

**Отчет о выполнении проекта № 22-71-10092**  
**«Разработка концепции, модели и архитектуры**  
**киберфизического окружения палат в стационарных**  
**учреждениях социального и медицинского обслуживания»**  
**в период с 1 июля 2023 г. по 30 июня 2024 г.**

**Сведения о фактическом выполнении плана работы в отчетный период**

**1. Модель функционирования киберфизического окружения умной палаты на логическом уровне, включая алгоритмы выявления нештатных ситуаций, выбора сценария поведения при возникновении ситуации, оценки эффективности исполнения сценариев.**

**1.1. Общие подходы к моделированию умной палаты на логическом уровне**

Моделирование функционирования киберфизического окружения умной палаты (УП) на логическом уровне выполняется с использованием методологий, позволяющих описывать поведение этого окружения в различных условиях. На высоком уровне это поведение может описываться с помощью сценариев. Сценарий – это некоторая последовательность событий, процесс, который может развиваться штатно (например, систематический сбор данных) или нештатно (подача сигналов тревоги или обнаружение любых обстоятельств, которые могут повлечь угрозу жизни и здоровья пациентов) и может быть частично автоматизирован. По сути, функционирование умной палаты на высоком уровне задаётся набором сценариев.

Состав этого набора сильно зависит от специфики задач, решаемых УП, отделения, в котором установлена УП, типа палаты, особенностей пациентов, заболеваний, используемого оборудования и методик лечения. Сценарии для конкретной УП должны определяться с привлечением медицинских специалистов, которые будут работать с данной УП, однако возможно построить формальные модели сценариев функционирования киберфизического окружения УП для некоторых общих случаев. Для задания таких наборов сценариев можно использовать иерархию диаграмм UML.

Диаграммы вариантов использования UML специфицируют поведение УП на наиболее высоком, концептуальном уровне. Такая диаграмма определяет состав акторов (действующих лиц), а также набор и обобщённую структуру выполняемых ими действий и какие части системы эти действия затрагивают.

Диаграммы активности UML позволяют моделировать процессы, связанные с обслуживанием пациентов в палате (прием пациента, мониторинг его состояния,

назначение лечения, сбор информации о здоровье и т.д.), передачей данных в киберфизическом окружении палаты, управлением ресурсами. Диаграмма активности также задаёт процессы обнаружения и реагирования на нештатные ситуации, такие как, например, остановка сердца или дыхания пациента. Диаграмма активности отражает компоненты умной палаты в укрупнённом виде, определяет, какие сценарии функционирования киберфизического окружения применяются при возникновении тех или иных условий. Для каждого пути на диаграмме вариантов использования строится по меньшей мере одна диаграмма активности.

Диаграммы последовательности UML более детально специфицируют отдельный сценарий функционирования киберфизического окружения, который представляется в виде пути на диаграмме активностей. Соответственно, для каждой диаграммы активностей разрабатывается столько диаграмм последовательностей, сколько в этой диаграмме активностей имеется путей без учёта циклов. Такие диаграммы детализируют состав взаимодействующих компонентов умной палаты, последовательность и тип данных и сигналов, которые между ними передаются, уточняют последовательность выполнения действий, возможность их параллельного выполнения.

Таким образом, с помощью иерархической структуры диаграмм на языке UML можно описать набор сценариев функционирования для УП, решающей определённую задачу. Примеры и некоторые типовые задачи рассмотрены далее.

## **1.2. Типовые схемы моделирования поведения УП**

Как указано выше, особенности поведения УП сильно зависят от их специфики. Однако для любых УП можно выделить типовые схемы поведения. Рассмотрим несколько наиболее распространённых моделей поведения, которые могут быть использованы и дополнены для выполнения широкого спектра задач.

Одной из типовых схем работы в УП является автоматизированное наблюдение за состоянием пациентов. Распространённый случай такого наблюдения – послеоперационный мониторинг. Послеоперационный мониторинг в отделении реанимации и интенсивной терапии крайне важен для результата лечения пациента и включает такие действия, как оценка дыхания, гемодинамики, температуры тела, контроль боли, уровня сознания, водно-электролитного баланса, уровня глюкозы крови, моторики ЖКТ, профилактику рвоты. Эти меры включают как физикальные осмотры, так и инструментальные методы, такие как пульсоксиметрия, ЭКГ и УЗИ, для предотвращения осложнений, таких как аспирационная пневмония.

Интенсивность мониторинга зависит от степени анестезиологического риска и сложности проведенной операции. Пациенты с низким риском и короткими операциями подвергаются стандартному мониторингу, включающему физикальную оценку и анальгезию до 12 часов после операции. Тем, кто перенес сложные операции или имеет высокий риск осложнений, требуется интенсивный мониторинг с регулярными проверками каждые 2 часа и комплексным уходом за дренажными

системами и катетерами, а также мультимодальной анальгезией и терапией, адаптированной к текущему состоянию пациента. Дополнительные меры могут включать измерение внутричерепного давления и ежедневный контроль массы тела. Для точности мониторинга артериального давления используются автоматические сфигмоманометры и артериальные катетеры, а для отслеживания кислорода в крови — периферические пульсоксиметры.

Процедуры включают ежедневный контроль водно-электролитного баланса, а также регулярные общие анализы крови, измерения уровней магния, фосфатов, ионизированного кальция и проведение печеночных тестов для пациентов на парентеральном питании. В условиях интенсивной терапии широко применяются портативные устройства для моментального анализа крови у постели больного, позволяя оперативно получать данные о химическом составе крови, глюкометрии и газах крови, что критически важно для своевременного реагирования на изменения состояния пациента.

Во время мониторинга выполняется периодическая регистрация показателей жизненно важных функций, признаков, симптомов, которая может осуществляться как в ручном, так и в автоматическом режиме. В ручном режиме медицинский работник непосредственно выполняет сбор данных и регистрирует их с помощью приложения. В автоматическом режиме регистрация данных выполняется с помощью датчиков и мониторов. Все собранные данные агрегируются и систематизируются на сервере, после чего выполняется их оценка в автоматическом режиме согласно заложенным в систему методикам. В случае выхода показателей за допустимые интервалы, пациенты (если это предусмотрено сценарием) и медицинские специалисты уведомляются об этом через приложение. Медицинский специалист просматривает собранные данные и полученные оценки в удобном для него виде, выбирает тактику лечения, экспортирует результаты в медицинскую информационную систему (МИС), на сервер (PACS) или в иные внешние системы. Диаграмма вариантов использования для этого сценария представлена на рисунке 1.

На ней выделены разные типы акторов - лиц, взаимодействующих с приложением. В данном случае имеется три типа - пациент, медицинский работник и администратор, и все они являются подтипами общего типа акторов - “Пользователь приложения”, поскольку каждый может пользоваться приложением со своими правами доступа. Это отражается на диаграмме с помощью отношения наследования. Таким образом, операции, ассоциированные с актором “Пользователь приложения”, доступны всем типам акторов, однако отдельные операции доступны только “привилегированным” акторам - медицинским специалистам и администраторам.

Частным случаем мониторинга пациентов в УП является автоматизированный визуальный мониторинг с использованием камер видеонаблюдения с согласия пациента или его законного представителя. Такой подход можно применять в медицинских учреждениях с ограниченными финансовыми возможностями, не оснащенных современным и дорогостоящим оборудованием, или использовать для

организации мониторинга в домашних условиях для больных, маломобильных или пожилых людей, которым необходим уход и кратковременное наблюдение. Схема мониторинга представлена на рисунке 2.

Собираемые данные включают в себя два основных набора. Первый набор содержит данные, полученные с камер наблюдения, которые фиксируют всю сцену. Эта информация обычно используется для выполнения функций отслеживания объектов для обнаружения человека, детектирования его появления и исчезновения в пространстве слежения, классификации его положения и динамики. Этот набор передаётся на сервер непосредственно.

Второй набор содержит данные с одной или нескольких персональных камер, которые расположены рядом с человеком и могут лучше захватывать изображения лиц. Эта информация может быть собрана с помощью простых устройств общего назначения – например, смартфонов. Поскольку возможности современных смартфонов позволяют производить расчеты и обрабатывать данные, а также учитывая тот факт, что сигнал подключения к сети может быть низким, анализ и обработка данных выполняется на смартфонах. Результаты обработки затем передаются на системный сервер через локальную сеть. Функции смартфона обычно включают распознавание лиц и эмоций.

Другие функции системы, такие как определение позы тела, обнаружение падения, идентификация иных опасных ситуаций (например, приступов боли), выполняются на сервере, а результаты визуализируются в мобильном приложении пользователя.

К другим сценариям работы УП относится использование носимого сенсорного устройства медицинского назначения (например, браслета) для контроля состояния пациента при нахождении его как в палате, так и вне её. Основная функция браслета заключается в непрерывном мониторинге ряда жизненно важных показателей и данных пользователя, таких как пульс, уровень сатурации кислорода в крови, геолокация и т.п. Также браслет способен распознавать факт падения пользователя и имеет тревожную кнопку для экстренной связи. Все собранные данные браслет передаёт на сервер с заданной частотой, что позволяет диспетчерам и медицинским специалистам в режиме реального времени анализировать состояние пациента и оперативно реагировать на чрезвычайные ситуации. Сервер также предоставляет пользователям через API доступ к конфигурации устройств и просмотру данных. Административные и диспетчерские панели позволяют управлять устройствами и пользователями, настраивать параметры мониторинга и обрабатывать сигналы тревоги. Все интерактивные функции системы доступны через веб-интерфейсы или мобильное приложение, что делает систему гибкой и масштабируемой для различных медицинских учреждений и частных пользователей. Соответствующая диаграмма показана на рисунке 3.

На рисунках 4, 5 представлены диаграммы активности и последовательности для контроля показателей у пациентов в УП. Диаграмма активности (рисунок 4) отражает процесс сбора и контроля показателей на сервере, что позволяет выявить экстренные сценарии развития ситуации, например, недопустимое значение индекса Кердо, что требует внимания врача. В данном случае предусмотрено три способа сбора данных – ручной, с использованием мобильного приложения, автоматический – с использованием «умных» мониторов, передающих на сервер данные о жизненно важных показателях пациента, и с помощью датчиков, передающих прочие данные (например, положение тела, характер движений). На сервере применяются специальные методики для обработки этих данных, после чего фиксируется результат анализа (оценка состояния пациента, рекомендации) и при необходимости выполняется отправка уведомлений медицинскому специалисту и пациенту. Далее специалист может ознакомиться с результатом в своём приложении и при необходимости принять врачебное решение. Подход к заданию методик описан в следующем пункте.

Приведённая диаграмма последовательности (рисунок 5) отражает сценарий взаимодействия пациента и врача в УП с учётом использования датчиков и мониторов для сбора данных о состоянии пациента, а также мобильных приложений пациента и врача. Сценарий предполагает регулярный предварительный сбор данных о состоянии пациента в автоматическом (с помощью датчиков и мониторов), полуавтоматическом (с помощью средств многомодального ввода информации в клиентском приложении пациента) и ручном (с помощью медицинского персонала) режимах. Эти данные поступают на сервер, систематизируются, оцениваются и визуализируются в клиентском приложении врача, что позволяет ему более оперативно оценить состояние пациента и принять решение о необходимости его осмотра в динамике и показаниях к оказанию неотложной помощи.

Кроме того, возможна также реализация сценариев поведения УП, сходных по функциональности с системами типа «умный дом», связанных с регулированием освещения, температуры, проветривания, и т.д., выполняемым по запросу пациента. Вместе с тем, такая функциональность имеет свои особенности, так как для запроса таких действий требуется разрешение, полученное напрямую от врача либо от управляющего модуля системы. Если в каких-то ситуациях сценарии конфликтуют, то выполняется сценарий по указаниям врача. Например, проветривание помещения по просьбе пациента может выполняться только в условиях отсутствия запрета на проветривание со стороны врача. Врач с помощью своего клиентского приложения управляет разрешениями для системы климат-контроля, а пациент может запросить те или иные действия в УП, которые выполняются при условии, если они разрешены.

### **1.3. Подход к оценке состояния пациента**

Рассмотрим сценарий использования мобильного приложения для сбора, обработки и оценки данных о здоровье пациента. Этот сценарий включает сбор исходных данных (набора показателей жизненно важных функций, симптомов,

признаков), оценку каждого показателя в отдельности, получение интегральной оценки и выработку рекомендаций на основе полученных результатов.

Ручное выполнение таких операций предполагает значительные затраты времени на сбор данных и расчеты. Поэтому целесообразно разработать методы и средства автоматизации оценки состояния пациента с использованием носимых компонентов (датчиков), модулей УП и информационных компонентов (мобильных приложений или приложений для ПК).

Для упрощения построения и развертывания таких автоматизированных систем необходимо:

1. Построить типовую модель методики оценки состояния пациента на основе баллов.
2. Создать типовое приложение для работы с методами такого рода.
3. Сформулировать метод адаптации приложения для реализации конкретной методологии.

Это позволит в дальнейшем адаптировать приложение для борьбы с различными заболеваниями с использованием схожих моделей и алгоритмов.

Реализация сценария подразумевает использование известных методик оценки здоровья человека в медицине. Распространённым классом методик является оценка с помощью шкал. Такая методика может быть описана следующим образом:

$$M = \langle I, S, C, R \rangle,$$

где:

- I — набор измеряемых или оцениваемых показателей здоровья (например, пульс, температура, индекс Кердо);
- S – набор правил, определяющих зависимость начисления баллов от значений показателей здоровья (например, «прибавить 1 балл, если температура превышает 37°C»);
- C — набор условий, определяющих, требуется ли выполнить то или иное действие или выдать рекомендацию при определенных условиях (например, «подать сигнал тревоги, если оценка выше 10»);
- R – набор действий, результатов и рекомендаций (например, «сообщить врачу», «заявить, что вероятность заболевания высока»).

Реализация такого метода в приложении требует модели данных. В минимальной конфигурации она изображена на рисунке 6.

В этой модели общий балл также рассматривается как показатель здоровья (Indicator). Правила расчёта (ScoringRule) определяются условиями и соответствующими условными операциями, которые представлены математическими выражениями. Например, сложение баллов в зависимости от температуры по методике, определенной в статье [Ценципер Л.М., Мотиенко А.И., Терехов И.С., Левоневский Д.К., Самочерных К.А., Кондратьев А.Н. Цифровое решение для определения тяжести синдрома пароксизмальной симпатической гиперактивности у пациентов с повреждением головного мозга. Вестник анестезиологии и реаниматологии. 2023;20(6):90-96. <https://doi.org/10.24884/2078-5658-2023-20-6-90-96>], описывается набором правил (таблица 1). В выражениях обозначения [S] и [T] означают исходный и целевой показатели соответственно. В остальном выражения строятся с использованием синтаксиса, подобного SQL.

Представленная модель, помимо определения индикаторов, правил и действий, позволяет подключать плагины ввода и вывода. Это необходимо с точки зрения интеграции с датчиками, модулями умных помещений, базами данных и другими внешними системами. Это позволяет использовать приложение не только как отдельный модуль, но и как интегрированный компонент инфраструктуры умных палат. Основные типы входных плагинов:

1. Нет (по умолчанию): требуется ручной ввод данных.
2. Плагин расчета: значение индикатора определяется как производная от других индикаторов. Примером может служить индекс Кердо, который рассчитывается на основе пульса и артериального давления.
3. Плагины для подключения к датчикам: реализуют автоматический ввод данных с различных измерительных приборов.
4. Плагины для баз данных: реализуют сбор данных из внешних источников (баз данных и медицинских информационных систем).

Основные типы плагинов вывода:

1. Нет (по умолчанию): данные отображаются на экране пользовательского устройства.
2. Плагины для баз данных: передают данные во внешние источники (базы данных и медицинские информационные системы).

Заполнение модели данных и подключение плагинов производится на базе типового приложения по сбору и обработке данных. Прототип такого приложения для диагностики конкретного заболевания был рассмотрен в работе [D. Levonevskiy, A. Motienko, I. Terekhov, Automation of diagnosis, stratification, and treatment of the paroxysmal sympathetic hyperactivity syndrome in the smart ward environment, in: 2nd International Conference on Computer Applications for Management and Sustainable Development of Production and Industry (CMSD-II-2022), 2022, 12564, pp. 74-80, doi:

10.1117/12.2669244] и получил развитие в [Ценципер Л.М., Мотиенко А.И., Терехов И.С., Левоневский Д.К., Самочерных К.А., Кондратьев А.Н. Цифровое решение для определения тяжести синдрома пароксизмальной симпатической гиперактивности у пациентов с повреждением головного мозга. Вестник анестезиологии и реаниматологии. 2023;20(6):90-96. <https://doi.org/10.24884/2078-5658-2023-20-6-90-96>]. Данный прототип разработан с использованием кроссплатформенных технологий (Flutter), поэтому возможно выполнение сборки для различных типов хостинговых устройств и операционных систем. Учитывая вышеизложенные вспомогательные модели и концепции, результирующий метод создания необходимого приложения состоит из следующих шагов:

1. Определить исходные данные: показатели; правила подсчета; правила получения результатов.
2. Определить необходимые входные плагины для каждого индикатора. Если нужных плагинов нет - их необходимо разработать.
3. Определить необходимые плагины для вывода данных и при необходимости разработать их.
4. Собрать типовое приложение с подключенными плагинами для необходимой платформы.
5. Установить приложение.
6. Наполнить базу данных приложения информацией об индикаторах и правилах. Структура приложения и его использование как компонента инфраструктуры умной палаты представлена на рисунке 7.

#### **1.4. Аprobация и численные оценки**

Для оценки качества работы УП необходимо использовать численные показатели её эффективности. Наиболее распространенные показатели могут отражать следующие аспекты их функционирования:

1. Качество медицинской помощи: сокращение времени нахождения пациента в больнице, улучшение точности диагностики, снижение летальности и количества осложнений после операций и т.д.
2. Комфорт пациентов: снижение числа жалоб, опросы пациентов и т.д.
3. Оптимизацию процессов медицинского ухода: улучшение доступности медицинской информации, сокращение времени на выполнение рутинных процедур и т.д.
4. Затраты на здравоохранение: снижение числа процедур и препаратов, снижение трудозатрат, уменьшение числа повторных визитов пациентов в больницу, ошибок при медицинском обслуживании и т.д.



Валидацию предлагаемого метода можно провести на примере реализации приложения для диагностики и стратификации синдрома пароксизмальной симпатической гиперактивности (ПСГА) с использованием показателей, отражающих сокращение времени на выполнение рутинных процедур при условии получения идентичного результата (т.е. без потерь в качестве оказываемой помощи). Для этого путем измерения, расчета и визуального осмотра извлекают набор показателей (пульс, частота дыхания, артериальное давление, температура, индекс Кердо и др.) Весь процесс занимает около 5 минут, его алгоритм приведён на рисунке 8.

Было рассмотрено несколько режимов работы системы в зависимости от степени автоматизации:

- уровень 0: все процессы выполняются вручную, и мы используем типовые значения времени для измерения, визуального контроля и расчета;
- уровень 1: расчеты и оценка полностью автоматизированы, но измерения и ввод данных по-прежнему выполняются вручную;
- уровень 2: некоторые показатели (частота пульса, артериальное давление) измеряются и автоматически извлекаются приложением со смарт-монитора.

Используя экспериментальные и модельные данные, мы построили гистограммы времени сбора и обработки информации. Они показаны на рисунке 9.

Результаты моделирования [Левоневский Д. К., Мотиенко А. И. Моделирование и автоматизация процессов сбора и обработки данных в умной медицинской палате // Программная инженерия. 2023. Том 14, № 10. С. 502—512. DOI: 10.17587/prin.14.502-512] подтверждают снижение времени обработки данных в зависимости от количества показателей здоровья  $N_A$ , которые могут быть автоматически получены или рассчитаны:

$$D_{02}(N_A) = T_0/T_2(N_A) \quad (1)$$

$$D_{12}(N_A) = T_1/T_2(N_A) \quad (2)$$

В формулах (1) и (2) каждый  $T_i$  — среднее время обработки данных для  $i$ -го уровня автоматизации.

Среднее время, необходимое для сбора и обработки информации, составляет 12 мин, 6 мин и 5 мин для уровней 0, 1, 2. Проведённое моделирование показывает, что автоматизация расчетов сокращает время реализации сценария в среднем на 50%, а использование умного монитора – дополнительно на 16%.

## **2. Архитектура киберфизического окружения умной палаты, позволяющую комплектовать это окружение компонентами исходя из задач и нагрузки.**

### **2.1. Общий подход к построению архитектуры УП**

Архитектура УП (медицинской киберфизической системы) включает в себя совокупность множества взаимосвязанных технологий и устройств, которые вместе создают интегрированную сеть для мониторинга, анализа и управления состоянием пациента в реальном времени. Для построения архитектуры вначале выделим и систематизируем основные компоненты палаты. Прежде всего выполним эту систематизацию на логическом уровне, оперируя не конкретными модулями системы, а их ролями в умной палате. При таком подходе выделяются следующие типы ролей:

1. Сбор данных – роль, включающая регистрацию различных медицинских показателей пациента, таких как пульс, давление, температура тела, уровень кислорода в крови, прочие показатели жизненно важных функций и симптомов. Сбор данных предполагает использование разнообразных сенсоров и устройств, например, носимых мониторов, камер.

2. Коммуникация – роль, обеспечивающая передачу данных между устройствами и людьми внутри палаты и за её пределы. Коммуникация между устройствами может включать беспроводную связь через Wi-Fi или Bluetooth, проводную через Ethernet, параллельные или последовательные порты (например, USB). Коммуникация между устройствами и людьми (пациентами, персоналом) реализуется с помощью пользовательских интерфейсов устройств и приложений, а также многомодальных интерфейсов, роль которых особенно важна для пациентов с ограниченными возможностями.

3. Хранение данных - роль, обеспечивающая хранение данных, полученных из различных источников с реализацией принципов целостности, доступности и конфиденциальности. Это может включать как локальные системы хранения, так и облачные решения для обеспечения масштабируемости и доступности данных.

4. Обработка данных и поддержка принятия решений – роль, включающая анализ собранных данных и формирование предварительных выводов о состоянии здоровья пациента. Таким образом система может предоставлять поддержку принятия решений медицинским специалистам, предлагая возможные сценарии действий на основе исторических данных и текущего состояния. Также она может генерировать предупреждения или рекомендации для медицинского персонала и корректировать работу подключенных устройств. Возможно использование машинного обучения для анализа данных и улучшения алгоритмов, что позволяет повышать точность прогнозов и эффективность предложений.

5. Принятие решений – роль, которая выполняется только человеком на основе собранных и обработанных данных.

6. Исполнение решений – роль, включающая реагирование на принятые решения, например, автоматическую подачу лекарств или изменение настроек медицинских аппаратов при подтверждении этих решений медицинскими специалистами.

7. Визуализация данных включает представление данных в понятной форме для медицинских специалистов и пациентов через пользовательские интерфейсы, такие как диспетчерские панели на компьютерах или мобильных приложениях, что позволяет проще интерпретировать медицинскую информацию и следить за изменениями состояния пациента.

При этом мы предполагаем, что выполнение всех этих действий происходит в защищённом контуре учреждения, поэтому компоненты и роли, связанные с защитой этого контура, считаем внешними по отношению к системе.

На физическом уровне умная палата состоит из модулей, взаимодействующих между собой, с пациентами и персоналом, и реализующих перечисленные выше роли, причём один модуль может выполнять множество ролей. Для каждого устройства и каждой выполняемой им роли можно определить множество параметров, основные из которых приведены в таблице 2.

Описать основные роли, задействованные в архитектуре умной палаты, их связи и вспомогательные сущности можно с помощью диаграммы классов (рисунок 10).

## **2.2. Классификация компонентов, используемых в умной палате.**

Можно выделить следующие основные типы модулей в УП:

1. Устройства мониторинга: используются для наблюдения за состоянием пациентов. Они предназначены для сбора, передачи, обработки и визуализации данных о пациентах, их симптомах, поведении, показателях жизненно важных функций. Устройства используются в операционных залах, реанимационных отделениях, для послеоперационного мониторинга в условиях интенсивной терапии, некоторые из них могут использоваться для дистанционного мониторинга пациентов в домашних условиях (например, медицинские браслеты). Подобные устройства различаются по специфике мониторинга (сердечная деятельность, температура, внутричерепное давление и т.п.), методу получения данных (инвазивный, неинвазивный, ультразвуковой) и области применения (от общих больничных палат до специализированных отделений, таких как отделение реанимации и интенсивной терапии), по конструкции (модульный, моноблочный).

Наиболее распространённым устройством такого рода является медицинский монитор. Медицинские мониторы классифицируются по количеству каналов (измеряемых параметров). Основные параметры, отслеживаемые медицинскими мониторами – инвазивное и неинвазивное артериальное давление, частота дыхания, показатели ЭКГ, температура тела, сатурация крови кислородом. Медицинские мониторы также могут включать дополнительные функции, такие как биспектральный

индекс для мониторинга глубины анестезии, модули капнометрии и капнографии для контроля углекислого газа, а также мультигазовые блоки для анализа анестезирующих газов. Также мониторы предоставляют возможность подключения дополнительных измерительных модулей, улучшая их адаптивность и многофункциональность в клинических условиях.

Типовыми примерами мониторов являются кардиомониторы (контролируют сердечный ритм, давление, пульс, и насыщение крови кислородом, могут использоваться для мониторинга глубины седации), тепловентиляторы (используются для поддержания температуры тела пациента, особенно после операций, когда пациенты могут испытывать озноб), аппарат «искусственная почка» (применяются для проведения гемодиализа при тяжелых заболеваниях почек), ультразвуковые устройства (предназначены для диагностики и мониторинга внутренних органов, включая оценку наличия свободной жидкости в брюшной полости и мониторинг сердечного выброса с помощью доплерографии пищевода), мониторы внутричерепного давления (используются для мониторинга давления в черепе, что критически важно для пациентов с тяжелыми травмами головы или неврологическими состояниями). Примеры моделей – портативные глюкометры Контур Плюс Уан, Акку-Чек Инстант, системы непрерывного мониторинга глюкозы FreeStyle Libre, Guardian Connect MMT – 7820, пальчиковый пульсоксиметр HUM AEROcheck, система удалённого мониторинга сердца Mintti Heartbook, автоматический тонометр Omron RS7 и др.

2. Устройства управления (актуаторы). К таким устройствам могут относиться инфузионные насосы для введения медикаментов, вентиляционные аппараты, автоматизированные системы для изменения положения кровати пациента, устройства климат-контроля. Также в настоящее время расширяются возможности применения медицинских роботов. Они облегчают работу медицинского персонала, автоматизируя рутинные задачи – забор крови, измерение температуры, дезинфекция помещений и т.д., при этом используя передовые технологии, например импульсный ксеноновый свет. В диагностике эти роботы способны выполнять ускоренные анализы крови и помогают в проведении эндоскопий. Реабилитационные роботы, в том числе терапевтические экзоскелеты и роботизированные протезы, поддерживают пациентов при восстановлении, а интеллектуальные инвалидные коляски адаптируются к окружающей среде. Роботы-ассистенты и консультанты существенно снижают административную нагрузку, автоматизируя запись на прием и управление потоком пациентов. Роботы-компаньоны играют роль питомцев, облегчая психологическую адаптацию и реабилитацию, особенно среди пожилых людей и детей. Роботы-тренажеры нацелены на обучение медицинского персонала, имитируя реальные клинические условия. Роботы для доставки упрощают логистику внутри медицинских учреждений, а в радиотерапии и радиохирургии роботы обеспечивают точечное облучение опухолей. Нанороботы и микророботы открывают новые перспективы в таргетированной доставке лекарств, минимизируя побочные эффекты и ускоряя процессы лечения. Актуаторы также используются для управления работой медицинского оборудования в палате. Автоматизация таких процессов помогает

поддерживать необходимые параметры лечения без постоянного вмешательства персонала. Примеры – функциональная кровать Linet Eleganza Smart, роботы компаньоны Paro, Lovot, робот-рука UR10.

3. Устройства хранения и обработки данных - информационные компоненты, не связанные непосредственно с физическими процессами. Они аккумулируют, анализируют и обрабатывают медицинскую информацию, что позволяет медицинскому персоналу оперативно получать данные о состоянии пациентов и своевременно реагировать на любые изменения состояния их здоровья. Кроме того, эти устройства интегрируются с электронными медицинскими картами, медицинскими информационными системами (МИС), серверами хранения медицинских данных (PACS), обеспечивая непрерывный доступ к истории болезни, содержащей результаты предыдущих исследований и назначения, информацию об аллергии и т.п. Это упрощает процесс диагностики и выбора методов лечения. В дополнение, такие устройства могут включать функции искусственного интеллекта и машинного обучения. Пример – мобильное приложение для диагностики и стратификации синдрома пароксизмальной симпатической гиперактивности ПСГАмер.

Кроме того, все модули УП классифицируются в соответствии с поддерживаемыми ими технологиями связи. Такая классификация может быть проведена для каждого уровня модели взаимодействия открытых систем ISO OSI, однако на практике, с точки зрения интеграции модулей в единую архитектуру, важны две группы уровней. Во-первых, это низкие уровни – физический и канальный, связанные с используемыми технологиями и стандартами передачи данных – проводная (Ethernet, USB, COM), беспроводная (Wi-Fi, Bluetooth, GSM). Во-вторых, это высокие уровни – прикладной и уровень представления, ответственные за форматы передаваемых данных и их интерпретацию. Наиболее известным стандартом в этой области является HL7.

### **2.3. Архитектура умной палаты и подход к её комплектации**

Архитектура УП предполагает организацию модулей в двух контурах – внешнем (киберфизическом) и внутреннем (информационном).

Во киберфизическом контуре УП выполняет сбор данных и получение сигналов от акторов (пациентов, среднего медицинского персонала, врачей, администраторов) и окружающей среды (климат, освещённость, возникновение чрезвычайных ситуаций) с помощью сенсоров и пользовательских интерфейсов. Все собранные данные попадают в информационный контур, где выполняется регистрация, агрегация, трансформация, анализ данных, машинное обучение, функционируют экспертные системы. Модули этого контура не имеют непосредственной связи с физическими объектами и процессами, но акторы могут влиять на функционирование этого контура с помощью пользовательских интерфейсов (например, врач может с помощью мобильного приложения или диспетчерской панели принять или отклонить рекомендацию экспертной системы, одобрить или запретить запрос пациента на регулировку климата в палате). Здесь же выполняется визуализация данных. В случае, если результат

обработки данных требует отражения в физических процессах УП, то управление возвращается в киберфизический контур и передаются соответствующие управляющие сигналы актуаторам, которые затем выполняют необходимые физические действия. Предлагаемая архитектура умной палаты представлена на рисунке 11.

На приведённом рисунке для упрощения каждый тип модуля представлен в единственном экземпляре. Также опущены компоненты преобразования данных и балансировки нагрузки. Пунктирными линиями обозначено непосредственное взаимодействие медицинского персонала с пациентом и оборудованием палаты, сплошными – основные процессы передачи данных в умной палате.

К основным особенностям предложенной архитектуры относятся модульность, расширяемость, возможность выполнять комплектацию киберфизического окружения компонентами исходя из задач УП и планируемой нагрузки, заменять отдельные компоненты на аналогичные при условиях поддержки ими идентичных протоколов либо использования вспомогательных модулей, подключаться к внешним сервисам.

Приведённая архитектура носит общий характер. Для того, чтобы выполнить комплектацию предложенной архитектуры компонентами для решения конкретных задач в рамках умной палаты, необходимы следующие действия.

1. Спецификация сценариев поведения умной палаты и построение их моделей в нотации UML, как было описано выше.

2. Выделение ролей, которые должны выполнять модули умной палаты в соответствии с данными из таблицы 2 и рисунка 10, и определение их параметров.

3. Определение набора модулей (программного и аппаратного обеспечения) для реализации ролей в рамках архитектуры (рисунок 11).

4. Проверка совместимости модулей, между которыми предусмотрено взаимодействие (рисунок 11). Если смежные модули используют несовместимые интерфейсы и протоколы, необходимо либо скорректировать набор модулей, либо ввести дополнительные модули, отвечающие за преобразование сигналов.

5. Определение количества модулей исходя из их нагрузки. К примеру, один монитор может отслеживать состояние от 1 до 12 пациентов. Если у задействованного модуля недостаточно ресурса для реализации роли с учётом кратности связей, то необходимо введение дублирующих устройств и модулей коммуникации, а также модулей балансировки нагрузки.

Подобный подход позволяет создать надёжную и функциональную умную медицинскую палату, адаптируемую к различным медицинским задачам.

## **Сведения о достигнутых конкретных научных результатах в отчетном периоде**

В ходе выполнения проекта получен ряд научных и практических результатов.

1. Предложена модель функционирования киберфизического окружения умной медицинской палаты на логическом уровне. Модель функционирования умной палаты определяется задачами этой палаты и задаётся с помощью набора сценариев – формализованных последовательностей событий, процессов, которые могут быть частично или полностью автоматизированы. Для задания набора сценариев используется иерархия диаграмм UML, состоящая из трёх уровней.

На наиболее высоком, концептуальном уровне поведение палаты моделируется с помощью диаграмм вариантов использования. Такие диаграммы определяют состав акторов (действующих лиц), набор и обобщённую структуру выполняемых ими действий, части системы, которые эти действия затрагивают. На среднем уровне диаграммы активности позволяют моделировать процессы, связанные с обслуживанием пациентов в палате (прием пациента, мониторинг его состояния, назначение лечения, сбор информации о здоровье и т.д.), передачей данных в киберфизическом окружении палаты, управлением ресурсами. На низком уровне диаграммы последовательности более детально специфицируют отдельный сценарий функционирования киберфизического окружения, который представляется в виде пути на диаграмме активностей.

Для различных типовых задач умной палаты, таких как автоматизированное наблюдение за состоянием пациентов, визуальный мониторинг, удалённый мониторинг, анализ состояния здоровья пациента, генерация рекомендаций, приведены диаграммы, иллюстрирующие сценарии функционирования палаты для этих случаев. Детально описаны и рассмотрены сценарии сбора, обработки и оценки данных о здоровье пациента с использованием медицинских методик, основанных на шкалах. Примером является методика диагностики и стратификации синдрома пароксизмальной симпатической гиперактивности. Сценарии, основанные на подобных методиках, включают сбор исходных данных (набора показателей здоровья), оценку каждого показателя в отдельности, получение интегральной оценки и выработку рекомендаций на основе полученных результатов. Проведена апробация представленного подхода.

2. Предложена архитектура киберфизического окружения умной палаты, которая представляет собой совокупность множества взаимосвязанных технологий и устройств, которые вместе создают интегрированную сеть для мониторинга, анализа и управления состоянием пациента в реальном времени. Рассмотрены типовые модули, используемые в умной палате – модули сбора данных, управляющие модули, модули обработки данных. Приведена их классификация.

Модули умной палаты реализуют логические роли, причём один модуль может выполнять множество ролей в системе. Эти роли были систематизированы и

параметризованы. К основным ролям отнесены сбор данных, коммуникация, хранение, обработка данных, принятие решений, исполнение решений, визуализация данных.

Представлена схема архитектуры умной палаты. К основным особенностям предложенной архитектуры относятся организация модулей в двух контурах – внешнем (киберфизическом) и внутреннем (информационном), модульность, расширяемость, возможность выполнять комплектацию киберфизического окружения компонентами исходя из задач УП и планируемой нагрузки, заменять отдельные компоненты на аналогичные при условиях поддержки ими идентичных протоколов либо использования вспомогательных модулей, подключаться к внешним сервисам. В киберфизическом контуре умная палата выполняет сбор данных и получение сигналов от акторов (пациентов, медсестёр, врачей, администраторов) и окружающей среды (климат, освещённость, возникновение чрезвычайных ситуаций) с помощью сенсоров и пользовательских интерфейсов. Все собранные данные попадают в информационный контур, где выполняется регистрация, агрегация, трансформация, анализ данных, машинное обучение, функционируют экспертные системы. Модули этого контура не имеют непосредственной связи с физическими объектами и процессами, но акторы могут влиять на функционирование этого контура с помощью пользовательских интерфейсов (например, врач может с помощью мобильного приложения или диспетчерской панели принять или отклонить рекомендацию экспертной системы, одобрить или запретить запрос пациента на регулировку климата в палате). Здесь же выполняется визуализация данных. В случае, если результат обработки данных требует отражения в физических процессах палаты, то управление возвращается в киберфизический контур и передаются соответствующие управляющие сигналы актуаторам, которые затем выполняют необходимые физические действия. Предложен подход к комплектации умной палаты, основанный на учёте совместимости модулей и их допустимой нагрузки.

3. Опубликовано 6 научных работ в рецензируемых изданиях, индексируемых в Scopus и RSCI, из них 2 – в журналах.

4. Отчётные материалы за второй год выполнения проекта представлены на следующей странице:

<https://spcras.ru/documents/grants/report-22-71-10092-2024.pdf>

5. Разработаны технические задания на разработку системы сбора, хранения и визуализации данных с помощью носимого устройства, а также её серверной части (приложение).



## Приложение. Технические задания

### Техническое задание на разработку системы, предназначенную для сбора, хранения и визуализации данных

#### 1. Общие сведения

1.1. Необходимо разработать систему, предназначенную для сбора, хранения и визуализации данных, полученных с носимых медицинских устройств заданных образцов, а также управления этими процессами.

1.2. Язык системы – русский.

1.3. Срок разработки - 5 месяцев.

1.4. Компоненты системы:

- носимые устройства;
- сервер;
- административная панель;
- диспетчерская панель;
- мобильное приложение;

1.5. Основные этапы разработки:

- проектирование;
- разработка серверной базы данных, ORM и API;
- параллельно:
  - разработка приложений сбора и обмена данными для носимых устройств;
  - разработка веб-интерфейсов сервера;
  - разработка мобильного приложения.
- тестирование.

#### 2. Носимые устройства

2.1. Виды носимых устройств - часы, брелок установленных моделей, функциональность которых позволяет выполнять сбор ряда показателей и сигналов:

- уровень заряда устройства;
- геолокация;
- пульс;
- сатурация;
- обнаружение факта падения;
- приём сигнала тревоги от пользователя (тревожная кнопка).

2.2. Устройства измеряют доступные показатели с частотой, заданной конфигурацией устройства.

2.3. Устройства взаимодействуют с сервером, используя API сервера. Взаимодействие включает:

- авторизацию на сервере;
- запрос конфигурации у сервера (конфигурация включает частоту измерения и отправки на сервер каждого измеряемого показателя, условия генерации сигнала тревоги для каждого измеряемого показателя - допустимые интервалы);
- отправку сообщения на сервер при нажатии тревожной кнопки или наступлении события тревоги согласно полученной с сервера конфигурации;
- периодическую отправку собранных данных на сервер с частотой, установленной конфигурацией.

2.4. Основная функциональность устройства:

- измерение температуры тела пользователя;
- измерение пульса пользователя;
- неинвазивное измерение оксигенации крови пользователя;
- распознавание и измерение параметров двигательной активности руки пользователя;
- отправка информации с датчиков носимого сенсорного устройства на удаленный сервер через Интернет с использованием сотовой связи;
- воспроизведение ограниченного количества специализированных тональных сигналов;
- программная конфигурация назначения кнопок носимого сенсорного устройства;
- программная конфигурация световой индикации носимого сенсорного устройства;
- управление и конфигурация устройства с использованием Bluetooth интерфейса и через Интернет с использованием сотовой связи.

2.5. Основные характеристики устройства:

- размеры не более: 55\*55\*20 мм;
- масса не более: 60 г;
- время автономной работы не менее: 16 ч. (при условии отправки данных пульс и сатурация каждые 10 минут, за период 10 минут; при условии съема данных с датчиков устройства каждые 5 секунд; при условии реализации событийной системы оповещения для критических состояний пользователя);
- наличие кнопки с программируемым функционалом на фронтальной поверхности;
- степень защищенности устройства от пыли и влаги: IP54;
- возможность эксплуатация устройства при температуре от 0 до 45 °С, и относительной влажности от 5 до 95% без конденсации.

2.6. Состав устройства:

- микроконтроллер;
- модуль связи GSM/GPRS;

- модуль связи Bluetooth;
- пульсоксиметрический датчик;
- датчик температуры;
- инерциальный датчик;
- звуковой излучатель;
- световые индикаторы режима работы и заряда;
- световой индикатор состояния «опасность»;
- тактовая кнопка «опасность»;
- контроллер заряда аккумуляторной батареи;
- аккумуляторная батарея.

Преимуществами предлагаемого программно-аппаратного решения является: широкие возможности интеграции в инфраструктуру УП, возможность локализации производства и сборки носимого сенсорного устройства в РФ, возможность модернизации и модификации аппаратной и программной частей устройства под задачи эксплуатации, продолжительное время автономной работы.

### 3. Сервер

3.1. Сервер выполняет сбор, хранение, визуализацию данных, собранных с носимых устройств клиентов, предоставляет API для носимых устройств и мобильного приложения, предоставляет веб-интерфейс административной и диспетчерской панелей.

3.2. Сервер хранит данные о сущностях предметной области:

- организации (клиенты) - название;
- учётные записи пользователей - логин, пароль, принадлежность к организации, роль, какие устройства пользователь может просматривать и/или настраивать;
- носимые устройства - название, IMEI, телефон, часовой пояс, принадлежность к организации;
- показатели - название, единица измерения;
- записи показателей носимых устройств - устройство, показатель, значение, метка времени;
- журнал событий носимых устройств - устройство, сообщение, тип (норма, тревога), метка времени;
- конфигурации носимых устройств - устройство, показатель, период измерения, период отправки, нижняя граница, верхняя граница, тревожное сообщение;
- конфигурации геозаборов - устройство, область, условие времени, условие генерации тревоги (вошёл в область, вышел, не вошёл, не вышел); путь представляет собой несколько последовательно заданных конфигураций геозаборов для одного устройства;
- область - тип (точка+радиус, область на карте), GeoJSON-описание.

3.3. API сервера включает в себя следующие возможности:

- авторизация носимого устройства (по IMEI, ...) или мобильного приложения (по логину, паролю);

- приём и регистрация статистики и сигналов от носимых устройств;
- отправление конфигураций носимым устройствам;
- чтение, изменение, создание, удаление объектов предметной области через запросы от мобильного приложения.

3.4. Доступ к возможностям сервера предоставляется, исходя из назначенных пользователю ролей:

- Суперадмин - полный доступ;
- Админ - полный доступ, кроме удаления объектов и данных;
- Диспетчер - доступ только на чтение ко всем данным;
- Администратор организации - полный доступ ко всем объектам в рамках одной организации, кроме удаления объектов и данных;
- Диспетчер организации - доступ только на чтение ко всем объектам в рамках одной организации;
- Пользователь организации - доступ только на чтение и конфигурирование к заданному набору устройств в рамках одной организации.

#### 4. Административная панель

4.1. Доступ к административной панели предоставляется через веб-интерфейс. Для этого пользователь должен ввести логин и пароль и иметь роль “Суперадмин” или “Админ”.

4.2. Административная панель содержит разделы:

- клиенты - управление организациями (создание, изменение, удаление);
- учётные записи - управление пользователями (создание, изменение, удаление, назначение ролей, привязка контролируемых устройств);
- устройства - управление носимыми устройствами и задание их конфигураций (настройка частот сбора и отправки данных, условий для генерации сигналов тревоги, задание геозаборов):

4.3. Пользователь с ролью “Суперадмин” имеет полный доступ к изменению содержимого разделов. Пользователь с ролью “Админ” имеет полный доступ, за исключением удаления клиентов, учётных записей, устройств.

#### 5. Диспетчерская панель

5.1. Доступ к диспетчерской панели предоставляется через веб-интерфейс. Для этого пользователь должен ввести логин и пароль и иметь роль “Суперадмин”, “Админ”, “Администратор организации”, “Диспетчер организации”.

5.2. Диспетчерская панель содержит:

- карту, на которой отображается местоположение всех доступных пользователю устройств, а также их геозаборы;
- журнал статистики и событий всех доступных устройств;
- при выборе устройства отображаются его параметры.

#### 6. Мобильное приложение

6.1. Доступ к мобильному приложению предоставляется пользователю при авторизации через логин и пароль при условии, что пользователь имеет одну из ролей “Суперадмин”, “Админ”, “Администратор организации”, “Диспетчер организации”, “Пользователь организации”.

6.2. Мобильное приложение дублирует функционал диспетчерской панели при наличии соответствующих прав у пользователя приложения.

6.3. Мобильное приложение позволяет пользователю с ролью “Пользователь организации” задать конфигурации для связанных с ним устройств.

## 7. Результат разработки

По итогам разработки заказчику предоставляются следующие результаты:

4.1. ПО сервера и комплект для его развёртывания.

4.2. Мобильное приложение.

4.3. Приложения для носимых устройств.

## Техническое задание на разработку серверной части приложения

### 1. Общая информация

3.1. Сервер выполняет сбор, хранение, визуализацию данных, собранных с носимых устройств клиентов, предоставляет API для носимых устройств и мобильного приложения.

3.2. Сервер разворачивается с использованием следующего ПО:

- операционная система Ubuntu версии 18 или выше;
- веб-сервер Apache 2 с подключённым модулем PHP 7;
- система управления базой данных PostgreSQL с модулем PostGIS для хранения геоданных.

3.3. Сервер состоит из следующих логических подсистем:

- база данных - хранит информацию об объектах предметной области и их свойствах;
- серверное приложение:
  - через HTTP(S) предоставляет API устройствам и приложениям для работы с объектами предметной области;
  - через HTTP(S) предоставляет веб-интерфейс для диспетчерской и административной панели;
  - имеет модули аналитики реального времени (для контроля жизненно важных показателей, геозаборов и т.п.);
  - имеет модули работы с пользователями (авторизация, регистрация, оповещения о событиях через email, SMS и т.п.)

## 2. База данных

2.1. База данных содержит в реляционном виде информацию о следующих видах объектов предметной области и их свойствах:

- организации (клиенты) - название;
- учётные записи пользователей - логин, пароль, принадлежность к организации, роль, какие устройства пользователь может просматривать и/или настраивать;
- носимые устройства - название, серийный номер и иные идентификационные данные – при наличии, часовой пояс, принадлежность к организации;
- показатели - название, единица измерения;
- записи показателей носимых устройств - устройство, показатель, значение, метка времени;
- журнал событий носимых устройств - устройство, сообщение, тип (отладка, норма, предупреждение, тревога), метка времени;
- конфигурации носимых устройств - устройство, показатель, период измерения, период отправки, нижняя граница, верхняя граница, тревожное сообщение;
- конфигурации геозаборов - устройство, область, условие времени, условие генерации тревоги (вошёл в область, вышел, не вошёл, не вышел); путь представляет собой несколько последовательно заданных конфигураций геозаборов для одного устройства;
- область - тип (точка+радиус, область на карте), GeoJSON-описание.

2.2. Для работы с объектами предметной области и их свойствами используется система классов в виде ORM. Каждый класс представляет собой один тип объектов и наследуется от общего типа, содержащего общий для всех объектов набор данных и функций.

## 3. API

3.1. API сервера включает в себя следующие возможности:

- авторизация носимого устройства (по серийному номеру, ...) или мобильного приложения (по логину, паролю);
- приём и регистрация статистики и сигналов от носимых устройств;
- отправление конфигураций носимым устройствам;
- чтение, изменение, создание, удаление объектов предметной области через запросы от мобильного приложения.

3.2. Доступ к возможностям API предоставляется, исходя из назначенных пользователю ролей:

- Суперадмин - полный доступ;
- Админ - полный доступ, кроме удаления объектов и данных;
- Диспетчер - доступ только на чтение ко всем данным;
- Администратор организации - полный доступ ко всем объектам в рамках одной организации, кроме удаления объектов и данных;
- Диспетчер организации - доступ только на чтение ко всем объектам в рамках одной организации;

- Пользователь организации - доступ только на чтение и настройку к заданному набору устройств в рамках одной организации.

Доступ для гостя (т.е. без авторизации) предоставляется только к функциям авторизации, регистрации, восстановления пароля. После прохождения авторизации клиент использует токены, идентифицирующие его сессию и позволяющие получить доступ к остальным функциям в соответствии с назначенной ролью.

3.3. Обращение к API выполняется с помощью HTTP-запросов GET и POST. Каждый запрос должен содержать название команды и её параметры. Ответом на запрос является JSON, всегда содержащий код ошибки (0 означает успешное выполнение операции) и, опционально, дополнительную информацию. Должен поддерживаться как минимум следующий список команд (таблица 3).

#### **4. Административная панель**

4.1. Доступ к административной панели предоставляется через веб-интерфейс. Для этого пользователь должен ввести логин и пароль и иметь роль “Суперадмин” или “Админ”.

4.2. Административная панель содержит разделы:

- клиенты - управление организациями (создание, изменение, удаление);
- учётные записи - управление пользователями (создание, изменение, удаление, назначение ролей, привязка контролируемых устройств);
- устройства - управление носимыми устройствами и задание их конфигураций (настройка частот сбора и отправки данных, условий для генерации сигналов тревоги, задание геозаборов):

4.3. Пользователь с ролью “Суперадмин” имеет полный доступ к изменению содержимого разделов. Пользователь с ролью “Админ” имеет полный доступ, за исключением удаления клиентов, учётных записей, устройств.

#### **5. Диспетчерская панель**

5.1. Доступ к диспетчерской панели предоставляется через веб-интерфейс. Для этого пользователь должен ввести логин и пароль и иметь роль “Суперадмин”, “Админ”, “Администратор организации”, “Диспетчер организации”.

5.2. Диспетчерская панель содержит:

- карту, на которой отображается местоположение всех доступных пользователю устройств, а также их геозаборы;
- журнал статистики и событий всех доступных устройств;
- при выборе устройства отображаются его параметры.

#### **6. Результат разработки**

По итогам разработки заказчику предоставляется комплект программного обеспечения сервера, включающий исходные коды и скрипт для развёртывания сервера в ОС Ubuntu.

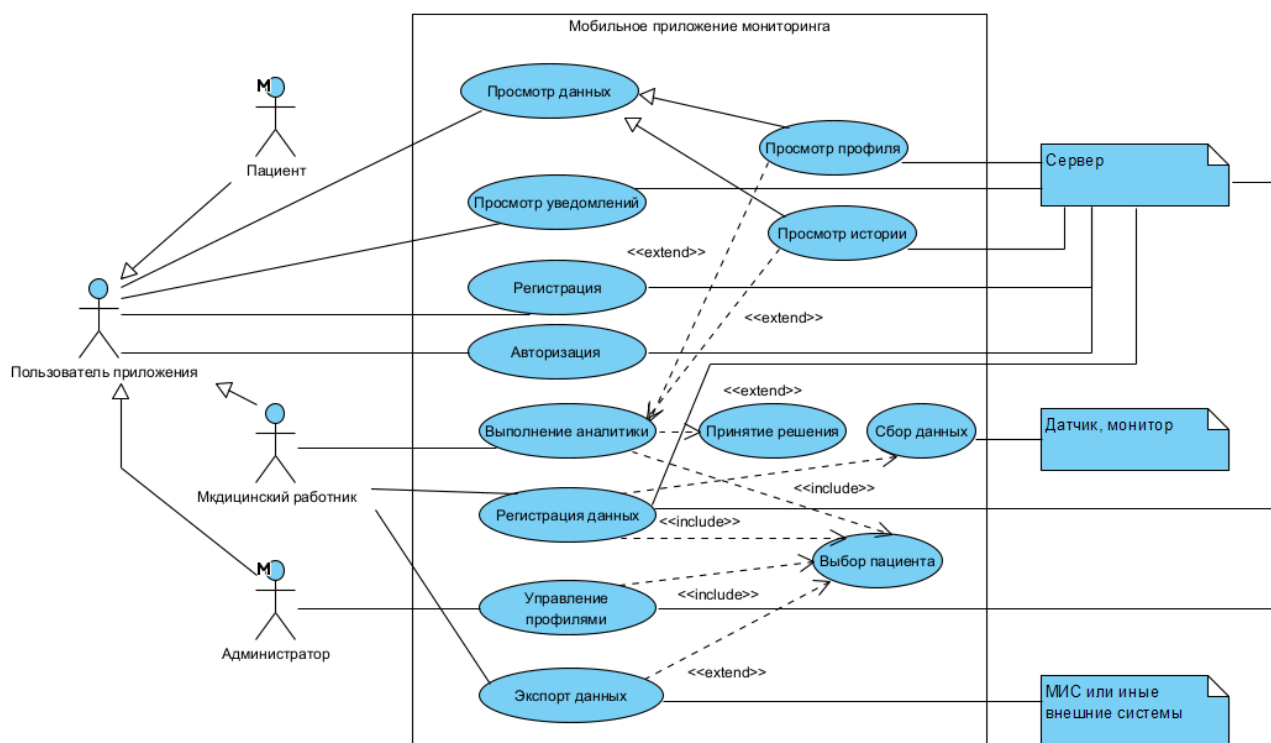


Рис. 1. Диаграмма вариантов использования при наблюдении пациента в стационаре с применением мобильных устройств, датчиков и мониторов

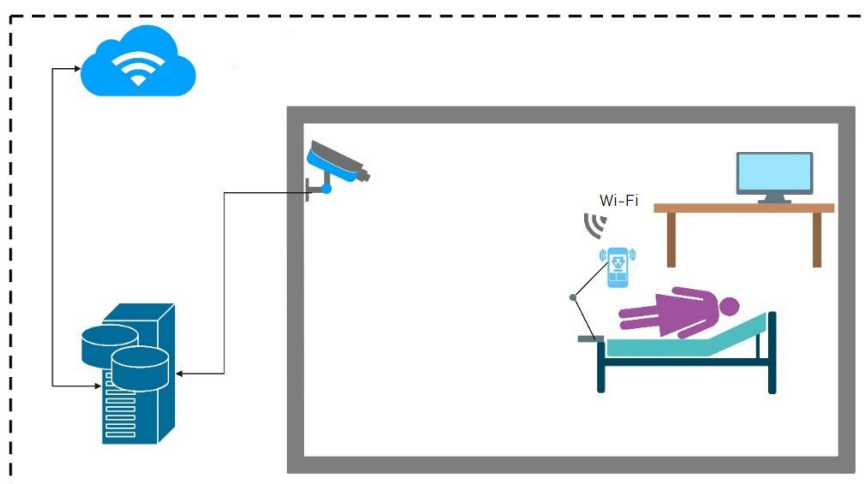


Рис.2. Схема автоматического визуального мониторинга



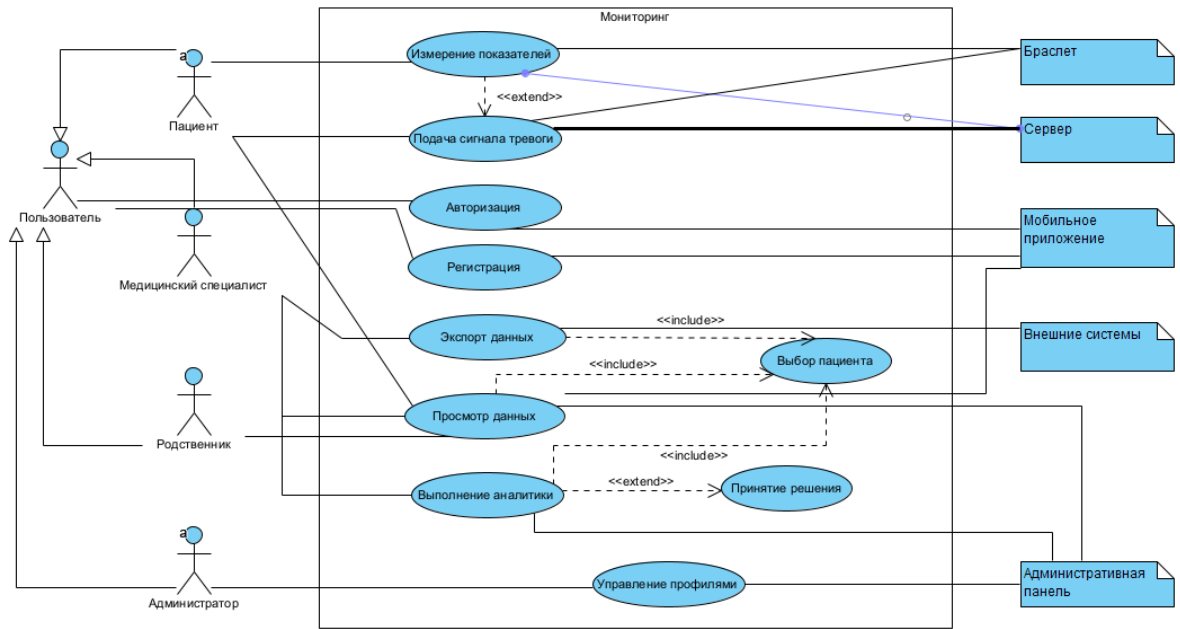


Рис. 3. Диаграмма вариантов использования при наблюдении пациента с использованием медицинского браслета

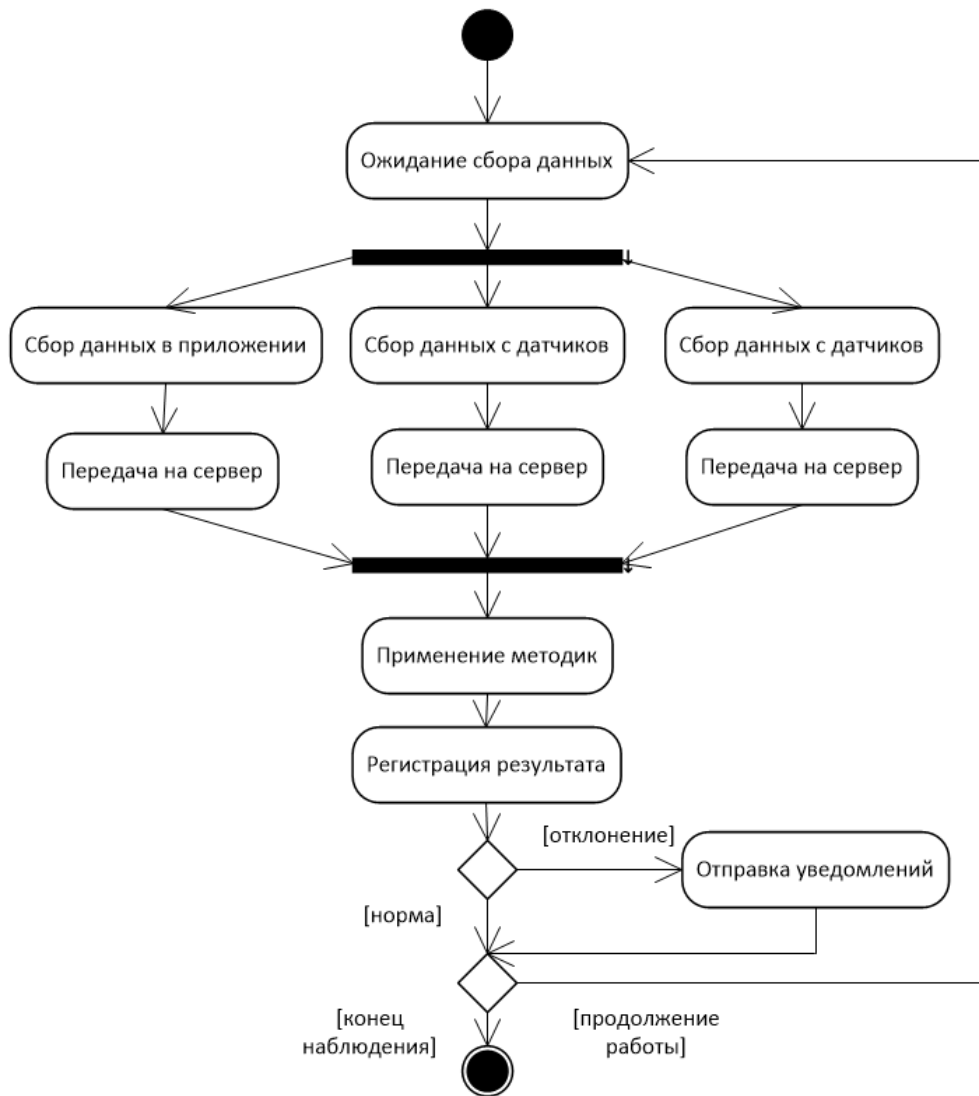


Рис. 4. Диаграмма активности для сбора данных с использованием мониторов, датчиков и мобильного приложения

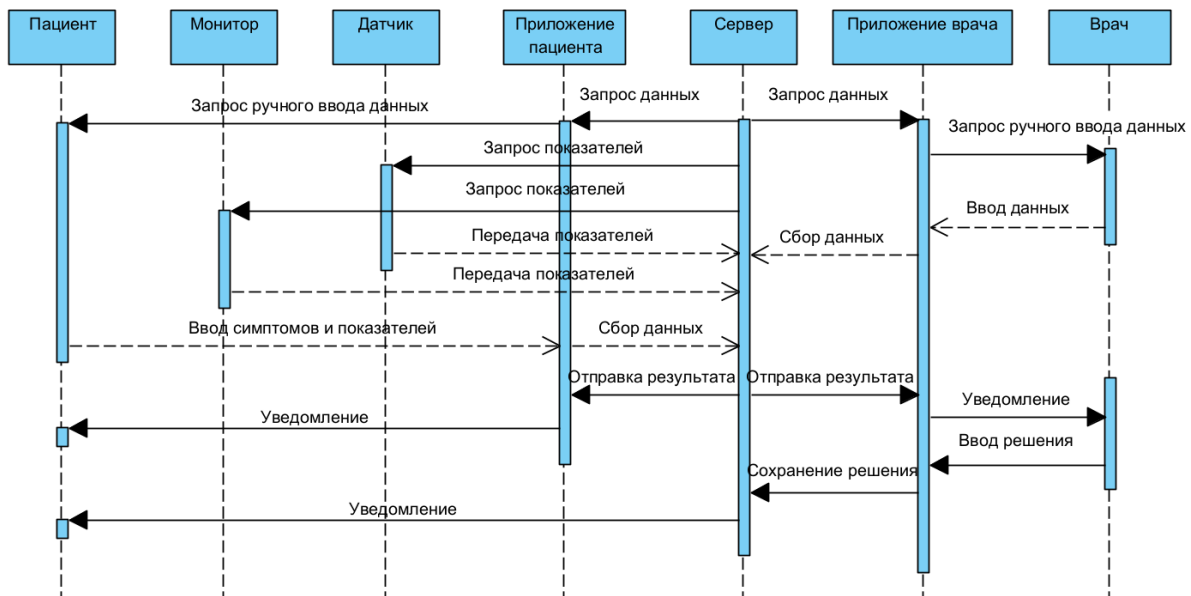


Рис. 5. Диаграмма последовательности для взаимодействия пациента и врача в УП

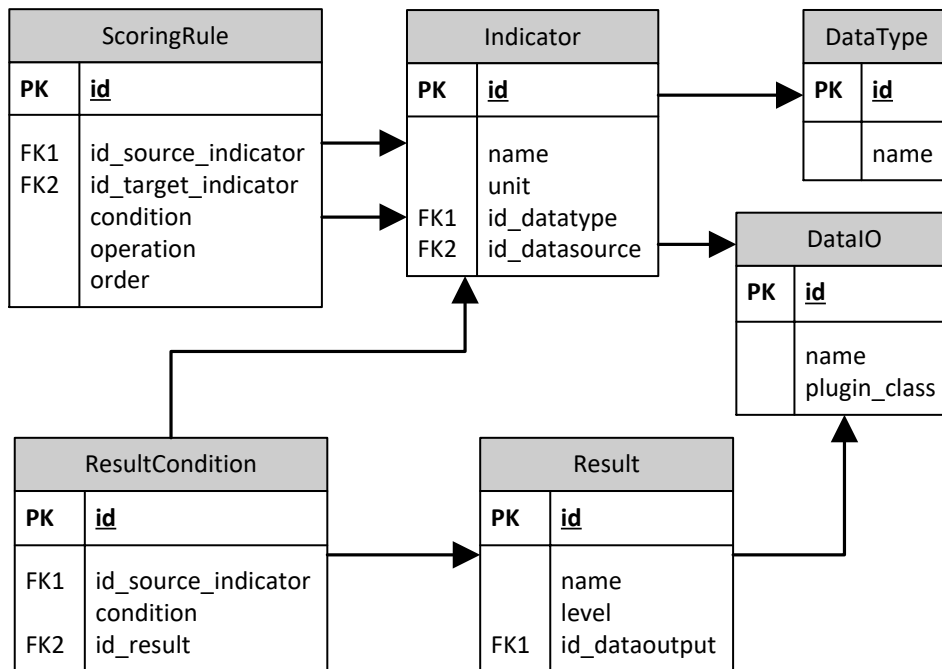


Рис. 6. Модель данных для обобщенной методики оценки

Таблица 1. Пример набора правил

| Исходный показатель  | Целевой показатель | Условие                | Операция  |
|--|--------------------|------------------------|-----------|
| [Температура]<br>(°C)  | [Общий балл]       | [S] >= 37 AND [S] < 38 | [T] + 1   |
|  |                    | [S] >= 38 AND [S] < 39 | [T] + 2   |
|  |                    | [S] >= 39              | [T] + 3   |
| [Гипергидроз]<br>(0=нет, 1=лёгкий,<br>2=умеренный,<br>3=тяжёлый) | [Общий балл]       | TRUE                   | [T] + [S] |

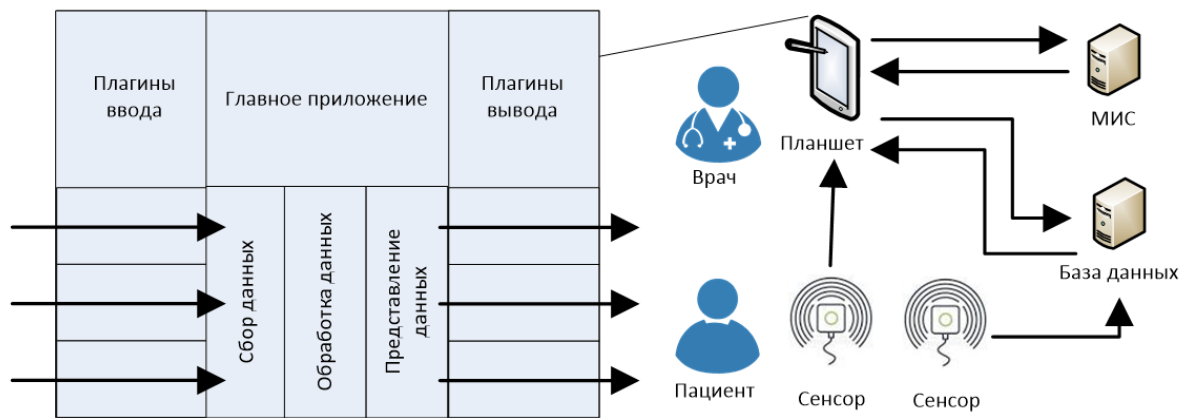


Рис. 7. Приложение как интегрированный компонент

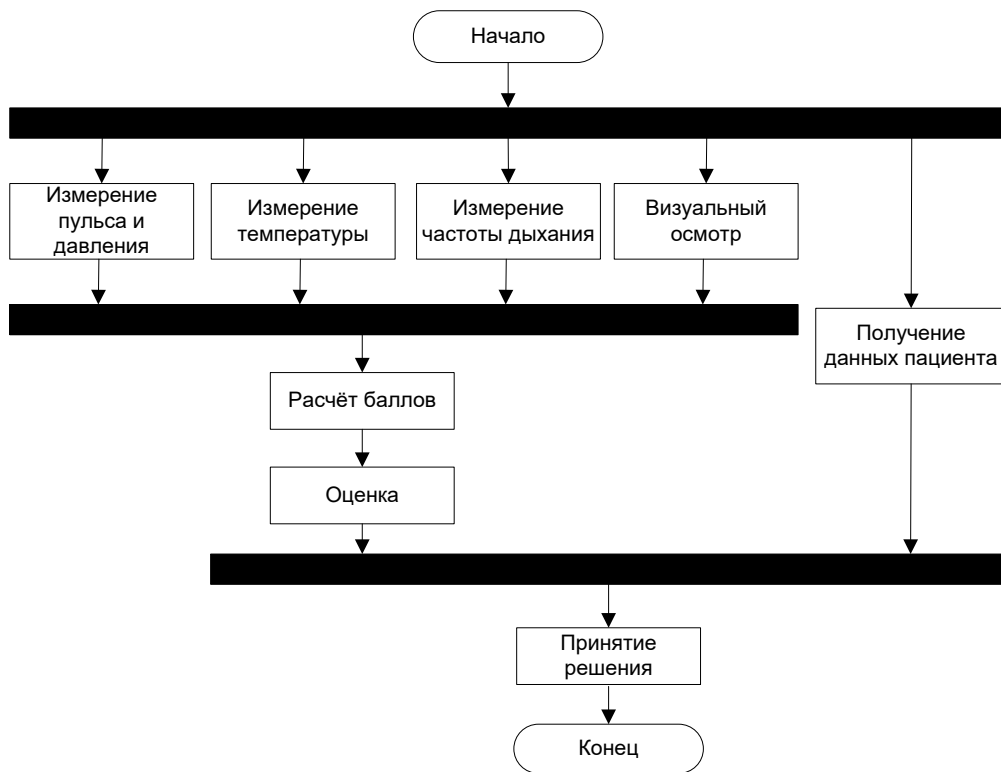


Рис. 8. Процесс диагностики и стратификации ПСГА

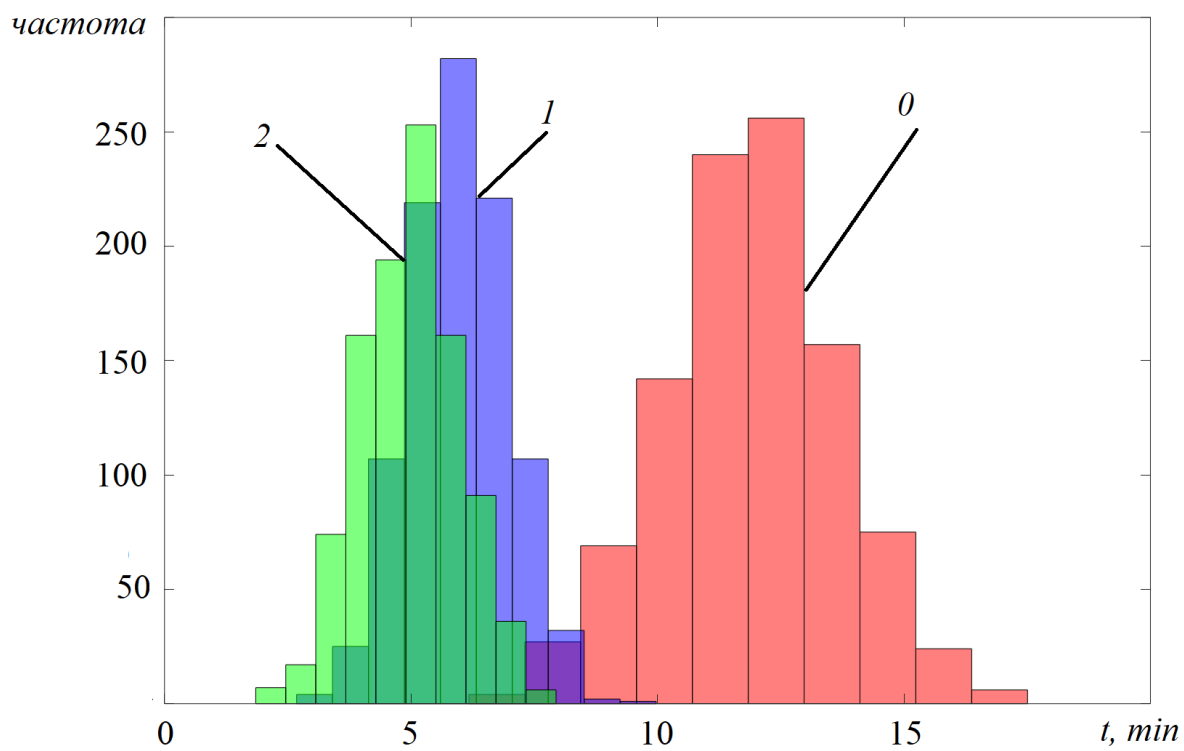


Рис. 9. Гистограммы времени сбора и обработки информации

Таблица 2. Логические роли модулей умной палаты

| Роль                | Параметры                     | Примеры   |
|---------------------|-------------------------------|---|
| Сбор данных         | Источник                      | Пациент   |
|                     | Показатель (симптом, признак) | Температура (°C)<br>Пульс (уд./мин)<br>Положение тела (выбор из списка)   |
|                     | Метод сбора данных            | Контактный<br>Визуальный  |
| Коммуникации        | Источник                      | Пациент<br>Медицинский специалист<br>Мобильное приложение<br>Сервер (PACS, МИС и т.п.)<br>Локальное хранилище данных                          |
|                     | Получатель                    | Пациент<br>Медицинский специалист<br>Мобильное приложение<br>Сервер (PACS, МИС и т.п.)<br>Локальное хранилище данных                          |
|                     | Интерфейс передачи данных     | Беспроводной (WiFi, Bluetooth, мобильные сети)<br>Проводной (Ethernet, USB)<br>Пользовательский интерфейс (экран)<br>Многомодальный интерфейс |
| Хранение данных     | Вид данных                    | Показатель (симптом, признак)<br>Результат анализа, исследования<br>Рекомендации<br>Решения   |
| Обработка данных    | Источник данных               | Сервер (PACS, МИС и т.п.)<br>Локальное хранилище данных   |
|                     | Получатель данных             | Сервер (PACS, МИС и т.п.)<br>Локальное хранилище данных   |
|                     | Алгоритм                      | Оценка на базе шкалы или иной детерминированной методики<br>Оценка на базе машинного обучения<br>Экспертная оценка                            |
| Принятие решений    | Алгоритм, методика            | Протокол диагностики и лечения  |
| Исполнение решений  | Источник данных               | Сервер (PACS, МИС и т.п.)<br>Локальное хранилище данных   |
|                     | Исполнительный механизм       | Устройство умной палаты   |
| Визуализация данных | Источник данных               | Сервер (PACS, МИС и т.п.)<br>Локальное хранилище данных   |
|                     | Интерфейс                     | Мобильное приложение<br>Диспетчерская панель<br>Электронная почта   |

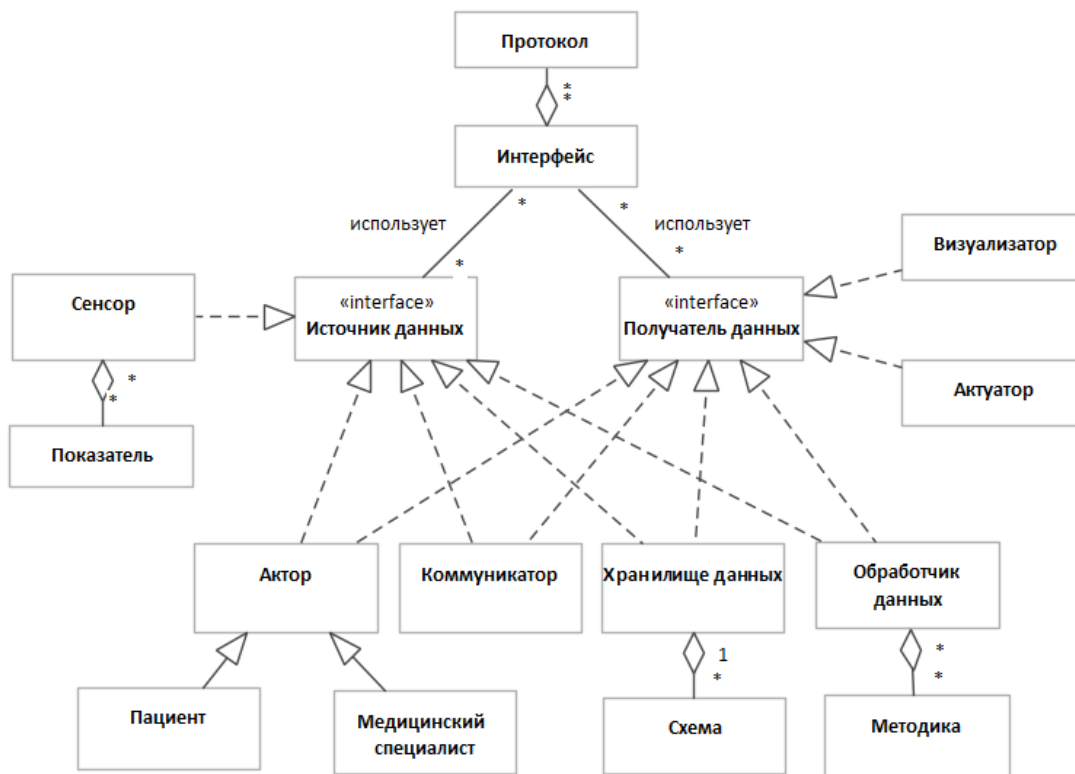


Рис. 10. Диаграмма классов для логического уровня функционирования умной палаты

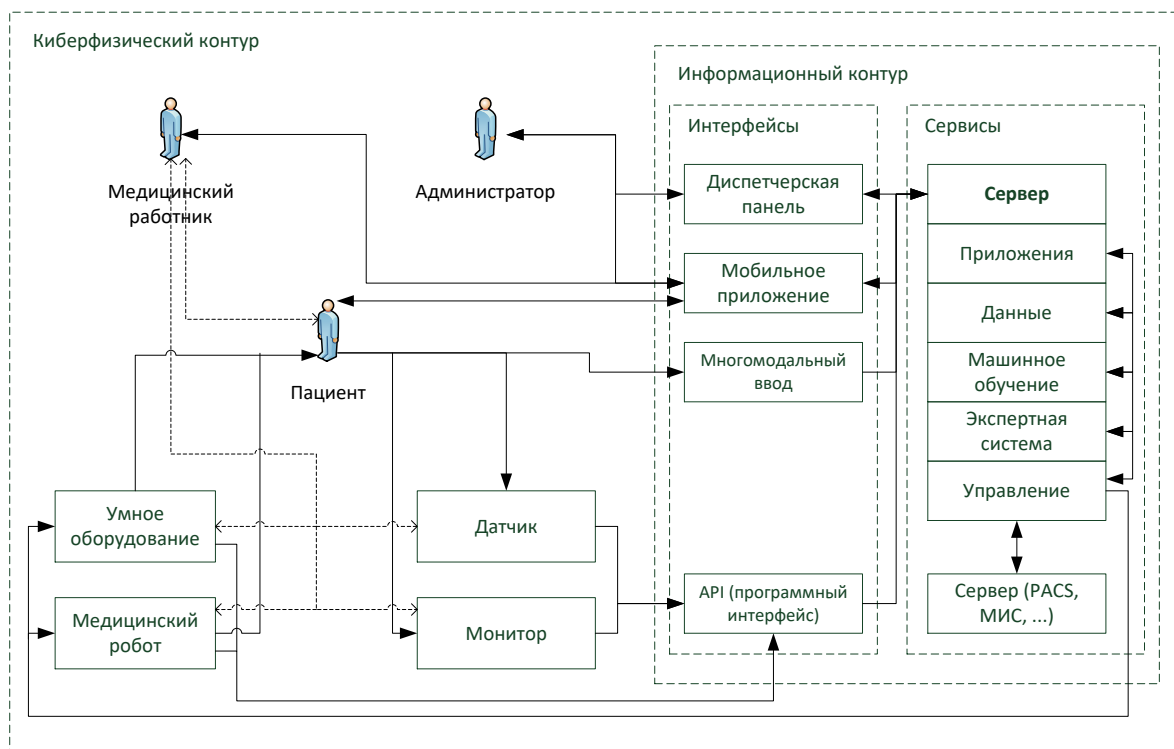


Рис. 11. Архитектура умной палаты



Таблица 3. Список команд API сервера

| Команда       | Описание                                 | Входные параметры  | Результат   |
|---------------|--|--|---|
| login         | Аутентификация                           | логин, пароль  | код ошибки, токен   |
| logout        | Завершение сеанса                        | токен  | код ошибки  |
| reset_request | Запрос ссылки/кода восстановления пароля | логин  | код ошибки; ссылка/код высылается на контактные реквизиты |
| reset_confirm | Восстановление пароля                    | логин, код   | код ошибки  |
| insert        | Создание объекта                         | токен, класс объекта, свойства объекта                   | код ошибки, ID созданного объекта                         |
| update        | Обновление свойств объекта               | токен, класс объекта, ID объекта, новые свойства объекта | код ошибки, список обновлённых свойств                    |
| delete        | Удаление объекта                         | токен, класс объекта, ID объекта                         | код ошибки  |
| get           | Получение данных об объекте по ID        | токен, класс объекта, ID объекта                         | код ошибки, свойства объекта                              |
| enumerate     | Получение списка объектов                | токен, класс объектов, фильтры, представление            | код ошибки, массив объектов                               |