

УТВЕРЖДЕН
RU.07622667.ПМП.001.ПК-ЛУ

ИНФОРМАЦИОННАЯ СИСТЕМА ЮМТ.ИСТОК

Описание архитектуры
RU.07622667.ПМП.001.ПК

Листов 14

Инва. № подл.	Подп. и дата
Взам. инв. №	Инва. № дубл.
Подп. и дата	Подп. и дата

2024

Литера

АННОТАЦИЯ

Настоящий документ содержит описание архитектуры Информационной системы IoMT.Istok (далее – система), которая представляет собой передовую российскую платформу для медицинского интернета вещей и содержит полный набор инструментов для дистанционного мониторинга состояния пациентов с возможностью использования полученной информации на всех этапах оказания медицинской помощи. Система предназначена для организации единого информационного пространства для сбора, предварительной обработки и передачи данных дистанционного мониторинга в медицинские информационные системы медицинских организаций (далее – МИС МО).

Описание архитектуры содержит описание системы, сведения о функциональных компонентах, описание особенностей технической реализации системы.

СОДЕРЖАНИЕ

Введение.....	4
1 Описание системы.....	5
1.1 Общая структура программного обеспечения	5
1.2 Архитектурные решения и выбор технологий.....	5
2 Архитектура системы	6
3 Функциональные компоненты системы	7
3.1 Подсистема «Портал»	7
3.2 Подсистема обмен данными	7
3.3 Подсистема журналирования событий	7
3.4 Подсистема взаимодействия с устройствами ПМП	7
3.5 Подсистема взаимодействия с МИС МО.....	8
3.6 Подсистема обработки данных.....	8
3.7 Подсистема хранение данных.....	9
3.8 Подсистема аналитики и отчетности	9
4 Структурные элементы системы и их интерфейсы	10
4.1 Описание технических аспектов реализации компонентов.....	10
4.2 Перечень используемых технологий.....	10
Перечень сокращений и терминов	13

ВВЕДЕНИЕ

Информационные системы с открытым исходным кодом пользуются популярностью благодаря своей гибкости, безопасности и надежности. Такая модель распространения программных продуктов позволяет пользователям свободно использовать, изменять и распространять программное обеспечение.

Информационная система IoT.Istok построена с использованием микросервисной архитектуры. Это подход, при котором приложение разбивается на небольшие автономные модули – микросервисы, каждый из которых отвечает за определенную функциональность. Такая архитектура обладает рядом преимуществ, включая гибкость, масштабируемость, упрощение развертывания и поддержки, а также улучшение надежности всей системы.

Использование микросервисной архитектуры позволяет легко добавлять, изменять и масштабировать отдельные компоненты системы, не затрагивая работу других микросервисов. Это способствует легкости разработки, обновления и поддержки системы в целом.

Таким образом, микросервисная архитектура, в сочетании с открытым исходным кодом, кластерной архитектурой и некоммерческой лицензией программного обеспечения, делает информационную систему IoT.Istok эффективной, надежной и легкой в обслуживании.

1 ОПИСАНИЕ СИСТЕМЫ

1.1 Общая структура программного обеспечения

Программное обеспечение состоит из нескольких функциональных подсистем, каждая из которых отвечает за определенные аспекты работы системы. Система включает в себя следующие подсистемы:

- Подсистема «Портал»;
- Подсистема обмена данными;
- Подсистема обработки данных;
- Подсистема хранения данных;
- Подсистема аналитики и отчетности;
- Подсистема журналирования событий;
- Подсистема взаимодействия с устройствами ПМП;
- Подсистема взаимодействия с МИС МО.

Наименование микросервисов реализации подсистем:

- IoT.Istok Gateway;
- IoT.Istok MIS;
- IoT.Istok Devices;
- IoT.Istok Portal;
- IoT.Istok Websocket;
- IoT.Istok Logs.

1.2 Архитектурные решения и выбор технологий

Вся архитектура построена с использованием микросервисной архитектуры, где каждый отдельный микросервис, ответственный за определенные функции, реализует функции подсистемы. Для обеспечения интеграции между подсистемами используется принцип «поставщик-потребитель». В процессе реализации использованы различные технологии, такие как Apache Tomcat, Docker, PostgreSQL, Keycloak и другие, чтобы обеспечить требуемую функциональность и производительность.

2 АРХИТЕКТУРА СИСТЕМЫ

Схема архитектуры системы приведена на рисунке 1.

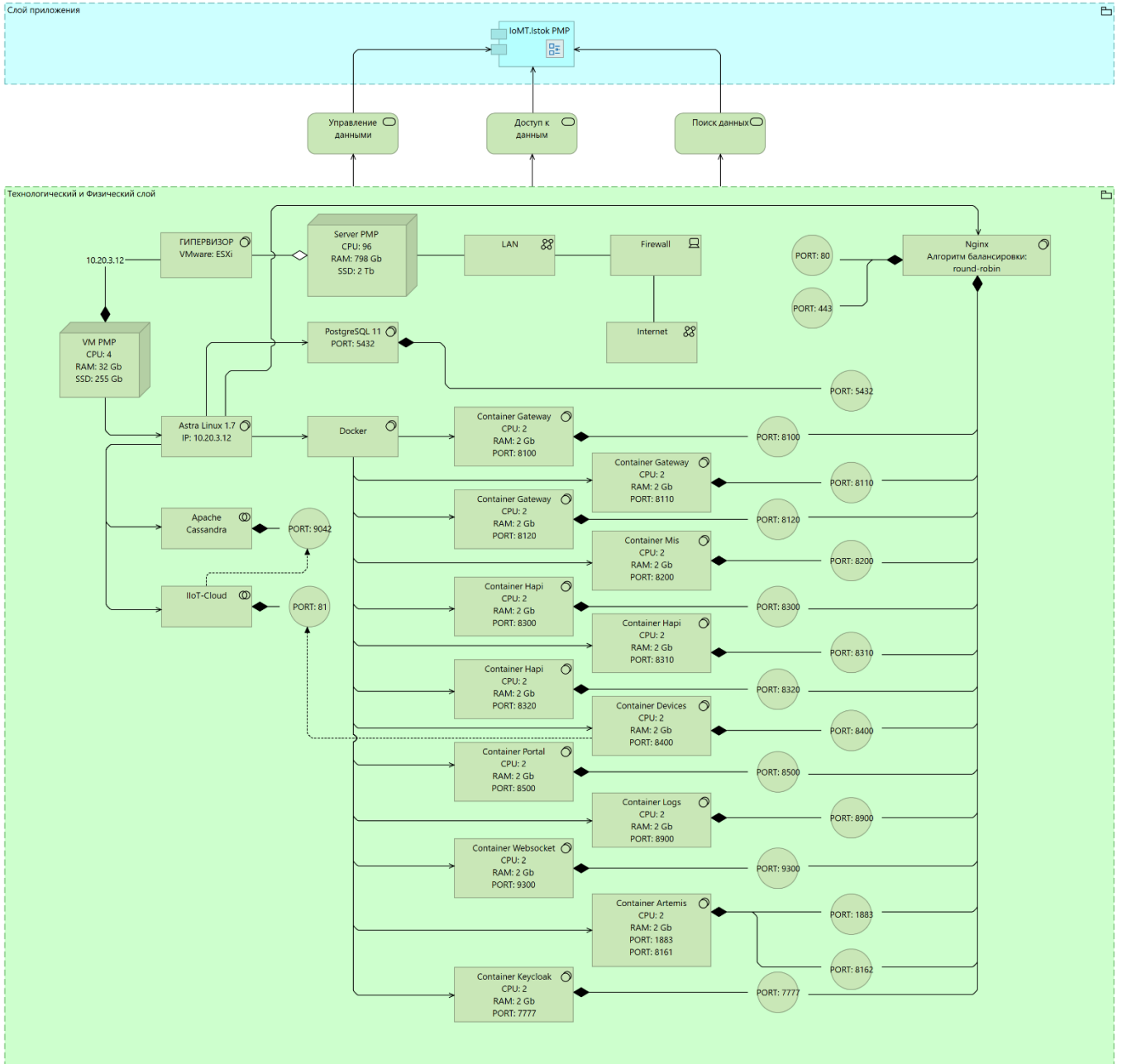


Рисунок 1 – Схема архитектуры системы

3 ФУНКЦИОНАЛЬНЫЕ КОМПОНЕНТЫ СИСТЕМЫ

3.1 Подсистема «Портал»

Подсистема «Портал» состоит из микросервиса IoMT.Istok Portal, который обеспечивает выполнение следующих функций:

- регистрация устройств ПМП;
- удаление устройств ПМП;
- создание шаблонов;
- создание медицинских организаций;
- регистрация производителей;
- добавление регионов;
- добавление новых пользователей;
- просмотр пороговых уровней
- просмотр информации по программам мониторинга.

3.2 Подсистема обмен данными

Подсистема обмена данными состоит из микросервисов IoMT.Istok MIS, IoMT.Istok Gateway, IoMT.Istok Websocket, IoMT.Istok Devices, которые обеспечивают выполнение следующих функций:

- прием данных от устройств ПМП;
- предоставление данных и результатов их обработки участникам взаимодействия;
- идентификация входящих пакетов от источников информации.

3.3 Подсистема журналирования событий

Подсистема журналирования событий состоит из микросервиса IoMT.Istok Logs, который обеспечивает выполнение следующих функций:

- прием данных со всех подсистем и микросервисов системы;
- прием журналов инцидентов с устройств ПМП;
- передача данных в подсистему аналитики и отчетности.

3.4 Подсистема взаимодействия с устройствами ПМП

Подсистема взаимодействия с устройствами ПМП состоит из микросервиса IoMT.Istok Gateway, который обеспечивает выполнение следующих функций:

- получение данных от устройств ПМП типа тонометр: показания систолического давления, показания диастолического давления, частота сердечных сокращений;
- получение данных от устройств ПМП типа глюкометр: уровень глюкозы в крови.

3.5 Подсистема взаимодействия с МИС МО

Подсистема взаимодействия с МИС МО состоит из микросервисов IoMT.Istok MIS, IoMT.Istok WebSocket, которые обеспечивают выполнение следующих функций:

- передача данных, полученных от МИС МО, в подсистему хранения данных;
- обновление ранее сохраненных программ мониторинга;
- предоставление данных по API по запросу МИС МО;
- отправка уведомлений в МИС МО при получении от подсистемы обработки данных измерения, входящего в диапазон, определённый настройкой правил реагирования на инциденты, при наличии подписки на них запрошенной от МИС МО по каналу оперативного оповещения, WebSocket;
- отправка получаемых данных в рамках соответствующих заказов на дистанционный мониторинг в МИС МО в API МИС МО в соответствии с условиями подписки, если МИС МО зарегистрировал соответствующую подписку по каналу оперативного оповещения, WebSocket;
- идентификация устройства ПМП по Идентификатору устройства, при поступлении заказа на дистанционный мониторинг из МИС МО.
- интеграция и обмен сообщениями между подсистемами системы;
- взаимодействие между интегрируемыми подсистемами и микросервисами по принципу «поставщик-потребитель»;
- возможность сохранения передаваемых сообщений во внутреннем хранилище данных, для обеспечения их гарантированной доставки;
- гарантированная доставка передаваемых сообщений;
- направления результатов ФЛК в МИС;
- прием информации о данных врача, оказывающего медицинскую услугу от МО.

3.6 Подсистема обработки данных

Подсистема обработка данных состоит из микросервиса IoMT.Istok MIS, который обеспечивает выполнение следующих функций:

- получение данных от подсистемы обмена данными;
- первичная идентификация пациента по информации из подсистемы хранения данных;
- обработка поступающих данных с учетом параметров, указанных в программе

мониторинга;

- выполнение потоковой аналитики для определения отклонения значений показателей относительно заданных пороговых значений;
- обработка результатов ФЛК и передача их в подсистему взаимодействия МИС МО;
- прием информации для верификации о зарегистрированных МО;
- прием информации о данных врача, оказывающего медицинскую услугу от подсистемы взаимодействия с МИС МО.

3.7 Подсистема хранение данных

Подсистема хранение данных состоит из СУБД PostgreSQL и микросервиса IoMT.Istok Napі, которые обеспечивают выполнение следующих функций:

- хранение данных;
- предоставление данных другим подсистемам и микросервисами системы.

3.8 Подсистема аналитики и отчетности

Подсистема аналитики и отчетности состоит из микросервиса IoMT.Istok Portal, который обеспечивает выполнение следующих функций:

- отображение информационной панели ключевых параметров функционирования системы;
- отображение ключевых параметров в разрезе регионов и медицинских организаций.

4 СТРУКТУРНЫЕ ЭЛЕМЕНТЫ СИСТЕМЫ И ИХ ИНТЕРФЕЙСЫ

4.1 Описание технических аспектов реализации компонентов

Архитектура данного проекта разработана в соответствии с микросервисным подходом. Это означает, что каждый функциональный элемент системы реализуется в виде отдельного сервиса, способного независимо масштабироваться в соответствии с требованиями. Для обеспечения взаимодействия между данными сервисами применяются API запросы с контролем переправки при возникновении ошибок.

В качестве решения для надежного хранения данных используется система управления базами данных PostgreSQL. Это реляционная СУБД с открытым исходным кодом, обеспечивающая ACID-соблюдение и поддержку широкого спектра функциональных расширений. Для хранения объемных данных, требующих высокую доступность, применяется NoSQL-система PoT.Istok Cloud, обладающая превосходной масштабируемостью и отказоустойчивостью.

В процессе развертывания компонентов системы используются виртуальные машины, созданные на базе виртуализации Альт Сервер Виртуализации 10. Это обеспечивает изоляцию экземпляров системы друг от друга, что улучшает безопасность, а также позволяет более гибко управлять распределением системных ресурсов.

4.2 Перечень используемых технологий

На стороне сервера (backend) применяется язык программирования Java (version 11, 17, 21) в сочетании с фреймворком Spring Boot (v.2.7.10, v 3.1.2). Это позволяет организовать эффективную обработку HTTP/HTTPS -запросов, управление бизнес-логикой проекта и взаимодействие с базами данных.

Для создания пользовательского интерфейса на стороне клиента (frontend) используются технологии React и Bootstrap. React обеспечивает разработку динамического интерфейса с высокой производительностью, в то время как Bootstrap используется для реализации адаптивности и визуальной привлекательности интерфейса. Генерация динамических HTML-страниц осуществляется с помощью шаблонизатора Freemarker.

Для защиты данных, передаваемых между компонентами и внешними системами, применяется протокол HTTPS, который исключает возможность перехвата информации посторонними лицами. Использование SSL-сертификатов увеличивает уровень доверия к системе, обеспечивая подтверждение подлинности связи и шифрование передаваемых данных.

Для централизованного управления доступом и авторизации в проекте используется Keycloak. Keycloak – это открытая платформа, которая позволяет удобно управлять идентификацией пользователей, обеспечивая безопасный вход в систему через единый интерфейс. Keycloak применяется для интеграции с внешними системами, обеспечивая аутентификацию и авторизацию пользователей, поддерживая при этом стандарты безопасности OAuth 2.0, OpenID Connect и SAML. Это обеспечивает не только безопасный доступ пользователей к различным частям системы, но и возможность интеграции с корпоративными системами управления доступом.

Использование Keycloak в проекте дает следующие преимущества:

- упрощение процесса аутентификации с помощью единого входа (Single Sign-On), что улучшает пользовательский опыт;
- гибкое управление пользовательскими ролями и доступами, что усиливает безопасность приложения;
- возможность работы с различными протоколами идентификации и безопасности, что упрощает взаимодействие с внешними API и сервисами.

Такое решение позволяет не только повысить удобство для конечных пользователей, но и обеспечивает комплексный подход к безопасности на всех уровнях взаимодействия с системой.

WebSocket – это технология, предоставляющая двусторонний канал обмена сообщениями между клиентом и сервером в реальном времени по протоколу TCP. Основное преимущество WebSocket состоит в том, что после установления соединения сервер может инициировать передачу данных к клиенту в любой момент, что является отличием от классического HTTP, где данные могут быть отправлены только в ответ на запрос клиента.

В контексте реализации данного проекта интеграция WebSocket позволяет:

- обеспечить немедленную доставку данных или событий пользователям, что особенно полезно для функционалов, требующих мгновенного уведомления клиентов, в том числе для чатов, информационных панелей с мониторингом данных в реальном времени и других подобных задач;
- снизить количество запросов к серверу за счет поддержания постоянного соединения, вместо многократных HTTP запросов, что уменьшает нагрузку на сервер и улучшает производительность системы в целом;
- увеличить скорость обмена данными, так как WebSocket устанавливает постоянное соединение и избавляется от накладных расходов HTTP, связанных с постоянным открытием и закрытием соединений.

Система управления контейнерами Docker – это открытая платформа для разработки, доставки и запуска приложений. Он позволяет отделить приложения от инфраструктуры и управлять инфраструктурой по аналогии с тем, как мы управляем приложениями.

Docker предоставляет возможность упаковывать и запускать приложение в изолированной среде-контейнере. Изоляция и безопасность позволяют одновременно запускать несколько контейнеров на одном хосте (хостом может быть наша локальная машина, дата центр, облачный провайдер). Контейнеры являются легковесными и содержат все необходимое для запуска приложения, что избавляет нас от необходимости полагаться на то, что установлено на хосте.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ТЕРМИНОВ

API	– (англ. Application Programming Interface) – программный интерфейс приложения, интерфейс прикладного программирования
IIoT	– (англ. Industrial Internet of Things) – Промышленный интернет вещей
IoMT	– (англ. Internet of Medical Things) – Интернет медицинских вещей
МИС	– Медицинская информационная система
МИС МО	– Медицинская информационная система медицинской организации
МО	– Медицинская организация
ОС	– Операционная система
ПО	– Программное обеспечение
Система	– Информационная система, совокупность всех подсистем информационной системы, объединённых в единое техническое решение
СУБД	– Система управления базами данных
Устройство ПМП	– Устройство для дистанционного мониторинга медицинских и функциональных показателей человека
ФЛК	– Форматно-логический контроль

