]. – Режим доступа : https://www.bsuir.by/m/12_100229_1_111674.pdf
- [15] Методология эргономической оценки программного обеспечения в области информационной безопасности [Электронный ресурс]. – Режим

доступа : <https://cyberleninka.ru/article/n/metodologiya-ergonomicheskoy-otsenki-programmnogo-obespecheniya-v-oblasti-informatsionnoy-bezopasnosti/viewer>

[16] WebSocket: разбираем как работает [Электронный ресурс]. –

Режим доступа : <https://habr.com/ru/sandbox/171066/>