

Информационная система «Programmatic-engine»

**ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

на 23 листах

2024 год

СОДЕРЖАНИЕ

1. Список сокращений и обозначений	4
2. Общие сведения	5
3. Назначение программы	6
3.1. Основные назначения программы	6
4. Общие положения	7
5. Цели и автоматизируемые функции	9
5.1. Цели ИС.....	9
6. Функциональные задачи	11
6.1. Управление пользователями	11
6.2. Управление данными	12
6.3. Управление инвентарем	12
6.4. Модерация контента.....	12
6.5. Интеграция с внешними системами	12
6.6. Статистика и аналитика	13
7. Архитектурная структура ИС	14
7.1. Уровни архитектуры.....	14
7.2. Уровень безопасности (Security Layer).....	15
7.3. Диаграмма архитектуры	16
7.4. Межпроцессное взаимодействие ИС.....	17
7.5. Управление аппаратными ресурсами	17
7.6. Обработка внештатных ситуаций и событий.....	18
7.7. Управление LED-индикаторами.....	18
7.8. Управление сетью и учетными записями.....	19
7.9. Управление сертификатами	19
7.10. Интеграция ИС.....	19

8. Применяемые технологии разработки программного обеспечения	21
9. Технические требования	22
9.1. Аппаратные и программные требования к программному обеспечению.....	22
9.2. Минимальный состав технических средств	22
9.3. Минимальный состав программных средств	22
9.4. Численность, функции и квалификация персонала	23

1. Список сокращений и обозначений

SSP — Supply side platform.	сервис programmatic-engine.ru Связующее звено между владельцами инвентаря и DSP.
DSP — Demand side platform.	площадки, продающие рекламу рекламным агентствам и рекламодателям. Обеспечивают взаимодействие конечных покупателей и владельцев инвентаря через SSP.
РК	рекламная кампания.
Рекламодатель	конечный пользователь, заказывающий рекламную кампанию.
Владелец инвентаря	владелец светодиодных рекламных конструкций.
DOOH	Digital out of home. Общее обозначение индустрии цифровой наружной рекламы.
Programmatic реклама.	реклама, размещаемая на условиях аукциона. Слоты для показа продаются по принципу наибольшей цены.
Медиа	видео, изображение, или составной макет. Рекламный креатив, подлежащий показу в рекламном слоте.
Модерация	процесс проверки медиа на соответствие ряду параметров, и определение возможности показа на инвентаре.
Статистика показов	данные с привязкой к инвентарю и медиа, достоверно подтверждающие показ в определенные временные границы.
API	интерфейс получения данных от платформы в соответствии с правилами и фильтрами, назначенными конкретному ключу пользователя.
Плеер	инвентарь
Слот	временной промежуток, предназначенный для трансляции медиа. Закреплен за рекламодателем путем статического набора в плейлист или выигрыша аукциона.
Технические требования	список технических характеристик медиа, которые проверяются при модерации.

2. Общие сведения

Документ содержит описание функциональных характеристик информационной системы «Programmatic-engine» (далее – ИС). ИС является SSP (Supply Side Platform) приложением для работы на рынке DOOH (Digital Out-Of-Home).

ИС предназначена для автоматизации бизнес-процессов в сфере наружной рекламы, включая сбор статистики показов для анализа эффективности, взаимодействие владельцев инвентаря и DSP для генерации статических плейлистов, координации вставки аукционной рекламы, предоставления инструментов модерации контента, контроля статистики показов через аналитические инструменты, а также выдачи данных по API для интеграции с внешними системами и доступа к необходимой информации для анализа и отчетности.

ИС представляет собой интеграционную платформу SSP, предназначенную для объединения различных участников рынка наружной рекламы, включая рекламодателей и владельцев инвентаря, которые могут использовать разные системы управления.

1. Участники рынка

- **Рекламодатели:** Стороны, размещающие рекламу, которые работают через Demand-Side Platforms (далее – DSP) для оптимизации своих рекламных кампаний.
- **Владельцы инвентаря:** Компании, владеющие рекламными местами, которые используют разнообразные системы управления рекламой для учета и контроля показов.

2. Объединение инвентарей

Платформа SSP объединяет инвентарные ресурсы различных владельцев, предоставляя рекламодателям централизованный доступ к рекламным местам. Это упрощает процесс размещения рекламы и управления кампаниями, позволяя рекламодателям выбирать наиболее эффективные рекламные места.

ИС обладает пользовательским интерфейсом и API, предназначенными для профессиональных участников индустрии DOOH. ИС не является публичной, так как ориентирована на узкий круг пользователей, что делает ее более целевой и эффективной для выполнения специфических задач. ИС обеспечивает доступность из любого места и на любом устройстве, в зависимости от роли пользователя.

3. Назначение программы

ИС является специализированной платформой SSP для работы на рынке DOOH. ИС предназначена для автоматизации и оптимизации бизнес-процессов, связанных с размещением наружной рекламы, и ориентирована на профессиональных участников этой индустрии. Поскольку ИС не является публичной, она предлагает узкоспециализированные инструменты, что повышает ее эффективность в выполнении специфических задач.

3.1. Основные назначения программы

1) Сбор статистики показа рекламы на инвентаре:

- Обеспечивает мониторинг и анализ эффективности рекламных показов, что позволяет владельцам инвентаря оценивать успешность своих рекламных кампаний и вносить необходимые коррективы.

2) Взаимодействие владельцев инвентаря и DSP:

- Способствует генерации статических плейлистов, упрощая планирование рекламных кампаний и обеспечивая более точное размещение контента в зависимости от целевой аудитории.

3) Вставка аукционной рекламы:

- Позволяет владельцам инвентаря и DSP координировать размещение аукционных объявлений, что способствует увеличению доходов от рекламы и оптимизации использования рекламных площадок.

4) Инструменты модерации для владельцев инвентаря:

- Обеспечивает возможность контроля и редактирования контента, отображаемого на инвентаре, что позволяет поддерживать высокое качество рекламы и соответствие требованиям.

5) Контроль статистики показов:

- Предоставляет владельцам инвентаря аналитические инструменты для отслеживания результатов рекламных кампаний, что позволяет принимать обоснованные решения на основе данных.

6) Выдача данных по API заинтересованным лицам:

- Обеспечивает интеграцию с внешними системами, позволяя пользователям получать необходимые данные для анализа и отчетности, что улучшает взаимодействие между различными платформами и участниками рынка.

4. Общие положения

ИС предназначена для автоматизации процессов, связанных с размещением наружной рекламы. ИС обеспечивает интеграцию различных участников рынка DOOH, включая владельцев инвентаря и DSP. Основной задачей ИС является повышение эффективности рекламных кампаний через оптимизацию процессов сбора, анализа и управления рекламным контентом.

1) Основные характеристики

- **Интуитивно понятный интерфейс:** ИС разработана с акцентом на удобство использования, что позволяет пользователям быстро осваивать её функционал.
- **Модульная архитектура:** ИС состоит из различных модулей, каждый из которых отвечает за определённый аспект работы, что обеспечивает гибкость и возможность масштабирования.
- **Интеграция с внешними сервисами:** ИС поддерживает интеграцию через API, что позволяет обмениваться данными с другими платформами и системами.
- **Обеспечение безопасности:** внедрены механизмы аутентификации и авторизации пользователей, что гарантирует защиту данных и контроль доступа к функционалу ИС.

2) Пользователи системы

ИС предназначена для использования следующими категориями пользователей:

- **Владельцы инвентаря:** управляют рекламными площадками и контролируют размещение рекламы.
- **DSP:** участвуют в закупке рекламных мест и управлении аукционными объявлениями.
- **Аналитики:** ответственные за анализ статистики и эффективности рекламных кампаний, предоставляют рекомендации по оптимизации.

3) Принципы работы

Система функционирует на основе следующих принципов:

- **Автоматизация процессов:** все основные бизнес-процессы автоматизированы, что снижает временные затраты и вероятность ошибок.
- **Доступность данных:** обеспечивается оперативный доступ к актуальной информации через интерфейс пользователя и API.
- **Гибкость и адаптивность:** ИС способна адаптироваться к изменениям в бизнес-требованиях и технических условиях, обеспечивая долгосрочную актуальность.

4) Ограничения и условия эксплуатации

При использовании системы могут возникать следующие ограничения:

- **Требования к инфраструктуре:** необходимость в наличии определённого программного обеспечения и аппаратного обеспечения для корректной работы системы.

- **Лимиты на объем данных:** ограничения по количеству одновременно обрабатываемых данных, зависящие от конфигурации системы и используемых ресурсов.

- **Роли и права доступа:** доступ к определённым функциям ИС может быть ограничен в зависимости от роли пользователя, что необходимо для обеспечения безопасности и контроля.

5. Цели и автоматизируемые функции

5.1. Цели ИС

Основная цель ИС — обеспечение эффективных показов наружной рекламы. Автоматизация процессов, перечисленных в разделе «Назначение программы», позволяет значительно сократить временные затраты, повысить точность операций и улучшить взаимодействие между всеми участниками рынка DOOH. Это не только улучшает общую производительность, но и способствует созданию более прозрачной и отзывчивой рекламной экосистемы.

1) Автоматизация управления пользователями:

- Обеспечение простой регистрации и аутентификации пользователей, что позволяет быстро и безопасно управлять учетными записями.
- Настройка ролей и прав доступа для обеспечения индивидуальных настроек и защиты данных, что повышает безопасность и контроль за действиями пользователей.
- Поддержка ведения логов и аудита, что позволяет отслеживать активность пользователей и анализировать безопасность системы.

2) Оптимизация управления данными:

- Централизованное хранение и управление данными, что упрощает доступ к информации и повышает эффективность работы с ней.
- Реализация функций импорта и экспорта данных, что облегчает интеграцию с внешними источниками и системами, а также позволяет пользователям работать с необходимой информацией в удобных форматах.
- Обработка и очистка данных для повышения их качества и достоверности, что способствует более точным анализам и отчетам.

3) Управление инвентарем рекламных мест:

- Предоставление владельцам рекламных площадок инструментов для эффективного управления доступными рекламными местами, что позволяет максимизировать их использование.
- Обеспечение функций добавления, редактирования и удаления рекламных мест, а также управления ценовыми моделями и доступностью, что увеличивает гибкость в управлении инвентарем.

4) Контроль качества контента:

- Возможность модерации рекламного контента перед его размещением для обеспечения соответствия установленным требованиям и стандартам.

- Функции редактирования и утверждения рекламы, что повышает качество и надежность размещаемых материалов.

5) Интеграция с внешними системами:

- Обеспечение взаимодействия с другими платформами и сервисами через API, что расширяет функциональность системы и улучшает интеграцию с CRM, CMS и аналитическими инструментами.

- Упрощение обмена данными между различными системами, что способствует более эффективному взаимодействию в рамках бизнес-процессов.

6) Анализ и отчетность:

- Генерация отчетов по заданным критериям, что позволяет пользователям получать необходимые данные для принятия обоснованных решений.

- Инструменты для анализа данных и подсчета статистики, что способствует более глубокому пониманию бизнес-процессов и повышению их эффективности.

6. Функциональные задачи

Функциональные задачи направлены на создание эффективной системы, которая будет способствовать автоматизации, оптимизации и улучшению качества бизнес-процессов в сфере размещения наружной рекламы.

ИС выполняет следующие функциональные задачи:

- 1) Управление пользователями;
- 2) Управление данными;
- 3) Управление заявками;
- 4) Обработка договоров;
- 5) Формирование отчетности и аналитики;
- 6) Настройка уведомлений;
- 7) Управление продажами;
- 8) Авторизация и безопасность;
- 9) Управление территориями.

6.1. Управление пользователями

- **Регистрация и аутентификация:** обеспечивает создание и управление учетными записями пользователей, а также проверку их подлинности для доступа к системе.

- **Управление ролями и правами доступа:** настройка разрешений для различных ролей пользователей, что позволяет контролировать доступ к функциям и данным ИС.

- **Профили пользователей:** управление личной информацией и настройками пользователей для обеспечения персонализированного опыта.

- **Логи и аудит:** мониторинг активности пользователей с записью действий для анализа и повышения безопасности ИС.

6.2. Управление данными

- **Хранение данных:** централизованное хранение и управление данными для упрощения доступа и обработки информации.
- **Импорт и экспорт данных:** возможность загрузки данных из внешних источников и выгрузки в различные форматы, что облегчает интеграцию с другими системами.
- **База данных:** использование реляционных баз данных для надежного хранения информации.
- **Обработка и очистка данных:** обработка, очистка и верификация данных для повышения их качества и актуальности.

6.3. Управление инвентарем

- **Управление рекламными площадками:** позволяет владельцам управлять доступными рекламными местами, обеспечивая удобство и эффективность.
- **Функции управления:** возможность добавления, редактирования и удаления рекламных мест, установки ценовых моделей и управления их доступностью.

6.4. Модерация контента

- **Контроль рекламного контента:** позволяет проверять и редактировать рекламный контент перед его размещением для соответствия установленным требованиям.
- **Редактирование и утверждение:** обеспечивает возможность редактирования и утверждения рекламы, что повышает качество и соответствие стандартам.

6.5. Интеграция с внешними системами

- **Взаимодействие с другими платформами:** обеспечивает интеграцию с различными системами и сервисами через API.
- **Поддержка различных интеграций:** включает интеграцию с CRM, CMS, аналитическими инструментами и другими системами, что улучшает функциональность.

6.6. Статистика и аналитика

- **Создание отчетов:** генерация отчетов по заданным критериям, что позволяет пользователям получать необходимые данные для анализа.
- **Анализ данных:** инструменты для анализа данных и подсчета статистики, что помогает в принятии обоснованных решений на основе собранной информации.

7. Архитектурная структура ИС

Архитектура ИС представляет собой структуру, которая определяет организацию и взаимодействие всех ее компонентов и слоев. Основная цель архитектуры — обеспечить масштабируемость, надежность и гибкость ИС, а также интеграцию различных бизнес-процессов и технологий.

7.1. Уровни архитектуры

1) Уровень пользовательского интерфейса (Frontend)

- **Веб-интерфейс:** основной интерфейс для пользователей, доступный через веб-браузеры. Включает интерфейсы для работы с заявками, договорами, отчетами и другими функциями.
- **Интерфейсы API:** возможность интеграции с внешними приложениями и системами через API.

2) Уровень бизнес-логики (Backend)

- **Сервер приложений:** обработка бизнес-логики и выполнение основных задач системы, таких как управление заявками, обработка заказов и генерация отчетов.
- **Модуль управления пользователями:** управление учетными записями, аутентификацией и авторизацией.
- **Модуль управления данными:** обработка и управление данными, включая их очистку, верификацию и интеграцию с внешними источниками.

3) Уровень хранения данных (Data Storage)

- **Базы данных:** хранение данных в реляционных (SQL) базах данных. Используются для хранения информации о пользователях, заявках, договорах и т.д.
- **Файловое хранилище:** системы для хранения файлов и документов, таких как договоры, отчеты и другие прикрепленные файлы.
- **Резервное копирование и восстановление:** механизмы для создания резервных копий данных и восстановления в случае сбоя.

7.2. Уровень безопасности (Security Layer)

- **Аутентификация и авторизация:** механизмы для проверки учетных данных пользователей и управления правами доступа.

- **Шифрование данных:** защита данных при передаче и хранении.

- **Мониторинг безопасности:** отслеживание и реагирование на потенциальные угрозы и нарушения безопасности.

1) Уровень поддержки и обслуживания (Support Layer):

- **Система поддержки:** обработка запросов и проблем пользователей.

- **Обновления:** процедуры для регулярного обновления системы и исправления ошибок.

- **Документация:** руководства пользователя, техническая документация и обучающие материалы.

2) Ключевые компоненты структуры ИС

а. Стек LAMP

LAMP — это программный стек, состоящий из следующих компонентов:

- **Linux:** операционная система, обеспечивающая стабильную и безопасную среду для работы программного обеспечения. Linux предоставляет необходимые инструменты и возможности для управления сервером, сетевыми ресурсами и процессами.

- **Apache:** веб-сервер, который обрабатывает входящие HTTP-запросы и отвечает за доставку веб-страниц пользователям. Apache конфигурируется для работы с динамическими и статическими контентом, поддерживая различные модули для повышения производительности и безопасности.

- **MySQL:** система управления базами данных, используемая для хранения и обработки данных ИС. MySQL обеспечивает надежное хранение, быстрый доступ и выполнение сложных запросов, что позволяет эффективно управлять данными, необходимыми для работы ИС.

- **PHP:** язык программирования, на котором разрабатывается серверная логика ИС. PHP позволяет создавать динамические веб-страницы, обрабатывать формы и взаимодействовать с базой данных, обеспечивая функциональность и гибкость ИС.

в. Фреймворк Laravel

- **Laravel** — это современный PHP-фреймворк, который используется для разработки веб-приложений. Он предлагает множество инструментов и функций, способствующих ускорению процесса разработки и улучшению структуры кода.
- **MVC-архитектура:** Laravel основан на паттерне Model-View-Controller (MVC), что способствует разделению логики ИС на три взаимосвязанные части. Это упрощает разработку, тестирование и поддержку кода.
- **Маршрутизация:** Laravel предлагает удобный механизм маршрутизации, позволяющий определять, как запросы от пользователей будут обрабатываться и какие контроллеры будут вызываться.
- **ORM (Eloquent):** встроенный объектно-реляционный маппер (ORM) Eloquent упрощает работу с базой данных, позволяя разработчикам использовать простые и понятные методы для выполнения запросов.
- **Безопасность:** Laravel включает множество средств для обеспечения безопасности ИС, таких как защита от SQL-инъекций, межсайтовых скриптов (XSS) и других угроз.
- **Миграции и сиды:** Laravel предоставляет механизмы для управления структурой базы данных с помощью миграций и начальных данных с помощью сидов, что упрощает процесс развертывания и поддержки базы данных.

7.3. Диаграмма архитектуры

Диаграмма архитектуры представляет взаимодействие между компонентами и уровнями ИС.

Основные элементы диаграммы включают:

- **Пользовательский интерфейс:** веб-приложения.
- **Сервер приложений:** обработка бизнес-логики.
- **База данных:** хранение и управление данными.
- **API и интеграционные шлюзы:** взаимодействие с внешними системами.
- **Системы безопасности:** аутентификация, шифрование, мониторинг.
- **Файловое хранилище:** документы, резервные копии.



Рисунок 2 – Диаграмма архитектуры ИС

7.4. Межпроцессное взаимодействие ИС

Модуль межпроцессного взаимодействия обеспечивает эффективную коммуникацию между различными компонентами системы через веб-технологии. Он основан на использовании протоколов HTTP/HTTPS и RESTful API, что позволяет компонентам системы обмениваться данными и выполнять удаленные вызовы функций.

Основные функции:

- **API для взаимодействия:** предоставляет интерфейсы для взаимодействия между модулями для облегчения интеграции и обмена данными.
- **Синхронные и асинхронные запросы:** поддержка различных типов запросов позволяет управлять потоками данных в реальном времени и выполнять задачи в фоновом режиме.
- **Безопасность:** реализованы механизмы аутентификации и авторизации, что обеспечивает защиту данных и контроль доступа к ресурсам.

7.5. Управление аппаратными ресурсами

Управление аппаратными ресурсами ИС осуществляется с использованием возможностей операционной системы Linux и фреймворка Laravel.

Основные функции:

- **Доступ к системным ресурсам:** использование встроенных функций PHP для взаимодействия с системными вызовами Linux, что позволяет управлять процессами и ресурсами на уровне ОС.

- **Управление конфигурацией:** настройка аппаратных ресурсов (например, процессоров, памяти) через конфигурационные файлы и команды Linux.
- **Мониторинг состояния ресурсов:** использование встроенных средств для отслеживания загрузки процессоров, использования памяти и других показателей.

7.6. Обработка внештатных ситуаций и событий

Управление внештатными ситуациями и событиями, возникающими в процессе работы ИС. Он основан на встроенных средствах Linux, PHP и Laravel.

Основные функции:

- **Обработка исключений:** реализация механизмов для перехвата и обработки ошибок, что позволяет системе оставаться устойчивой в случае возникновения непредвиденных обстоятельств.
- **Логирование событий:** использование встроенных средств для записи событий и ошибок в журналы, что упрощает диагностику и анализ проблем.
- **Уведомления:** настройка системы уведомлений для информирования администраторов о критических событиях или сбоях в работе.

7.7. Управление LED-индикаторами

Модуль управления LED-индикаторами отвечает за контроль и настройку состояния светодиодных индикаторов, используемых в системе для визуального отображения информации и статусов.

Основные функции

- **Управление состоянием индикаторов:** использование встроенных средств Linux и PHP для изменения состояния LED-индикаторов (включение, выключение, изменение цвета) в зависимости от определенных событий или условий.
- **Настройка режимов работы:** возможность настройки различных режимов отображения (мигание, постоянное свечение и т.д.) через интерфейсы, разработанные на базе Laravel.
- **Мониторинг состояния:** отслеживание текущего состояния индикаторов и ведение журналов их работы для анализа и диагностики.

7.8. Управление сетью и учетными записями

Управление сетью и учетными записями обеспечивает управление сетевыми настройками и учетными записями пользователей системы.

Основные функции

- **Настройка сетевых параметров:** использование встроенных средств Linux для управления IP-адресами, маршрутизацией и другими сетевыми настройками.
- **Управление учетными записями пользователей:** возможность создания, редактирования и удаления учетных записей через интерфейс администратора, реализованный на Laravel.
- **Контроль доступа:** реализация механизмов аутентификации и авторизации для ограничения доступа к различным функциям ИС в зависимости от ролей пользователей.

7.9. Управление сертификатами

Управление сертификатами отвечает за работу с цифровыми сертификатами, необходимыми для обеспечения безопасности соединений и аутентификации.

Основные функции

- **Создание и хранение сертификатов:** использование встроенных средств Linux для генерации, установки и хранения SSL/TLS сертификатов.
- **Обновление и истечение сертификатов:** мониторинг сроков действия сертификатов и автоматическое обновление при необходимости.
- **Настройка конфигурации:** управление настройками, связанными с безопасностью, через интерфейсы на PHP и Laravel, чтобы обеспечить корректную работу с сертификатами.

7.10. Интеграция ИС

Интеграция обеспечивает взаимодействие системы с внешними сервисами и ИС через API.

Основные функции

- **REST API:** реализация RESTful API для обмена данными между системой и внешними приложениями, что позволяет легко интегрировать функциональность и данные.
- **Документация API:** предоставление детальной документации по API, включая методы, параметры и примеры использования, что упрощает интеграцию для разработчиков.
- **Безопасность взаимодействия:** реализация механизмов аутентификации для защиты API и контроля доступа к данным и функциям системы.

8. Применяемые технологии разработки программного обеспечения

1) Фреймворк Laravel

Laravel — это фреймворк для разработки веб-приложений на PHP, который обеспечивает удобный и элегантный синтаксис. Он поддерживает современные практики программирования, такие как MVC (Model-View-Controller), что позволяет структурировать код и облегчает его сопровождение. Laravel также предоставляет множество встроенных инструментов, таких как:

- **ORM Eloquent:** для удобной работы с базами данных.
- **Маршрутизация:** простое определение маршрутов и обработка HTTP-запросов.
- **Безопасность:** встроенные механизмы защиты от SQL-инъекций, CSRF и XSS.
- **Миграции:** управление изменениями в структуре базы данных.

2) Облачный сервер Rusonux

Облачный сервер от Rusonux предоставляет надежную и масштабируемую инфраструктуру для развертывания веб-приложений. Преимущества использования облачного сервера:

- **Масштабируемость:** возможность увеличивать ресурсы по мере роста нагрузки.
- **Безопасность:** современные системы защиты данных и резервного копирования.
- **Высокая доступность:** минимальные простои и надежная работа ИС.

9. Технические требования

ИС функционирует при его запуске в веб-браузере. Все функции ИС доступны после ее запуска.

9.1. Аппаратные и программные требования к программному обеспечению

Этот раздел документа описывает минимальные аппаратные и программные требования, необходимые для успешной работы ИС, а также минимальный состав технических и программных средств, а также квалификацию персонала.

Аппаратные требования:

- **Процессор:** необходимо наличие как минимум одного процессора, обеспечивающего достаточную производительность для обработки запросов и выполнения задач системы.
- **Оперативная память (ОЗУ):** рекомендуется минимум 2 ГБ ОЗУ для обеспечения стабильной работы ИС и обработки многозадачности.
- **Накопитель:** необходим SSD объемом не менее 100 ГБ, что обеспечивает быструю загрузку и выполнение операций с данными.
- **Выделенный IP:** требуется выделенный IP-адрес для обеспечения стабильного и безопасного доступа к системе.
- **Ассоциирование имени и IP:** поддержка ассоциирования доменного имени с IP-адресом для удобства доступа к ИС.

9.2. Минимальный состав технических средств

Облачный сервер Rusonyx: рекомендуется использовать облачный сервер Rusonyx, который соответствует перечисленным выше аппаратным требованиям. Это гарантирует надежность, масштабируемость и доступность ресурсов.

9.3. Минимальный состав программных средств

Операционная система: Linux — рекомендуется использование стабильной версии, поддерживающей необходимые пакеты и зависимости.

- **LAMP:** установка стека LAMP (Linux, Apache, MySQL, PHP) для обеспечения работы веб-приложений.
- **Фреймворк:** Laravel — использование фреймворка Laravel для разработки и поддержки программного обеспечения, что обеспечивает быструю разработку и простоту поддержки.

9.4. Численность, функции и квалификация персонала

- **Численность:** минимально необходим один уверенный пользователь ПК для администрирования и работы с программным обеспечением.
- **Функции:** персонал должен быть способен выполнять следующие задачи:
 - Установка и настройка программного обеспечения.
 - Мониторинг работы системы и устранение неполадок.
 - Обновление и поддержка программных компонентов.
- **Квалификация:** необходимы базовые знания работы с ПК, умение работать с командной строкой Linux, а также навыки работы с веб-приложениями и базами данных.