

УТВЕРЖДЕН
RU.07622667.ПМП.001.ПК-ЛУ

ИНФОРМАЦИОННАЯ СИСТЕМА ИОМТ.ИСТОК

Описание технической архитектуры

RU.07622667.ПМП.001.ПК

Листов 13

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

2024

Литера

АННОТАЦИЯ

Настоящий документ содержит описание технической 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 медицинских и функциональных показателей человека с устройств Персонального медицинского помощника (далее – устройств ПМП) в медицинские информационные системы (далее – МИС) медицинских организаций (далее – МО) и составлен в рамках регистрации системы в едином реестре российских программ для электронных вычислительных машин и баз данных (далее – реестр Минцифры).

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
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	8
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 технической реализации компонентов.....	10
4.2 Используемые технологий	10

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 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 и некоммерческой лицензией программного обеспечения, делает информационную систему 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.

Перечень микросервисов, реализующих систему:

- IoMT.Istok Gateway (описан в разделе 3);
- IoMT.Istok MIS (описан в разделе 3);
- IoMT.Istok Devices (описан в разделе 3);
- IoMT.Istok Portal (описан в разделе 3);
- IoMT.Istok Websocket (описан в разделе 3);
- IoMT.Istok Logs (описан в разделе 3);
- IoMT.Istok Redirect (описан в разделе 3);
- IoMT.Istok Napı (описан в разделе 3);
- IoMT.Istok Artemis (реализован на базе Apache Artemis 2.19.0, описан, разделе 4);
- IoMT.Istok Keycloak (реализован на базе Keycloak 20.0.3, описан в разделе 4).

2 АРХИТЕКТУРА СИСТЕМЫ

Схема архитектуры системы приведена на рисунке 1.

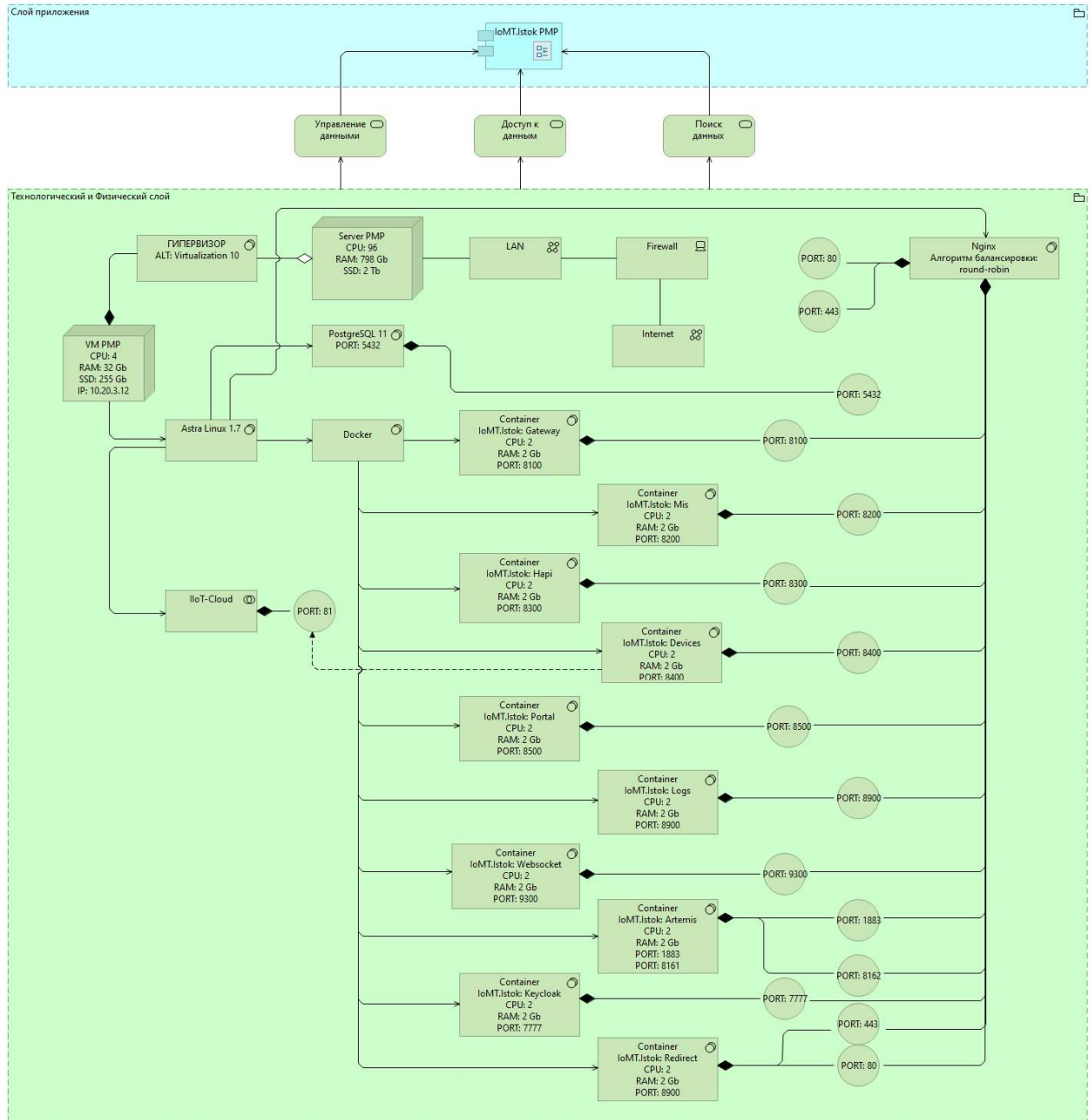


Рисунок 1 – Схема архитектуры системы

В процессе реализации использованы различные технологии, такие как Docker (версия 19.03.15), Keycloak (версия 20.0.3), PostgreSQL 11, и другие, описанные в разделе 4, с целью обеспечения требуемой функциональности и производительности.

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, IoMT.Istok-Redirect, которые обеспечивают выполнение следующих функций:

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

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;
- направления результатов форматно-логического контроля (далее – 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 может состоять из СУБД PostgreSQL 11, Postgres Pro (запись в реестр Минцифры № 104 от 18.03.2016), Tantor (запись в реестре Минцифры №23645 от 12.08.2024), IoT.Istok Cloud (в составе IoT.Istok, запись в реестре Минцифры №15250 от 25.10.2022) и микросервиса IoMT.Istok-NaPi (реализует стандарт обмена медицинской информацией по формату FHIR (fast healthcare interoperability resources), реализован на базе NaPi JPA server 7.6.0), 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 технической реализации компонентов

Микросервисный подход - 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 хранения данных могут использоваться системы управления базами данных PostgreSQL 11, Postgres Pro, Tantor. Это реляционные СУБД с открытым исходным кодом, обеспечивающие ACID-соблюдение и поддержку широкого спектра функциональных расширений.

Для хранения объемных данных, требующих высокую доступность, применяется система IoT.Istok Cloud, обладающая масштабируемостью и отказоустойчивостью.

Для обеспечения работоспособности компонентов системы используются виртуальные машины, созданные на базе виртуализации ALT Virtualization 10 (запись в реестре Минцифры №6487 от 07.04.2020). Это обеспечивает изоляцию экземпляров системы друг от друга, что улучшает безопасность, а также позволяет более гибко управлять распределением системных ресурсов.

4.2 Используемые технологии

1. Операционная система Astra Linux 1.7 на базе ядра Linux. Обеспечивающая степень защиты обрабатываемой информации до уровня государственной тайны «особой важности» включительно. Сертифицирована в системах сертификации средств защиты информации Минобороны РФ, ФСТЭК и ФСБ России. Включена в реестр Минцифры (Реестровая запись №369 от 08.04.2016).

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

3. Создание пользовательского интерфейса на стороне клиента (frontend) используются технологии React (версия 19) и Bootstrap (версия 5.3). 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 (версия 2.3).

4. Защита 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.

5. Централизованное управления доступом и авторизации в проекте используется Keycloak (версия 20.0.3). Keycloak – это открытая платформа, которая позволяет удобно управлять идентификацией пользователей, обеспечивая безопасный вход в систему через единый интерфейс. 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.

6. Поточковый протокол передачи данных WebSocket – 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;
- увеличить скорость обмена данными, так как WebSocket устанавливает постоянное соединение и избавляется от накладных расходов HTTP, связанных с постоянным открытием и закрытием соединений.

7. Система управления контейнерами Docker (версия 19.03.15) – это открытая платформа для разработки, доставки и запуска приложений. Она 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.

8. Реверс прокси Nginx (версия 1.27.0) - web-сервер, который способен держать до 10 тысяч одновременных соединений, расходуя при этом минимальное количество ресурсов. Используется в качестве прокси-сервера и обратного прокси-сервера. Обладает модульной архитектурой, конфигурируемой отдельными файлами.

9. Брокер сообщений Apache Artemis (версия 2.19) с открытым исходным кодом, реализующий спецификацию JMS 1.1. Имеющий возможности — кластеризации, хранения сообщений в различных СУБД, кэширование, ведение журналов.

Примечание – для проверки состава модулей и технологий, возможен непосредственный доступ к стенду. Для получения доступа необходимо ознакомиться с разделом 5 документа «Инструкции по удаленному доступу и эксплуатации системы».

