

Министерство образования Республики Беларусь

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет информационных технологий и управления  
Кафедра интеллектуальных информационных технологий

*К защите допустить:*

Заведующий кафедрой ИИТ

\_\_\_\_\_ Д. В. Шункевич

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА  
к курсовому проекту по дисциплине  
«Проектирование баз знаний»  
на тему:

**РЕШАТЕЛЬ ЗАДАЧ ИНТЕЛЛЕКТУАЛЬНОЙ  
СИСТЕМЫ ПО МЕДИЦИНЕ**

БГУИР КП5 1-40 03 01 073 ПЗ

Студент гр. 221701  
Руководитель

И. Д. Телица  
Д. А. Сальников

Минск 2024

# СОДЕРЖАНИЕ

Перечень условных обозначений . . . . .	2
Введение . . . . .	3
1 Анализ подходов к решению проблемы . . . . .	4
1.1 Анализ предметной области . . . . .	4
1.2 Анализ аналогов разрабатываемой системы . . . . .	5
1.3 Анализ подходов к разработке . . . . .	11
1.3.1 Технология OSTIS . . . . .	11
1.3.2 Semantic Web . . . . .	12
1.4 Анализ подходов к управлению проектом . . . . .	13
1.4.1 Agile . . . . .	13
1.4.2 Waterfall (Водопад) . . . . .	14
1.4.3 Метод критического пути . . . . .	15
1.4.4 Scrum . . . . .	15
1.5 Вывод . . . . .	16
2 Проектирование системы . . . . .	18
2.1 Постановка задачи . . . . .	18
2.2 Потенциальные пользователи . . . . .	18
2.3 Архитектура системы . . . . .	20
2.3.1 Архитектура базы знаний . . . . .	20
2.3.2 Архитектура решателей задач . . . . .	21
2.3.3 Пример работы одного из агентов-решателей . . . . .	22
2.3.4 Архитектура веб-приложения . . . . .	23
2.4 Вывод . . . . .	26
3 Разработка системы . . . . .	27
3.1 Используемые средства разработки . . . . .	27
3.1.1 Метасистема OSTIS . . . . .	27
3.1.2 Docker . . . . .	28
3.1.3 Python . . . . .	28
3.2 Демонстрация базы знаний . . . . .	29
3.3 Демонстрация и тестирование агента анализа крови . . . . .	32
3.4 Демонстрация и тестирование веб-приложения . . . . .	33
3.5 Вывод . . . . .	34
Заключение . . . . .	35
Список использованных источников . . . . .	36

## ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ

SC — Semantic Code, семантический код;

SCs- Semantic Code String, строковый (линейный) семантический код;

SCg - Semantic Code Graph, графовый семантический код;

SCn - Semantic Code Natural, гипертекстовый семантический код;

БЗ - база знаний.

# ВВЕДЕНИЕ

В современном медицинском обслуживании информационные технологии играют ключевую роль в обеспечении качественного и доступного здравоохранения. В этой связи разработка и внедрение интеллектуальных систем, способных обрабатывать и анализировать медицинские данные, становятся всё более актуальными.

Современная медицинская практика сталкивается с рядом вызовов, таких как увеличение объема информации, требующей анализа и обработки, необходимость повышения эффективности диагностики и лечения, а также улучшения качества медицинского обслуживания для пациентов.

Одной из основных проблем, с которой сталкиваются врачи и медицинский персонал, является нехватка времени на анализ медицинских данных, принятие решений и ведение документации. Также часто возникают сложности в точной диагностике и выборе оптимального лечения из-за объема информации и необходимости учета индивидуальных особенностей каждого пациента.

Разработка и внедрение интеллектуальной системы по медицине, такой как медицинский ассистент, могут значительно облегчить задачи медицинского персонала, улучшить точность диагностики, оптимизировать процессы принятия решений и повысить качество медицинского обслуживания для пациентов.

Целью данного проекта является разработка и внедрение интеллектуальной системы по медицине в виде медицинского ассистента, который будет помогать врачам и медицинскому персоналу в анализе и обработке медицинских данных, поддерживая принятие обоснованных и эффективных медицинских решений.

## **Задачи для достижения поставленной цели:**

- изучение и анализ предметной области медицины;
- создание базы знаний;
- проектирование архитектуры и функциональности медицинского ассистента;
- реализация программного обеспечения медицинского ассистента;
- тестирование и оптимизация работы медицинского ассистента.

Важно отметить, что на сегодняшний день многие хотят знать, с помощью чего эффективнее всего можно контролировать своё здоровье и с помощью каких технологий можно обеспечить достаточно полную диагностику здоровья и реальных актуальных рисков заболеваний, независимо от уровня понимания здоровья человеком.

# 1 АНАЛИЗ ПОДХОДОВ К РЕШЕНИЮ ПРОБЛЕМЫ

## 1.1 Анализ предметной области

Область медицины охватывает широкий спектр дисциплин, включая здравоохранение, диагностику, лечение и профилактику заболеваний. Среди множества проблем в этой области особенно актуальной является необходимость эффективного управления медицинской информацией. С ростом объема данных и сложности медицинских процессов возникает потребность в создании интеллектуальных систем, способных обрабатывать, анализировать и использовать эту информацию для поддержки принятия решений в медицинской практике.

Эта проблема актуальна по нескольким причинам:

- Различные симптомы. Множество заболеваний человеческого организма проявляются через симптомы, которые могут быть схожи между собой, что затрудняет точную диагностику. Например, боль, слабость и повышенная температура могут быть признаками различных заболеваний, таких как грипп, пневмония или инфекционные заболевания желудочно-кишечного тракта. Это усложняет задачу врачей при определении конкретного заболевания и назначении соответствующего лечения.

- Недостаток данных. Для обучения моделей искусственного интеллекта необходимо большое количество данных, но доступ к таким данным ограничен из-за конфиденциальности и недостатка стандартизации.

- Сложность обработки изображений. Сложность обработки медицинских изображений также является проблемой, особенно в контексте анализа органов и тканей человеческого организма. Точная интерпретация медицинских снимков требует высокой экспертизы и может быть подвержена ошибкам, что усложняет автоматизацию этого процесса с помощью алгоритмов машинного обучения.

- Недостаточная точность алгоритмов. Некоторые алгоритмы могут быть недостаточно точными для диагностики сложных и разнообразных заболеваний человеческого организма, что ограничивает их применимость в клинической практике.

- Различие заболеваний. Болезни часто проявляются в различных формах и стадиях, что усложняет создание универсальных моделей для их диагностики. Таким образом, необходимо разработать подходы, которые учитывают эту вариабельность для более точной диагностики и прогнозирования.

Таким образом, хотя в области искусственного интеллекта в медицине достигнуты значительные успехи, точная диагностика заболеваний остается сложной задачей, требующей дальнейших исследований и улучшений в

алгоритмах и технологиях.

## 1.2 Анализ аналогов разрабатываемой системы

### 1) Woebot(Рис. 1.1)[1]

Компания Woebot Labs запустила в эксплуатацию сервис, который они называют первым в мире автоматизированным чатом для помощи людям, страдающим от психиатрических заболеваний.

Эта система рассматривается ее создателями как доступное, персонализированное и удобное лечение для миллионов людей, страдающих от психических заболеваний, связанных с изменениями настроения, такими как депрессия и тревожность.

Чатбот использует методы когнитивно-поведенческой терапии, чтобы выслушать и дать совет любому, кто обратится к боту за помощью. С помощью разговора на обычном "человеческом" языке приложение позволяет пользователю получить эмоциональную поддержку, которая ему нужна, и научиться практическим методам, позволяющим изменить ход мыслей о происходящих в его жизни явлениях.

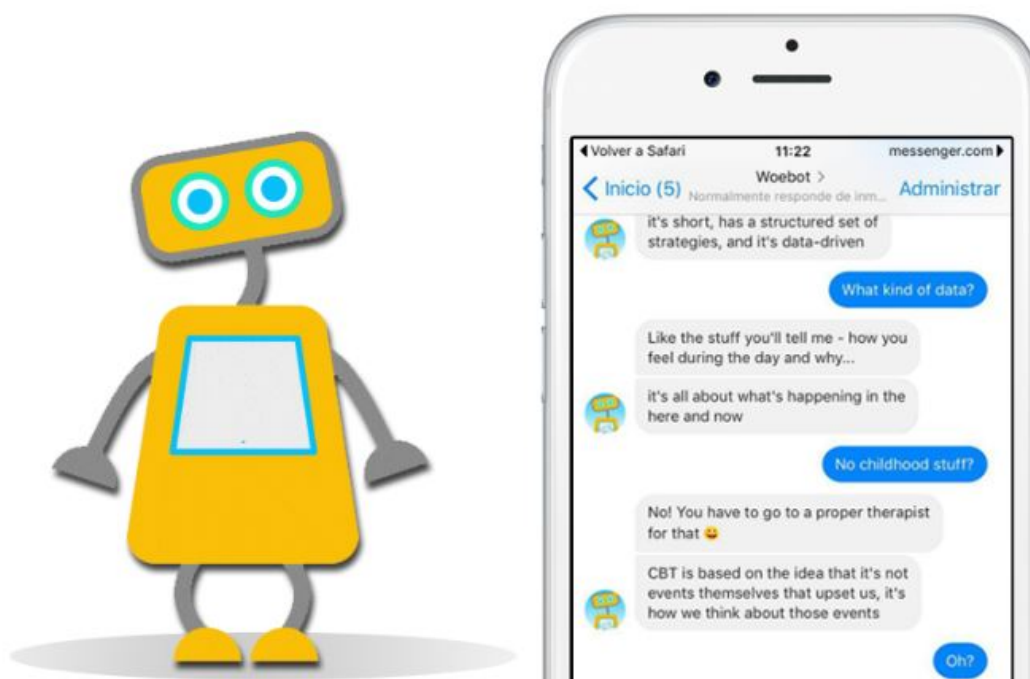


Рисунок 1.1 – Пример использования помощника Woebot

### Преимущества:

- сайт предлагает широкий спектр информации о различных аспектах здоровья, включая симптомы, лечение и профилактику различных заболеваний, питание, физическую активность и т.д.;
- информация о заболеваниях, симптомах и лечении представлена в достаточной степени подробно;

- наличие статей и руководств, которые могут быть полезны для понимания различных аспектов здоровья и медицины;
- сайт имеет простой и удобный интерфейс, что делает его доступным для широкого круга пользователей;
- информация представлена в виде статей, списков симптомов, советов по лечению и т.д., что упрощает поиск нужной информации;
- наличие поисковой строки позволяет пользователям быстро находить интересующие их темы и статьи.

#### **Недостатки:**

- сайт не предоставляет явной информации о своих источниках данных и контенте;
  - в большинстве статей нет ссылок на научные исследования или медицинские источники, что может вызывать сомнения в достоверности представленной информации;
  - отсутствие анонсированных авторов статей или специалистов, которые подтверждали бы их квалификацию, также может влиять на восприятие сайта как недостаточно достоверного источника медицинской информации.
- В целом, сайт Woebot предоставляет обширную информацию о различных аспектах здоровья и медицины, что может быть полезным для пользователей, заинтересованных в самостоятельном изучении этих тем. Однако отсутствие явной информации о источниках и недостаточная ссылочность на научные исследования может вызвать вопросы о достоверности представленной информации. Поэтому при использовании данного ресурса важно проявлять осторожность и проверять информацию в других источниках, особенно если речь идет о серьезных медицинских вопросах или лечении. Также недостатком данного помощника является то, что он рассчитан на психологическую помощь, а консультации по вопросам заболеваний, например, мышечных тканей или легких путей он не способен проводить.

#### **2) HealthTap(Рис. 1.2) [2]**

HealthTap - это инновационная медицинская онлайн-платформа, предоставляющая широкий спектр медицинских консультаций и услуг через виртуальный чат. Эта платформа была разработана для того, чтобы сделать медицинские консультации более доступными и удобными для пользователей, независимо от их местоположения.

Пользователи HealthTap могут получать квалифицированные медицинские советы и консультации от настоящих медицинских специалистов в режиме реального времени. Это позволяет пользователям получать экспертные мнения по широкому спектру медицинских вопросов, начиная от общих симптомов и заканчивая конкретными запросами.

#### **Преимущества:**

- HealthTap предлагает обширный объем информации о здоровье и медицине, включая симптомы, диагностику, лечение различных заболева-

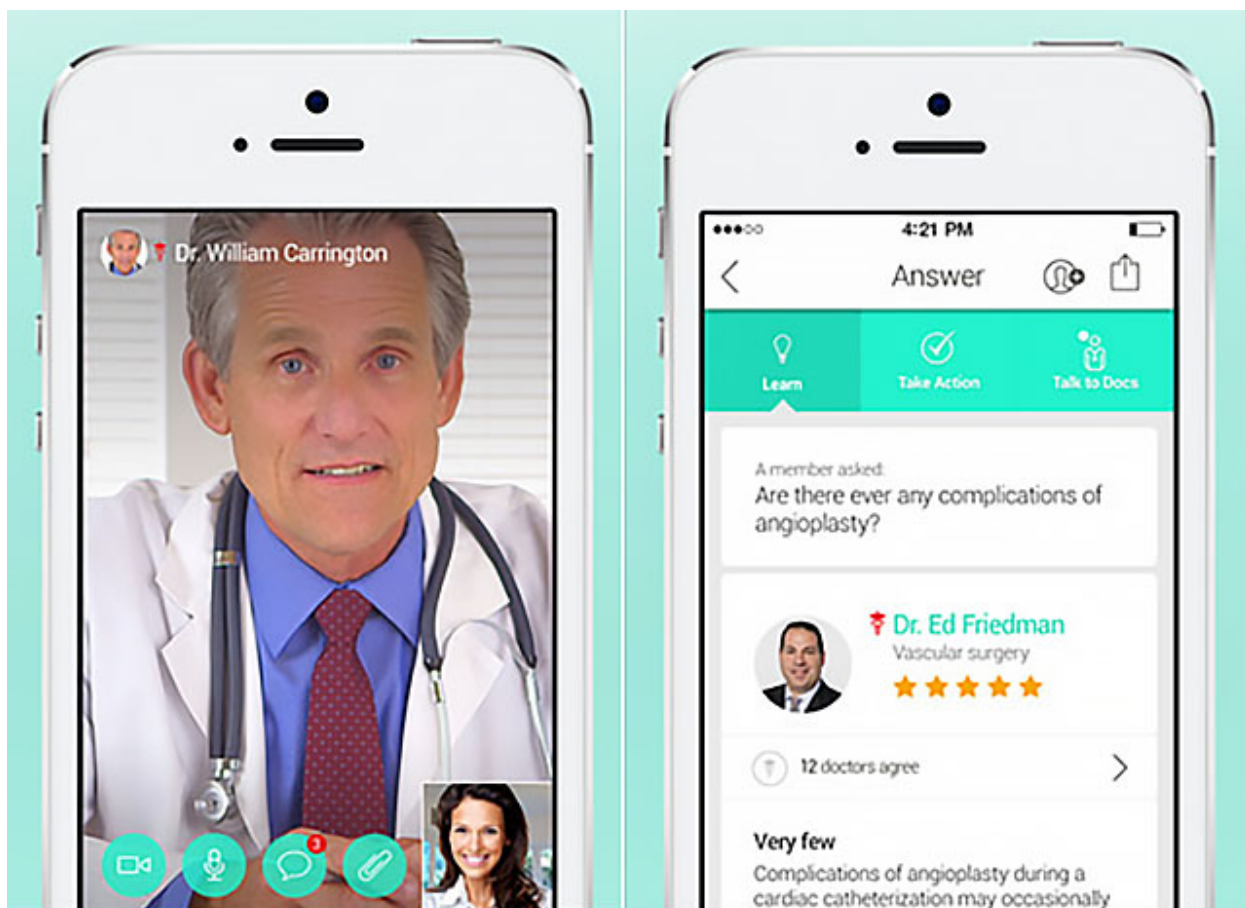


Рисунок 1.2 – Пример использования помощника HealthTap

ний, советы по здоровому образу жизни и т. д..

- сайт содержит разнообразные статьи, ответы на вопросы пользователей, видеоинструкции, часто задаваемые вопросы и т. д., что делает его информативным и полезным ресурсом для пользователей, заинтересованных в здоровье.

- Конфиденциальность и безопасность данных. Защита конфиденциальности данных пользователей находится в приоритете, обеспечивая им уверенность в безопасности информации.

- Статьи и ответы на вопросы часто сопровождаются ссылками на научные исследования, клинические рекомендации и другие авторитетные источники, что способствует доверию к контенту.

- Сайт имеет простой и интуитивно понятный интерфейс, что облегчает поиск и чтение информации для пользователей.

- Наличие функции поиска позволяет пользователям быстро находить нужные им темы и статьи.

- HealthTap также предоставляет возможность задавать вопросы медицинским экспертам и получать консультации в режиме онлайн, что делает его полезным ресурсом для получения персонализированной медицинской информации.



## **Недостатки:**

– Платная услуга в большинстве случаев. Один из недостатков HealthTap заключается в том, что доступ к его услугам часто предполагает плату, что может ограничивать доступность для пользователей с ограниченными средствами.

– Не всегда доступно вне США. Географические ограничения могут сделать HealthTap менее доступным для пользователей за пределами Соединенных Штатов.

Сайт HealthTap представляет собой полезный и информативный ресурс о здоровье и медицине, который предлагает широкий спектр информации для пользователей.

Утверждение о том, что контент разрабатывается медицинскими специалистами и ссылки на научные исследования улучшают доверие к представленной информации.

Однако, как и в случае с любым другим медицинским ресурсом, важно проявлять осторожность и проверять информацию в других независимых источниках, особенно при решении серьезных медицинских вопросов.

### **3) Ada Health(Рис. 1.3)[3]**

Платформа Ada, доступная через бесплатное приложение для смартфонов, проводит пользователей через серию вопросов о демографических характеристиках, основных параметрах здоровья и любых симптомах, которые они испытывают, а также о начале и тяжести этих симптомов. Затем приложение использует алгоритм искусственного интеллекта для сравнения данных, предоставленных пользователями, с обширной медицинской библиотекой Ada, чтобы составить список заболеваний, которые могут быть причиной симптомов.

Алгоритм был обучен специфике сотен медицинских заболеваний, что позволяет ему выявлять мельчайшие различия между симптомами и основными факторами здоровья, связанными с заболеваниями, которые могут проявляться схожим образом.

Разработчики утверждают, что в будущем Ada станет стандартным диагностическим инструментом для врачей. Это уже происходит - пользователи могут поделиться своей оценкой состояния здоровья со своим врачом или (в Великобритании) они могут проконсультироваться с квалифицированным терапевтом через функцию "Чат с врачом". Ada также станет гораздо более постоянным спутником здоровья, помогая пациентам и врачам разумно отслеживать данные о здоровье в течение длительного времени, чтобы обеспечить предиктивное и проактивное лечение.

## **Преимущества:**

– Ada Health обеспечивает высокую точность диагнозов, основанных на симптомах и медицинских данных. Это позволяет пользователям получать надежные информацию о своем здоровье.

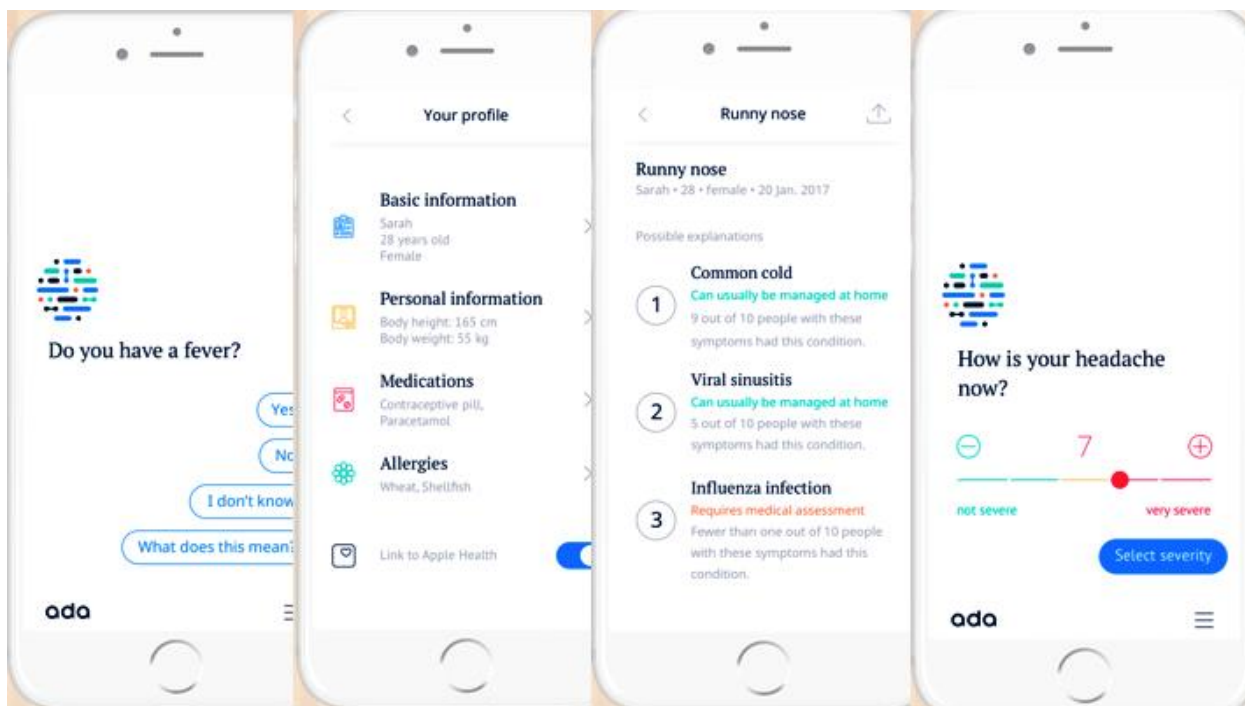


Рисунок 1.3 – Пример использования помощника Ada Health

- Сайт также предоставляет инструмент для самодиагностики, позволяющий пользователям оценить свои симптомы и получить предварительные рекомендации по дальнейшим действиям.

- Возможность ввода симптомов и получения рекомендаций. Пользователи могут легко и подробно описывать свои симптомы, что делает систему доступной и интуитивно понятной.

- Интеграция с базой знаний и медицинскими данными.

- Сайт имеет простой и интуитивно понятный интерфейс, что облегчает навигацию и поиск информации для пользователей в базе знаний с актуальными медицинскими данными, что повышает информативность и надежность предоставляемой информации.

- Ada предлагает информацию о различных аспектах здоровья, включая симптомы, возможные причины, диагностику и лечение различных заболеваний.

- Ada утверждает, что его контент разрабатывается медицинскими экспертами и основан на актуальных клинических рекомендациях и научных исследованиях.

- Статьи и информация часто сопровождаются ссылками на медицинские источники и авторитетные медицинские организации, что способствует доверию к представленной информации.

- Функция самодиагностики позволяет пользователям оценить свои симптомы и получить предварительные рекомендации по медицинской помощи.

## Недостатки:

– Ограничения в медицинских консультациях. Ada Health не заменяет полноценных врачей и может быть ограничена в предоставлении консультаций в случае серьезных заболеваний или нестандартных ситуаций

Сайт Ada Health представляет собой полезный и информативный ресурс о здоровье и медицине, который предлагает широкий спектр информации для пользователей. Утверждение о том, что контент разрабатывается медицинскими экспертами, и ссылки на научные исследования улучшают доверие к представленной информации.

После рассмотрения всех аналогов и выявления их достоинств и недостатков, можно сделать вывод о том, что ни один из аналогов полностью не удовлетворяет всем требованиям, которые должны быть в нашей медицинской системе.

Исходя из анализа аналогов, можно выделить следующие основные требования к медицинской системе:

### 1) Наполняемость базы знаний (БЗ):

– система должна иметь обширную и достоверную базу знаний о различных аспектах здоровья и медицины, включая симптомы, диагностику, лечение и профилактику различных заболеваний;

– информация должна быть представлена в понятной и доступной форме, чтобы пользователи могли легко находить нужную информацию.

### 2) Достоверность источников:

– система должна основываться на достоверных источниках данных, таких как клинические рекомендации, научные исследования и медицинские практики;

– информация должна быть проверена медицинскими экспертами и специалистами, чтобы обеспечить высокую точность и достоверность.

### 3) Доступ к информации:

– система должна предоставлять удобный и интуитивно понятный интерфейс для пользователя, облегчающий поиск и чтение медицинской информации;

– наличие функции поиска и категоризации информации позволит пользователям быстро находить нужные им темы и статьи;

– возможность получения персонализированных консультаций от медицинских экспертов или использования инструментов самодиагностики может повысить ценность системы для пользователей.

### 4) Безопасность данных:

– Система должна обеспечивать конфиденциальность и безопасность медицинских данных пользователей, соблюдая соответствующие стандарты и законодательство о защите данных.

### 1.3 Анализ подходов к разработке

Для разработки медицинского ассистента нужно создать базу знаний, для реализации которой могут использоваться:

- Технология OSTIS
- Semantic Web

#### 1.3.1 Технология OSTIS

OSTIS - открытые семантические технологии проектирования интеллектуальных систем. OSTIS позволяет представлять знания в виде семантических сетей, используя SC-код для пополнения базы знаний и машины обработки знаний. SC-код – универсальный способ кодирования информации в технологии OSTIS. База знаний OSTIS может описывать любой вид знаний, а её решатель задач основан на многоагентном подходе и позволяет легко комбинировать любые модели решения задач. Интерфейс данной системы представляет собой подсистему со своей базой знаний и решателем задач. Интерфейс может быть описан с помощью SC-кода. SC-код и технология OSTIS являются ключевыми инструментами для моделирования и обработки знаний. Знания представляются с помощью SC-кода, где в качестве семантики используется теория множеств, а в качестве синтаксиса – теория графов. Разработанные фрагменты будут представлять собой семантические сети. Основные положения:

- база знаний OSTIS может описывать любой вид знаний;
- решатель задач OSTIS основан на многоагентном подходе и позволяет легко комбинировать любые модели решения задач;
- интерфейс ostis-системы представляет собой подсистему со своей БЗ и решателем задач (также может быть описан с помощью SC-кода);
- использование универсального способа представления (кодирования) информации, получившего название SC-код.

Алфавит SC-кода представляет собой базовое синтаксическое разбиение множества sc-элементов на следующие виды:

- sc-узел;
- sc-ребро;
- sc-дуга общего вида;
- sc-дуга основного вида

SC-код позволяет представлять знания в унифицированном виде. В рамках разработки веб-приложения был выбран SC-код, поскольку он обеспечивает формальность, гибкость и эффективность при разработке интеллектуальных систем:

- Формальная семантика. SC-код основан на формальной семантике. Это позволяет точно определить смысл и логику знаний, благодаря чему

OSTIS может выполнять рассуждения и выводить новые знания на основе логических правил

- Гибкость и масштабируемость. SC-код предоставляет гибкие возможности для моделирования сложных знаний и структур. Можно описывать объекты, отношения и атрибуты в SC-коде, а также создавать иерархии и композиции для организации знаний, что позволяет разрабатывать масштабируемые системы с высоким уровнем абстракции.

- Интеграция с другими языками программирования. SC-код может быть использован в сочетании с другими языками программирования, такими как Python, Java или C++, что позволяет разработчикам использовать мощные возможности этих языков в сочетании с семантическими возможностями OSTIS.

- Поддержка семантического поиска. Благодаря использованию SC-кода, OSTIS предоставляет возможности семантического поиска, которые позволяют находить связанные знания на основе их содержания и семантической связи, что делает поиск более точным и эффективным по сравнению с традиционными методами поиска.

SC-код позволяет представлять знания в унифицированном виде.

### 1.3.2 Semantic Web

Семантическая паутина — надстройка над существующей Всемирной паутиной, придуманная для того, чтобы сделать размещаемую в Интернете информацию пригодной для машинной обработки. Доступная в сети информация удобна для прочтения и понимания человеком.

Semantic Web создана для того, чтобы сделать информацию пригодной для автоматического анализа, синтеза выводов и преобразования данных и сделанных на их основе заключений в различные представления, полезные на практике.

Машинная обработка возможна благодаря двум характеристикам Semantic Web:

- наличие URI;
- использование семантических сетей и онтологий.

URI — унифицированный идентификатор ресурса или адрес, используемый для указания ссылок на какой-либо объект (веб-страницу, файл или ящик электронной почты). URI используются для однозначного именования объектов. Иными словами, каждый объект глобальной семантической сети имеет уникальный URI.

Отдельные URI создают не только для страниц, но и для объектов реального мира (людей, городов, художественных произведений и так далее), и даже для абстрактных понятий (например, «имя», «должность», «цвет»). Благодаря уникальности URI одни и те же предметы можно называть

одинаково в разных местах семантической паутины. Используя URI, можно собирать информацию об одном предмете из разных мест. Рекомендуется включать в адрес URI название одного из протоколов Всемирной паутины (HTTP или HTTPS).

Описание желательно предоставлять в двух форматах:

- в формате, удобном для чтения человеком;
- в формате, удобном для чтения машиной.

В качестве формата, удобного для чтения машиной, используется язык RDF. Язык RDF позволяет описывать структуру семантической сети в виде графа. Каждому узлу и каждой дуге графа можно назначить отдельный URI. Утверждения, записанные на языке RDF, можно интерпретировать с помощью онтологий.

Для создания онтологий рекомендуют использовать языки RDF Schema и OWL. Онтологии создаются для получения из данных логических заключений, а в их основе лежат математические формализмы, называемые дескрипционными логиками.

В качестве редактора онтологий и фреймворка для построения базы знаний используется Protégé.

Платформа Protégé поддерживает два основных способа моделирования онтологий посредством редакторов Protégé-Frames и Protégé-OWL.

Онтологии, построенные в Protégé, могут быть экспортированы во множество форматов, включая RDF (RDF Schema), OWL и XML Schema.

Protégé поддерживается значительным сообществом, состоящим из разработчиков и учёных, правительственных и корпоративных пользователей, использующих его для решения задач, связанных со знаниями, в таких разнообразных областях, как биомедицина, сбор знаний и корпоративное моделирование.

## **1.4 Анализ подходов к управлению проектом**

Также для обеспечения эффективного выполнения проекта необходимо воспользоваться одной из методологий управления проектом. Выберем наиболее подходящую из:

- Waterfall (Водопад)
- Метод критического пути
- Agile
- Scrum

### **1.4.1 Agile**

Методология управления проектами Agile является одним из наиболее распространённых подходов к управлению проектами. Однако, по сути, Agile

— принцип управления проектами.

В основе Agile лежат следующие принципы:

- совместная работа
- скорость и эффективность
- итеративность и ориентация на данные
- личность важнее процессов

Когда дело доходит до внедрения Agile, команды часто выбирают определённую методологию, которую они будут использовать наряду с принципами Agile. Это может быть Scrum, Канбан, экстремальное программирование, Crystal или даже Scrumban. Делается это потому, что использование методологии Agile вместе с более подробно сформулированным подходом позволяет сформировать законченную философию управления проектом и практический план для достижения отличных результатов.

**Кому подойдёт.** Систему Agile может использовать практически любая команда, потому что в её основе лежат довольно универсальные принципы. Самое сложное здесь — решить, какую методологию использовать совместно с этим подходом.

### 1.4.2 Waterfall (Водопад)

Каскадная модель управления, также известная как «водопад», довольно популярна. «Водопад» — это настоящая методология с очень чёткими правилами. Каскадная методология, также известная как цикл разработки программного обеспечения (ЦРПО) представляет собой линейный процесс, в котором работа ниспадает каскадом (как водопад) и организована в последовательном порядке.

При использовании этого подхода все рабочие задачи связываются друг с другом зависимостями. Это означает, что для того, чтобы начать работу над задачей, должна быть выполнена предшествующая ей задача. Благодаря этому работа идёт по плану, а также обеспечивается чёткий обмен информацией в течение всего процесса.

Хотя некоторые современные организации считают данный подход устаревшим, эта методология отлично подходит для создания предсказуемого и хорошо продуманного плана проекта.

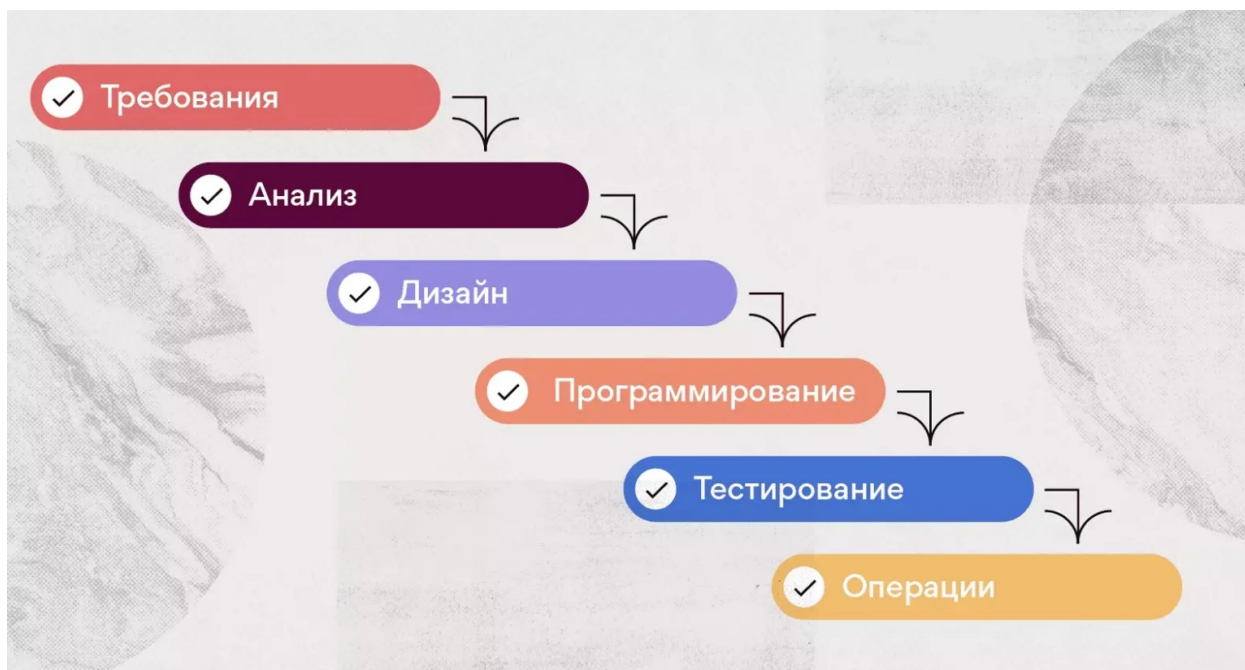


Рисунок 1.4 – Пример модели управления Waterfall

**Кому подойдёт.** Поскольку каскадная методология управления является очень подробной, она хорошо подходит для работы над крупными проектами с множеством заинтересованных сторон. Эта модель обеспечивает наличие чёткой информации о необходимых действиях в течение всего проекта и зависимостей, позволяющих отследить работу, которую следует выполнить для достижения целей.

### 1.4.3 Метод критического пути

Метод критического пути применяется для определения критически важных задач в проекте и планирования работы над ними. Сюда входит создание зависимостей между задачами, отслеживание целей проекта и хода работ над ним, определение приоритета результатов и управление сроками — все это очень похоже на структуру разбивки работ.

Цель этой методологии состоит в надлежащем управлении успешными проектами в масштабе так, чтобы вехи и ожидаемые результаты были размечены правильно.

**Кому подойдёт.** Метод критического пути лучше всего подходит для небольших и средних проектов и команд. Связано это с тем, что в крупных проектах множество ожидаемых результатов и заинтересованных сторон, а метод критического пути не предназначен для сложных проектов.

### 1.4.4 Scrum

Методология Scrum предусматривает использование коротких «спринтов», из которых формируется цикл проекта. Эти промежутки длятся от



одной до двух недель и рассчитаны на команды в составе не более 10 человек. Это основное отличие от каскадной методологии, где отдельные задачи связываются друг с другом зависимостями.

У Scrum много уникальных особенностей, одной из которых является наличие мастера Scrum или, другими словами, руководителя проекта, который проводит ежедневные Scrum-совещания, демонстрации, спринты и ретроспективы после окончания спринтов. Все эти встречи нужны для общения ключевых участников проекта и своевременного выполнения задач.

Несмотря на то, что технически Scrum является самостоятельной методологией управления проектами, её часто ассоциируют с системой Agile. Связано это с тем, что два этих подхода объединены общими принципами, в том числе принципом важности совместной работы и тем, что личность ценится выше процессов.

**Кому подойдёт.** Командам, применяющим подход Agile, также следует прибегнуть к методологии Scrum, или, по крайней мере, попробовать её в действии. Так как спринты проводятся для небольших команд, этот подход работает как для небольших, так и для крупных коллективов.

## 1.5 Вывод

Проведенный анализ позволяет сделать вывод об актуальности и важности разработки и внедрения системы медицинского ассистента. Такие системы становятся все более необходимыми в нашем цифровом мире, где доступ к информации и консультациям в области здоровья и медицины играют ключевую роль для обеспечения качественного и своевременного медицинского обслуживания. Медицинские ассистенты предоставляют пользователю доступ к обширной базе знаний, консультациям медицинских экспертов и инструментам самодиагностики, что повышает уровень информированности и помогает в принятии обоснованных решений по вопросам здоровья.

Исходя из проведенного анализа можно сделать заключение о необходимости следующих требований к базе знаний системы медицинского ассистента:

- Обширность и достоверность информации. База знаний должна содержать подробную и актуальную информацию о различных аспектах здоровья и медицины, проверенную медицинскими экспертами и основанную на научных исследованиях и клинических рекомендациях.

- Удобство доступа и навигации. Пользователям должен быть предоставлен удобный интерфейс для поиска и чтения медицинской информации, а также функции категоризации и фильтрации контента для быстрого доступа к нужным темам.

- Поддержка персонализированных консультаций. База знаний долж-

на поддерживать возможность получения персонализированных консультаций от медицинских экспертов, а также предоставления инструментов самодиагностики для пользователей.

Также на основе проведенного анализа можно сделать заключение о необходимости следующих требований к разрабатываемым решателям задач:

1) Достоверность информации. Решатель задач должен реализовывать свою работу на основе разработанной базы знаний, содержащей подробную и актуальную информацию о различных аспектах здоровья и медицины, проверенную медицинскими экспертами и основанную на научных исследованиях и клинических рекомендациях, а также предоставлять на ее основе достоверную информацию о результате работы.

2) Удобство доступа. Пользователям должен быть предоставлен удобный интерфейс для вызова и отслеживания результатов работы вызываемых решателей задач в интеллектуальной системе.

3) Критический подход к интерпретации данных. Пользователю должна быть предоставлена информация о необходимости прохождения дальнейшей диагностики у соответствующего профильного специалиста в связи с отсутствием возможности поставить в достаточной степени достоверный результат по результатам самодиагностики.

Для разработки системы была выбрана технология OSTIS и выделены её преимущества. Эта технология поможет улучшить функциональность и эффективность медицинского ассистента, обеспечивая высокую точность диагностики, скорость обработки данных, автоматизацию процессов, адаптивность к изменениям и безопасность информации.

Наиболее предпочтительным для разработки и реализации данного проекта его участниками был выбран подход к управлению 'Метод критического пути', включающий в себя принципы Agile и элементы Scrum, такие как наличие руководителя проекта и ретроспективы по результатам окончания спринта. Таким образом, будет достигнута возможность планировать совместную разработку и реализацию проекта, а также проводить своевременные корректировки и анализ проделанной работы, где каждый из участников будет выполнять поставленные задачи.

Также для обеспечения работоспособности и возможности реализации работы некоторых из решателей задач, в частности, направленных на диагностику заболеваний по результатам анализов, необходимо актуализировать и расширить ранее разработанную базу знаний в области медицинских анализов.

## **2 ПРОЕКТИРОВАНИЕ СИСТЕМЫ**

### **2.1 Постановка задачи**

1) Для обеспечения работоспособности решателей задач медицинского ассистента, направленных на реализацию диагностики заболеваний, необходимо актуализировать и расширить ранее разработанную базу знаний в области медицинских анализов.

2) Разработать следующие решатели задач:

а) агент-решатель задач, предназначенный для регистрации пользователя в интеллектуальной системе.

б) агент-решатель задач, предназначенный для авторизации пользователя в интеллектуальной системе.

с) агенты-решатели задач, предназначенные для диагностики заболеваний на основе анализов пациента:

– по результатам общего анализа крови.

– по результатам биохимического анализа крови на содержание витаминов.

– по результатам гормонального анализа крови.

д) агент-решатель задач, предназначенный для навигации пользователя в базе знаний и формирования ссылки с текстом, включающим в себя описание запрашиваемого узла в базе знаний.

е) агент-решатель задач, предназначенный для предоставления рекомендаций по здоровью при определенной болезни для пациента.

3) Реализовать веб-приложение, включающее в себя взаимодействие с базой знаний и решателями задач интеллектуальной системы.

а) Интегрировать в веб-приложение базу знаний интеллектуальной системы, включая процесс заполнения медицинской карты пациента.

б) Реализовать процесс регистрации пользователя в системе при помощи соответствующего агента-решателя.

с) Реализовать процесс авторизации пользователя в системе при помощи соответствующего агента-решателя.

д) Реализовать процесс прохождения и отображения результатов диагностики на основе приведенных пользователем данных в системе при помощи соответствующих агентов-решателей.

### **2.2 Потенциальные пользователи**

Медицинский ассистент в области медицины будет предлагать свои услуги широкому кругу пользователей, каждый из которых имеет уникальные потребности и цели. Вот некоторые из потенциальных пользователей и сценарии их использования:

Пациенты:

а) Люди, ищущие разъяснения по своим симптомам, могут обратиться к медицинскому ассистенту в поисках достоверной информации, которая поможет им понять возможные причины их заболеваний, а также определить следующие шаги для дальнейшего лечения.

б) Медицинский помощник предлагает обширную базу данных о методах лечения и уходе за здоровьем, что может стать ценным ресурсом для тех, кто столкнулся с новым диагнозом и ищет способы улучшения своего состояния.

в) Система облегчает взаимодействие пациента с соответствующим медицинским специалистом, предоставляя им удобный доступ к нужной информации и услугам.

Медицинские специалисты:

– Ассистент обеспечивает врачам быстрый доступ к информации о своих пациентах и их медицинских записях, что способствует более эффективному уходу и лечению.

– Система обеспечивает медицинскому персоналу возможность быстрого доступа к данным и анализу информации, что позволяет им более точно оценивать состояние пациентов и принимать обоснованные решения о лечении.

На рисунке 2.1 изображена диаграмма использования медицинского ассистента.

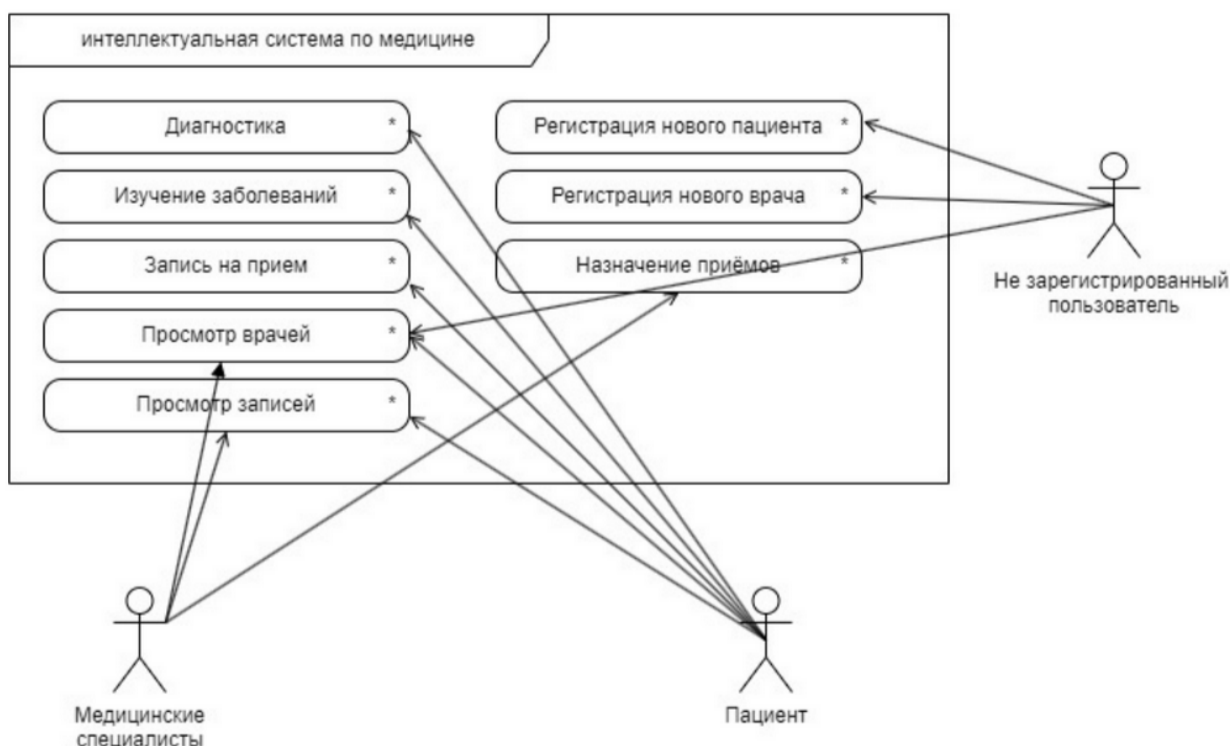


Рисунок 2.1 – Диаграмма использования системы

## **2.3 Архитектура системы**

Интеллектуальная система по медицине должна иметь следующую архитектуру:

- База знаний данной интеллектуальной системы по медицине.
- Решатели задач персонального ассистента по медицине.
- Интерфейс персонального ассистента по медицине, представленный в виде веб-приложения.

### **2.3.1 Архитектура базы знаний**

В базе знаний по медицине используется технология OSTIS, а в качестве средства представлений знаний - SC-код и SCg, с помощью которых были формально описаны знания и представлены знания в форме семантической сети.

Основная функция базы знаний для интеллектуальной системы по медицине заключается в обеспечении доступа к надежной и актуальной информации о причинах заболеваний, методах лечения и профилактике. Это позволяет увеличить точность диагностики болезней по заданным симптомам. Данная база знаний также предполагает возможность использования не только для диагностики заболеваний, а также и для получения актуальной и достоверной информации о конкретных заболеваниях.

Интеллектуальная система позволяет пользователям проводить предварительную диагностику заболеваний по конкретным симптомам, а также получать необходимую информацию о запрашиваемом заболевании и рекомендациям по здоровью при данном заболевании.

База знаний данной интеллектуальной системы включает в себя 6 разделов:

- ПрО медицинских процедур;
- ПрО медицинских карт;
- ПрО медицинской диагностики;
- ПрО заболеваний;
- ПрО методов лечения;
- ПрО организации медицинских учреждений.

Данные разделы также в свою очередь делятся на подразделы. Структура базы знаний изображена на рисунке 2.2.

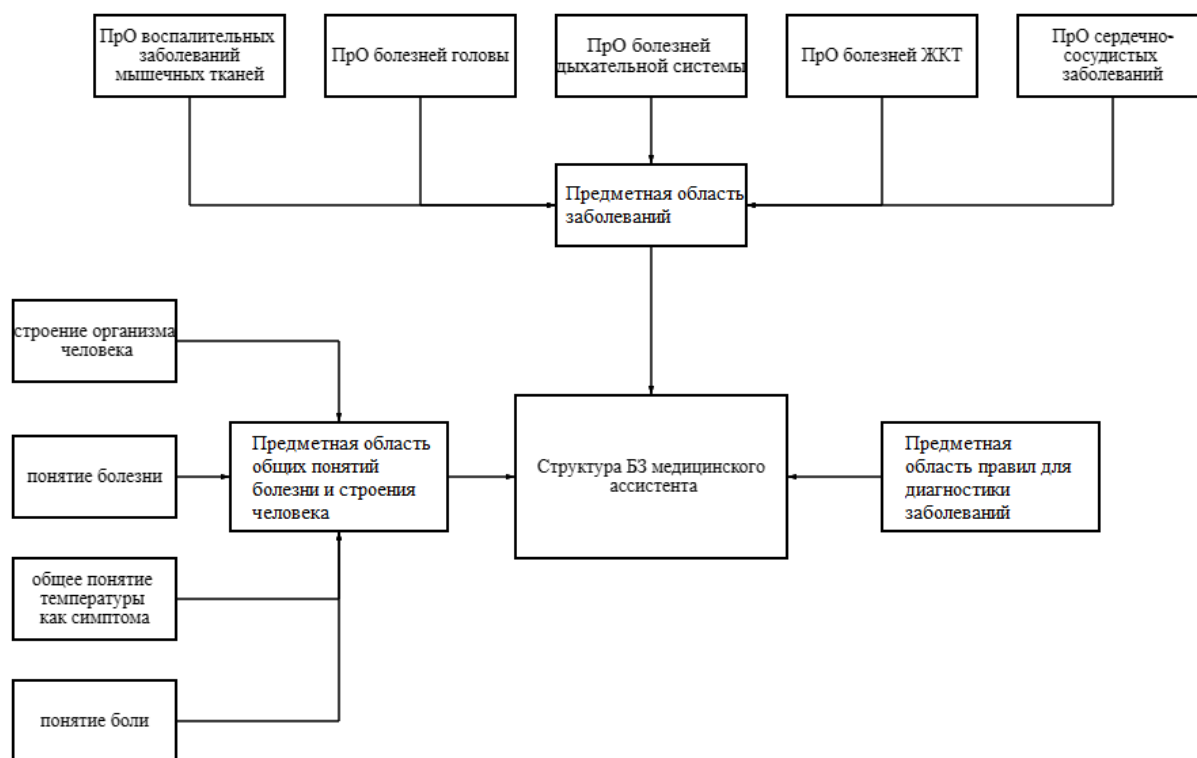


Рисунок 2.2 – Структура базы знаний

Важной задачей проектирования является интеграция актуальной медицинской информации, что позволит пользователям быстро находить необходимые данные о здоровье. Система будет регулярно обновляться, учитывая последние исследования и практики, что повысит осведомленность пользователей и улучшит качество медицинского обслуживания.

### 2.3.2 Архитектура решателей задач

Основная цель интеллектуальной системы в области медицины заключается в создании эффективного решателя задач, который предоставит пользователям точные и своевременные рекомендации на основе введенного запроса. Решатель задач будет служить центральным компонентом системы, обрабатывающим пользовательские запросы и обеспечивающим доступ к структурированной базе знаний о заболеваниях, их причинах, методах лечения и профилактике, а также предоставлять возможность самостоятельной диагностики пользователям. Кроме этого решатели задач будут задействованы при прохождении пользователем процедуры регистрации и авторизации в веб-приложении. Исходя из этого обязательными для интеграции в веб-приложение являются следующие агенты-решатели:

- агент-решатель задач, предназначенный для регистрации пользователя в интеллектуальной системе.

- агент-решатель задач, предназначенный для авторизации пользователя в интеллектуальной системе.

Рекомендуемыми, но не обязательными для интеграции в веб-приложение в рамках текущего курсового проекта в свою очередь будут являться следующие агенты-решатели:

- агенты-решатели задач, предназначенные для диагностики заболеваний на основе анализов крови пациента.

- агент-решатель задач, предназначенный для навигации пользователя в базе знаний и формирования ссылки с текстом, включающим в себя описание запрашиваемого узла в базе знаний

- агент-решатель задач, предназначенный для предоставления рекомендаций по здоровью при определенной болезни для пациента

### **2.3.3 Пример работы одного из агентов-решателей**

**Диагностики заболеваний на основе общего анализа крови.**  
Для выявления заболевания на основе общего анализа крови были выбраны 3 основных показателя, на основе которых будет производиться диагностика заболевания:

- Показатель лейкоцитов.
- Показатель эритроцитов.
- Показатель тромбоцитов.

Для заболеваний, являющихся диагностируемыми по данным показателям, в базе знаний интеллектуальной системы дополнительно были формализованы соответствующие характерные значения анализируемых показателей.

#### **Алгоритм работы агента-решателя:**

1) Агент проверяет, были ли переданы три ссылки, содержащие показатели лейкоцитов, эритроцитов и тромбоцитов. Если хотя бы одного показателя нет, агент завершает свою работу с ошибкой. Если все показатели присутствуют, агент получает их значения для дальнейшей обработки.

2) Агент выполняет проверку, какие тесты соответствуют заданным показателям. Для этого он использует метод извлечения информации о тестах из базы знаний, применяя итераторы для 3 и 5 элементов. Сначала ищутся минимальные значения показателей, затем — максимальные. Каждый найденный показатель сопоставляется с определенным тестом.

3) Агент проверяет входные параметры на соответствие всем тестам, полученным из базы знаний. Если входные параметры соответствуют определенному тесту, он добавляется в список результатов.

4) Если не найдено ни одного соответствия, агент выводит параметр `no_diseases`. В противном случае результатом будет множество соответствующих тестов.

### 2.3.4 Архитектура веб-приложения

Для разработки веб-приложения медицинской системы был выбран микросервисный подход. Веб-приложение медицинской системы, разработанное с использованием микросервисного подхода, представляет собой распределённую архитектуру, состоящую из нескольких независимых компонентов. Такое приложение позволяет гибко управлять различными аспектами медицинских данных, упрощает обновления и масштабирование системы. Основные компоненты системы включают в себя:

- Клиентский браузер;
- веб-сервер;
- сервер базы данных;
- микросервис для взаимодействия с платформой Ostis;
- платформа Ostis.

Веб-приложение медицинской системы, состоящее из перечисленных компонентов, структурируется по микросервисной архитектуре, которая организована в несколько ключевых слоев. Такое разбиение упрощает масштабируемость и управление системой, а также улучшает её отказоустойчивость.

Основные слои архитектуры включают:

- Клиентский слой – это интерфейс, через который пользователи взаимодействуют с системой. Он представлен в виде клиентского приложения (обычно веб-интерфейса), с которым работают через браузер. На этом уровне реализуются элементы интерфейса пользователя, такие как формы, кнопки, панели, таблицы, и обеспечивается интуитивное взаимодействие с данными. Все запросы, выполняемые с клиентского слоя, направляются на серверную часть системы для обработки, а затем возвращаются в виде ответов или обновлённого интерфейса.

- Слой микросервисов (бизнес-логика). На уровне микросервисов располагается основная бизнес-логика приложения. Каждый микросервис является отдельным, независимым компонентом, который может работать автономно, что облегчает разработку и развёртывание. Разделение бизнес-логики на микросервисы позволяет реализовать масштабируемую и устойчивую к сбоям архитектуру: при необходимости можно расширить конкретный сервис без изменения других частей системы.

- Слой данных включает в себя сервер базы данных, который хранит и управляет всей медицинской информацией. На этом уровне организуется доступ к данным о пациентах, их истории приёмов, назначениях, записи о медицинских процедурах и других аспектах, необходимых для функционирования системы.

Для обеспечения эффективного взаимодействия между архитектурными слоями приложения, каждый из них обращается к функциональности других через API — программный интерфейс, который позволяет компо-



нентам взаимодействовать друг с другом, не зная деталей их внутренней реализации. API стандартизирует обмен данными и вызов функциональных возможностей. Такой подход позволяет каждому компоненту развиваться независимо, так как другие части системы обращаются к нему через установленный интерфейс, не затрагивая внутреннюю логику.

Для создания API в микросервисной архитектуре используется подход REST, который обеспечивает удобный и понятный интерфейс для взаимодействия между клиентом и сервером. RESTful API, благодаря своей гибкости и простоте, является оптимальным выбором для распределённых систем. Он позволяет легко масштабировать приложение, а также организовать обмен данными между независимыми сервисами.

Диаграмма взаимодействия компонентов веб-приложения представлена на рисунке 2.3.

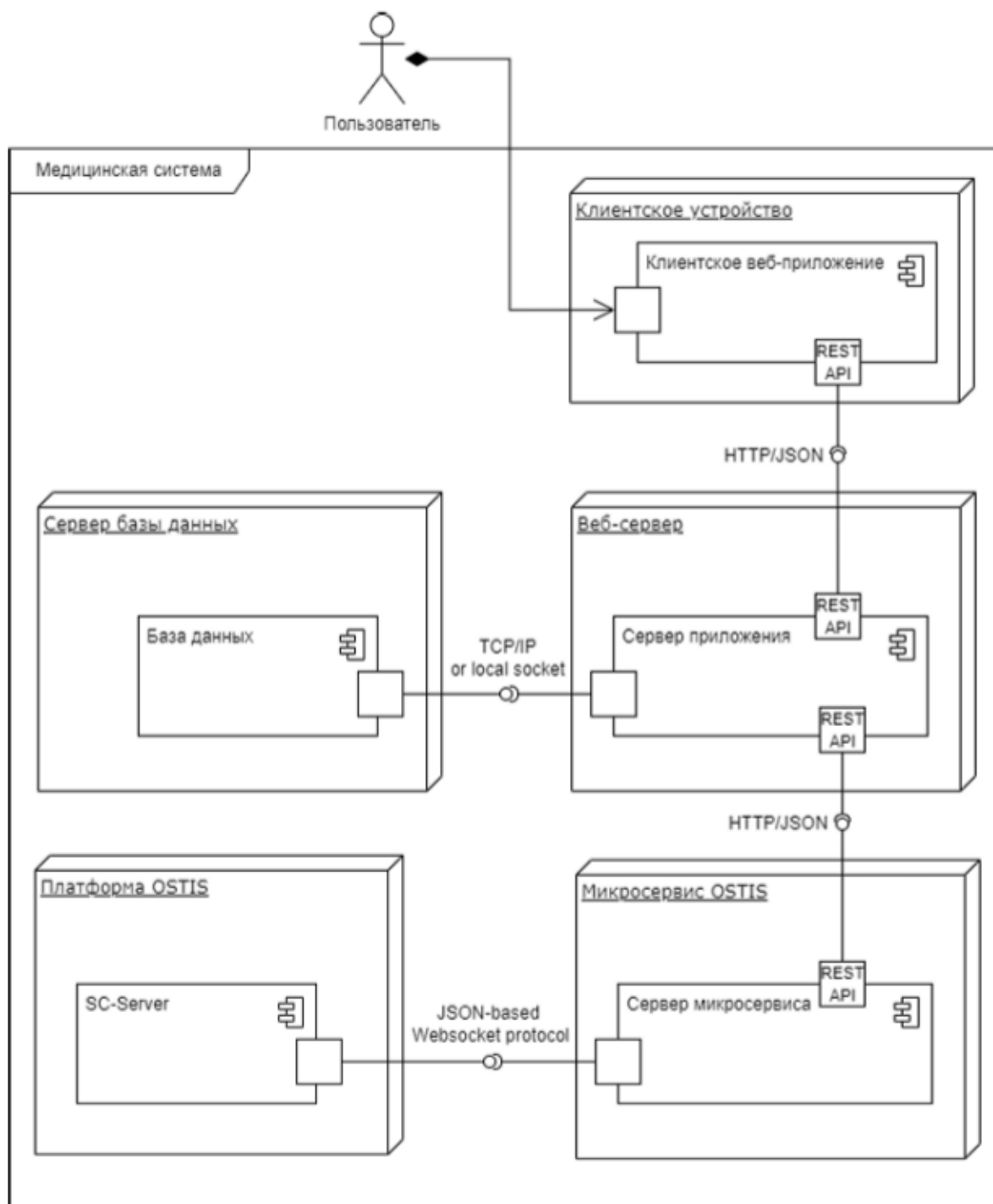


Рисунок 2.3 – Диаграмма взаимодействия компонентов веб-приложения

## 2.4 Вывод

Спроектированная модель медицинского ассистента представляет собой инновационную систему, объединяющую современные технологии и методы медицинского анализа для предоставления персонализированной диагностики. Ассистент будет оснащен функционалом по анализу симптомов, предоставлению диагностических рекомендаций и поддержке в принятии решений как для пациентов, так и для медицинского персонала.

Реализация данной модели имеет потенциал решить множество проблем, с которыми сталкиваются как пациенты, так и медицинские учреждения. Среди основных преимуществ можно выделить:

- Оптимизация работы медицинского персонала. Система может обрабатывать большой объем данных и предоставлять врачам ценную информацию для поддержки в принятии решений, что помогает сократить время диагностики и улучшить качество обслуживания пациентов.

- Обширность и достоверность информации. Интеллектуальная система содержит подробную и актуальную информацию о различных аспектах здоровья и медицины, проверенную медицинскими экспертами и основанную на научных исследованиях и клинических рекомендациях.

- Удобство доступа. Пользователям предоставлен удобный интерфейс для вызова и отслеживания результатов работы вызываемых решателей задач в интеллектуальной системе.

- Поддержка персонализированных консультаций. Интеллектуальная система предоставляет инструменты самостоятельной диагностики для пользователей.

Реализация модели также предполагает обеспечение безопасности и конфиденциальности данных пациентов, а также необходимость непрерывного обновления и адаптации системы в соответствии с изменяющимися потребностями медицинского сообщества.

## 3 РАЗРАБОТКА СИСТЕМЫ

### 3.1 Используемые средства разработки

#### 3.1.1 Метасистема OSTIS

Для реализации базы знаний, на основе которой будет функционировать медицинский ассистент была использована технология OSTIS.

В технологии OSTIS (Open Semantic Technology for Intelligent Systems) используется специальный язык программирования, называемый SC-код (Semantic Code).

SC-код – способ универсального смыслового представления (кодирования) информации в памяти компьютерных систем. Он основан на теории графов (синтаксис) и теории множеств (семантика), что обеспечивает универсальность и унифицированность (единообразие) представления информации, удобство машинной обработки и восприятия человеком.

С помощью SC-кода можно описывать базы знаний, решатели задач и интерфейс интеллектуальной системы.

SC-код – это компьютерный код семантических сетей, являющийся упрощенным представлением семантических сетей – с минимальным алфавитом и бинарными связками.

SC-code является абстрактным языком, но его можно визуализировать в различных формах. Формы внешнего представления SC-кода:

- SCs (текстовый линейный)
- SCn (гипертекстовый)
- SCg (графический)

Для описания понятий использовался подязык SC-кода – SCg.

SCg (Semantic Code graphical) является частью технологии OSTIS и представляет собой графический язык для описания знаний и моделирования визуальных структур.

SCg используется для визуального представления знаний, сущностей, связей и операций в виде графов. Он предоставляет графический способ описания и визуализации знаний, что делает его более доступным и интуитивно понятным для пользователей, не имеющих опыта в программировании.

Особенность SCg:

Узлы (Nodes). Узлы в SCg представляют собой сущности или объекты, которые могут быть описаны с помощью SC кода. Каждый узел имеет определенный тип и характеристики, которые могут быть заданы с использованием атрибутов.

SCg является удобным инструментом для визуализации и моделирования знаний в технологии OSTIS. Он позволяет легко создавать и взаимодействовать с графическими структурами знаний, что делает его полезным

для разработки интеллектуальных систем и решения сложных задач.

Для создания базы знаний в SCg коде использовался инструмент КВЕ.

КВЕ (Knowledge Base Editor) - это инструмент для создания фрагментов базы знаний (БЗ) в SCg (Semantic Code graph). SCg - это формальный язык для описания знаний с использованием семантической графики. КВЕ предоставляет пользователю графический интерфейс для создания и редактирования знаний в виде семантических графов.

Основные возможности КВЕ включают:

- Графический интерфейс. КВЕ предоставляет интуитивно понятный графический интерфейс, который позволяет пользователям визуализировать и редактировать семантические графы. Это облегчает создание и редактирование структуры базы знаний.

- Визуализация. КВЕ предоставляет средства для визуализации семантических графов, что помогает пользователям лучше понять структуру и связи между знаниями в базе данных.

Благодаря КВЕ пользователи могут легко создавать и управлять базами знаний в SCg, что делает процесс разработки и использования семантических моделей более эффективным и удобным.

### **3.1.2 Docker**

Docker — это открытая платформа для разработки, доставки и запуска приложений. Docker предоставляет возможность упаковать и запустить приложение в слабо изолированной среде, называемой контейнером. Преимуществами Docker являются:

- Платформенная независимость— приложение, контейнеризированное при помощи Docker возможно развернуть и запустить на всех наиболее популярных платформах.

- Контейнеры легковесны и содержат в себе все необходимое для запуска приложения: операционную систему, программное обеспечение, библиотеки, конфигурацию.

- Удобство распространения и совместного использования контейнеров.

### **3.1.3 Python**

Python — это интерпретируемый высокоуровневый язык программирования с динамической типизацией. Его основными особенностями являются простота синтаксиса и широкий выбор библиотек для различных областей разработки, что делает его универсальным инструментом как для новичков, так и для профессионалов. Python активно применяется в веб-разработке,



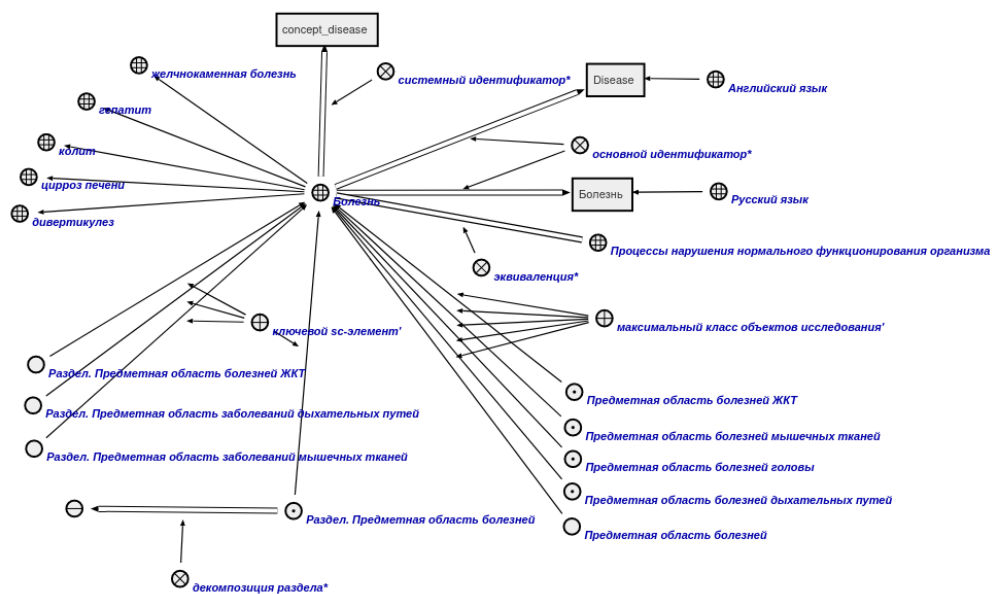


Рисунок 3.2 – Определение болезни в системе

На рис. 3.3 пример определения понятия кашель.

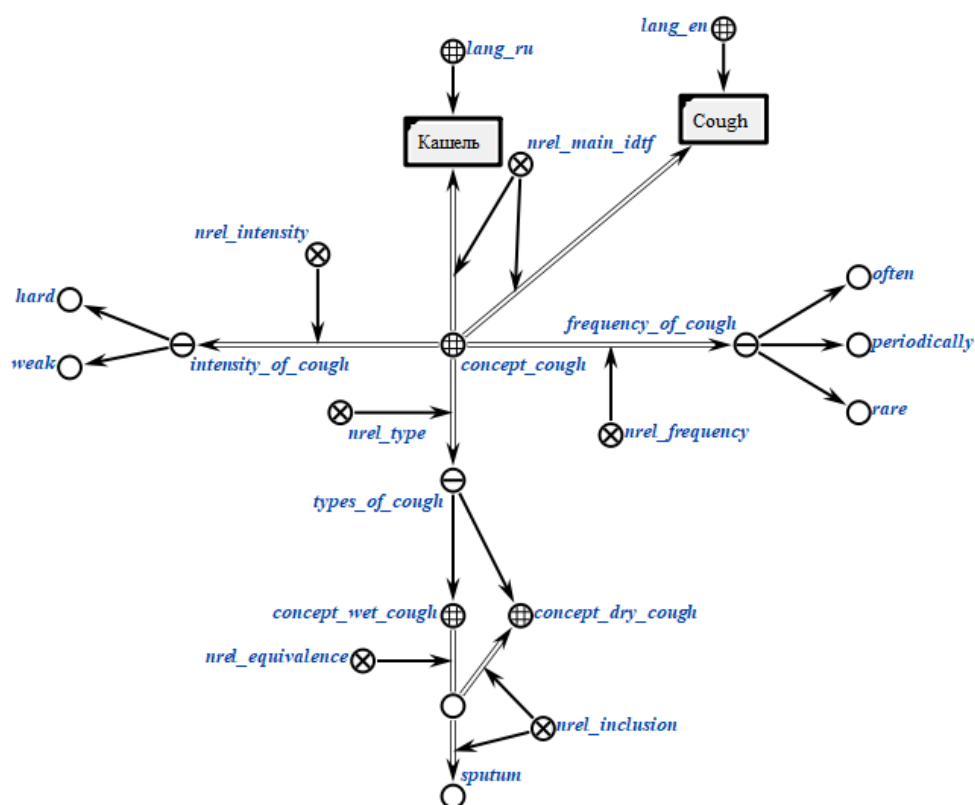


Рисунок 3.3 – Определение кашля

На рис. 3.4 пример погружения в систему понятия кашель.

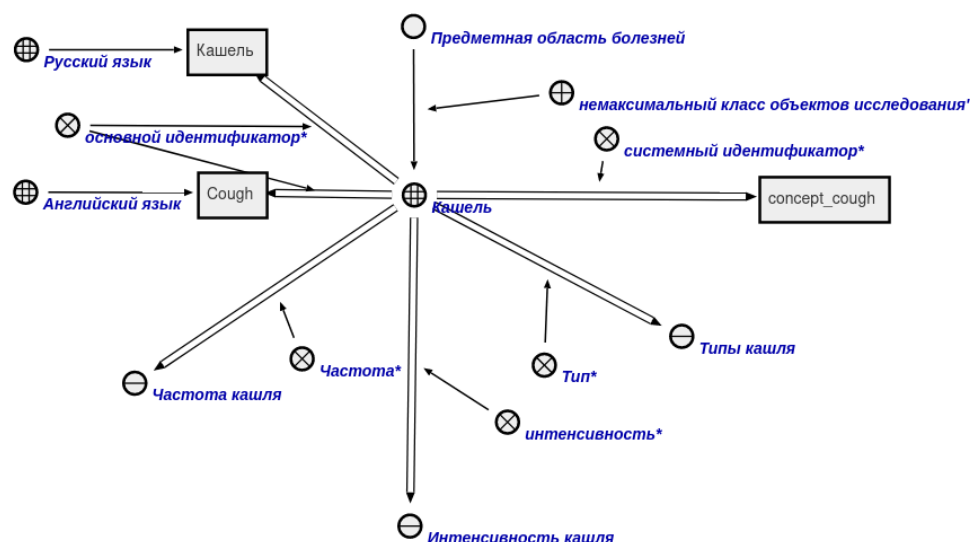


Рисунок 3.4 – Определение кашля в системе

На рис. 3.5 пример отображения предметной области(а) и раздела предметной области(б) в системе.

#### Предметная область болезней

= основной идентификатор\*:

- Предметная область болезней

∈ Русский язык

- Subject domain of diseases

∈ Английский язык

= системный идентификатор\*:

subject\_domain\_of\_diseases

= частная предметная область\*:

- section\_subject\_domain\_of\_diseases\_of\_the\_gastrointestinal\_tract
- section\_subject\_domain\_of\_head\_diseases
- section\_subject\_domain\_of\_inflammatory\_diseases\_of\_muscle\_tissues
- section\_subject\_domain\_of\_respiratory\_diseases

= частная предметная область\*:

Раздел. Предметная область болезней

∈ ключевой sc-элемент\*:

Раздел. Предметная область болезней

⇒ немаксимальный класс объектов исследования\*:

- concept\_hepatitis
- concept\_diverticulosis
- concept\_cholelithiasis
- concept\_cirrhosis\_of\_the\_liver
- concept\_colitis

⇒ максимальный класс объектов исследования\*:

Болезнь

∈ структура

∈ предметная область

(а) Предметная область болезней

#### Раздел. Предметная область болезней

= основной идентификатор\*:

- Раздел. Предметная область болезней

∈ Русский язык

- Section. Subject domain of diseases

∈ Английский язык

= системный идентификатор\*:

section\_subject\_domain\_of\_diseases

= идентификатор\*:

- Раздел базы знаний IMS, описывающий симптомы, болезни и причины их возникновения

∈ Русский язык

- A section of the IMS knowledge base describing the symptoms, diseases and causes of their occurrence

∈ Английский язык

= декомпозиция раздела\*:

- section\_subject\_domain\_of\_respiratory\_diseases
- section\_subject\_domain\_of\_inflammatory\_diseases\_of\_muscle\_tissues
- section\_subject\_domain\_of\_head\_diseases
- section\_subject\_domain\_of\_diseases\_of\_the\_gastrointestinal\_tract

= частная предметная область\*:

Предметная область болезней

⇒ ключевой sc-элемент\*:

- Предметная область болезней
- Болезнь

∈ атомарный раздел

∈ sc-модель базы знаний

(б) Раздел. Предметная область болезней

Рисунок 3.5 – Пример погруженной в систему предметной области



### 3.3 Демонстрация и тестирование агента анализа крови

На рисунке 3.6 представлен пример запуска решателя задач:

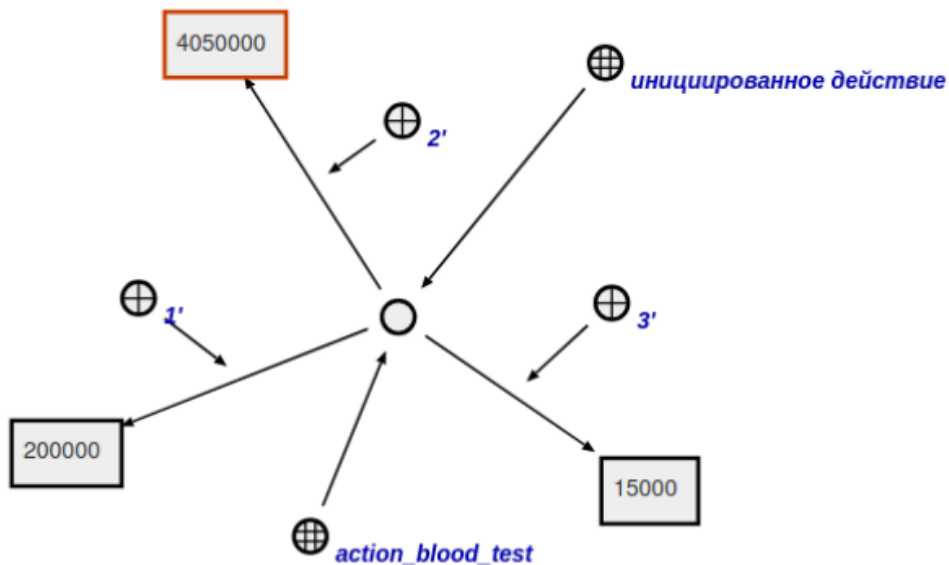


Рисунок 3.6 – Пример запуска решателя задач

На рисунке 3.7 представлен успешный результат работы агента:

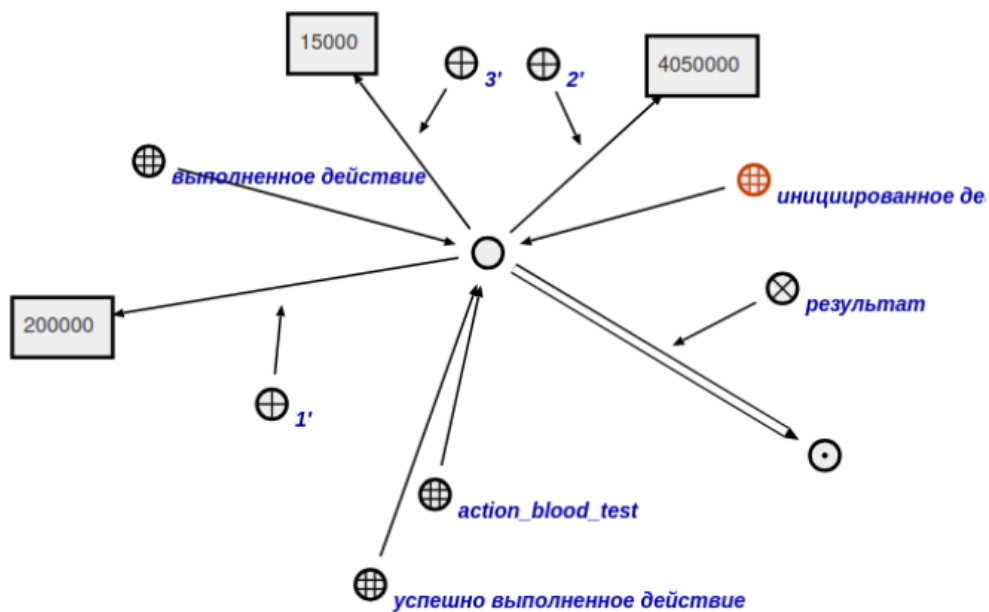


Рисунок 3.7 – Успешный результат работы агента

На рисунке 3.8 представлен пример результата работы агента:

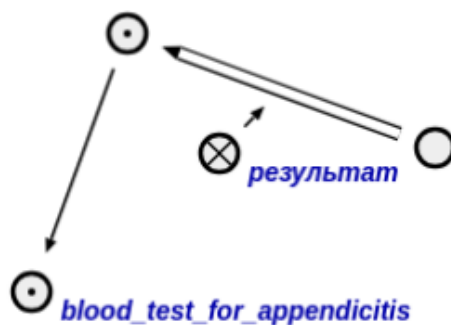


Рисунок 3.8 – Пример результат агента

### 3.4 Демонстрация и тестирование веб-приложения

На рисунках 3.9 и 3.10 представлены соответственно окно регистрации и окно авторизации пользователя в системе. После заполнения одного из приведённых окон и отправки пользователем данных соответствующий агент-решатель реализует процесс регистрации или авторизации пользователя в системе, предварительно проверив корректность введенных данных, а также отобразит данные в системе о результате своей работы.

## Регистрация

Имя

Фамилия

дд.мм.гггг

Выберите регион

Номер телефона

Адрес электронной почты

Пароль

Назад

Зарегистрироваться

Рисунок 3.9 – Окно регистрации пользователя в системе

Адрес электронной почты

Пароль

Войти

или

Впервые у нас? [Зарегистрироваться](#)

Рисунок 3.10 – Окно авторизации пользователя в системе

### 3.5 Вывод

В данном разделе были выделены и описаны основные инструменты, используемые в ходе разработки интеллектуальной медицинской системы. Среди них были выделены следующие ключевые компоненты:

- метасистема OSTIS — универсальная основа для проектирования интеллектуальных систем, обеспечивающая высокую гибкость, адаптивность и совместимость.

- Docker — платформа для контейнеризации, позволяющая развертывать приложения на различных операционных системах, обеспечивая простоту распространения и использования.

- Python — высокоуровневый язык программирования с широким набором библиотек, подходящий для реализации интеллектуальных систем и работы с данными.

Данные инструменты обеспечивают надежную основу для реализации интеллектуальной медицинской системы, обеспечивая необходимые функциональные возможности и технические характеристики для дальнейшего тестирования и эксплуатации разрабатываемой системы.

## ЗАКЛЮЧЕНИЕ

В ходе работы над данным курсовым проектом было произведено существенное расширение и актуализация медицинской научной информации, находящейся в базе знаний интеллектуальной системы. Также были реализованы решатели задач, являющиеся основным компонентом интеллектуальной системы по медицине, а также позволяющие пользователю активно взаимодействовать с базой знаний интеллектуальной системы и обеспечивающие некоторые принципы работы веб-приложения, реализованного для обеспечения удобного и функционального доступа пользователей к взаимодействию с разрабатываемой интеллектуальной системой. Был получен опыт в сфере командной работы над проектом, включающий опыт использования выбранного оптимальным подхода к управлению - 'Метод критического пути', опыт совместного планирования, разработки и реализации проектов в качестве руководителя проекта.

Таким образом в результате проектирования интеллектуальной медицинской системы была реализована архитектура системы, учитывающая выявленные в ходе анализа требования к функциональности, безопасности, а также масштабируемости системы. В связи с этим данная система позволяет обеспечить удобство и надежность использования как для медицинских специалистов, так и для пациентов.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Woebot. — Режим доступа: <https://woebothealth.com/>. — Дата доступа: 27.03.2024.

[2] HealthTap. — Режим доступа: <https://www.healthtap.com/>. — Дата доступа: 27.03.2024.

[3] Ada Health. — Режим доступа: <https://ada.com/>. — Дата доступа: 27.03.2024.

[4] Semantic Web. — Режим доступа: <https://www.ontotext.com/knowledgehub/fundamentals/what-is-the-semantic-web/>. — Дата доступа: 27.03.2024.

[5] Голенков, В. В. Открытая технология онтологического проектирования, производства и эксплуатации семантических совместимых гибридных интеллектуальных компьютерных систем / В. В. Голенков, Н. А. Гулякина, Д. В. Шункевич. — Минск : БГУИР, 2021.

[6] H.Ralston, Stuart. Davidson's Principles and Practice of Medicine / Stuart H.Ralston, Ian D.Penman, Mark W.J. Strachan. — McGraw Hill, 2022. — 4384 P.

[7] Шункевич, Д.В. Агентно-ориентированные модели, методика и средства разработки совместимых решателей задач интеллектуальных систем / Д.В. Шункевич, В.В. Голенков. — Программные продукты и системы, 2020. — Режим доступа: [https://libeldoc.bsuir.by/bitstream/123456789/42228/1/Golenkov\\_Agentno\\_orientirovannyye.pdf](https://libeldoc.bsuir.by/bitstream/123456789/42228/1/Golenkov_Agentno_orientirovannyye.pdf). — Дата доступа: 04.10.2024.

[8] Гулякина, Н.А. Семантические модели и метод согласованной разработки баз знаний / Н.А. Гулякина, И.Т. Давыденко. — Программные продукты и системы., 2020. — Режим доступа: <https://cyberleninka.ru/article/n/semanticheskie-modeli-i-metod-soglasovannoy-razrabotki-baz-znaniy>. — Дата доступа: 21.09.2024.

[9] Profilakticheskaya meditsina. — Режим доступа: <https://www.mediasphera.ru/issues/profilakticheskaya-meditsina/2022/2/downloads/ru/1230549482022021081>. — Дата доступа: 27.03.2024.

[10] ОБЩЕЕ УСТРОЙСТВО ЧЕЛОВЕКА И ЕГО СТАНОВЛЕНИЕ. — Режим доступа: <https://cyberleninka.ru/article/n/obschee-ustroystvo-cheloveka-i-ego-stanovlenie>. — Дата доступа: 27.03.2024.