

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Кафедра информационных систем управления

ЧЕРНЫШЁВА Ксения Юрьевна

ТЕХНОЛОГИЯ ПОСТРОЕНИЯ АДАПТИВНЫХ СИСТЕМ
УПРАВЛЕНИЯ КОРПОРАТИВНОЙ МЕДИЦИНОЙ

Дипломная работа

Научный руководитель:

кандидат технических наук,
доцент А.Н. Вальвачев

Допущена к защите

«___» _____ 2025 г

Зав. кафедрой информационных систем
управления

доктор технических наук, доцент,

А.М. Недзьведь

Минск, 2025

Белорусский государственный университет
Факультет прикладной математики и информатики

Кафедра _____ информационных систем управления

(наименование кафедры)

Утверждаю

Заведующий кафедрой _____ Недзьведь А.М.

(подпись)

(фамилия, инициалы)

Дата “ 10 ” февраля 2025 г.

ЗАДАНИЕ НА ДИПЛОМНУЮ РАБОТУ

Обучающемуся (студенту, курсанту) _____ Чернышёва К.Ю.

(фамилия, инициалы)

1. Тема дипломной работы Технология построения адаптивных систем управления корпоративной медициной (наименование темы)

Утверждена приказом ректора БГУ от “ 12 ” ноября 2024 г. №1295-ПС

2. Исходные данные к дипломной работе

1) Афоничкин, А.И. Модель оценки экономической устойчивости предприятий /А.И

Афоничкин, Л.И. Журова // Фундаментальные исследования. – 2015. – No 10-1. – С. 131-136;

2) Календжян, С. Организация эффективной системы управления корпоративной медициной в российских промышленных компаниях / С.Календжян, А.Сальников, О. Гумилевская //Экономическая политика – 2016. – Т.11. – No5. – С. 118-139

3) Корсакова, С.В. Коммерциализация медицины и ее влияние на этику врача С.В.Корсакова,В.Е.Корсакова // Научное обозрение. Медицинские науки. – 2022. – No 6. – С. 18-22;

4) The Corporate Practice of Medicine – Structuring Your Business [Электронный ресурс]. – Режим доступа: <https://www.lambmcerlane.com>. Дата доступа: 02.02.2025.

3. Перечень подлежащих разработке вопросов или краткое содержание расчетно-пояснительной записки:

1) Анализ КМ Норильск-никель и рекомендованной литературы;

2) Постановка задачи, создание моделей и алгоритмов;

3) Разработка программного обеспечения;

4. Перечень графического материала (с точным указанием обязательных чертежей и графиков)

5. Консультанты по дипломной работе с указанием относящихся к ним разделов

6. Примерный календарный график выполнения дипломной работы

10.02.2025 – 23.02.2025: Изучение существующих систем КМ и основных теоретических вопросов; получение задания; анализ литературы.

24.02.2025 – 09.03.2025: Постановка задачи; разработка моделей компании, структуры базы данных как основы КМ.

10.03.2025 – 23.03.2025: Разработка алгоритмов построения моделей.

24.03.2025 – 13.04.2025: Разработка алгоритмов построения модели и жизненного цикла КМ.

14.04.2025 – 27.04.2025: Разработка программного обеспечения с визуализацией данных.

28.04.2025–05.05.2025: Тестирование компонентов, отладка и развёртывание приложения.

06.05.2025 – 19.05.2025: Оформление дипломной работы.

7. Дата выдачи задания _____ 10.02.2025 _____

8. Сроки сдачи законченной дипломной работы _____ 20.05.2025 _____

Руководитель _____ / Вальвачев А.Н. _____ /

(подпись)

(Инициалы, фамилия)

Подпись обучающегося _____

(подпись студента).

Дата _____ 10.02.2025 _____

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	7
ГЛАВА 1. АНАЛИЗ ЛИТЕРАТУРЫ И ПОСТАНОВКА ЗАДАЧИ	12
1.1 Основные проблемы корпоративной медицины	12
1.2 Анализ литературы.....	14
1.3 Постановка задачи	22
ГЛАВА 2. МОДЕЛИ И АЛГОРИТМЫ	26
2.1 Модели.....	26
2.1.1 Модель компании.....	26
2.1.2 Унифицированная модель акторов.....	27
2.1.3 Модель врача	28
2.2 Алгоритмы последовательности действий на разных уровнях	29
2.3 Выводы	31
ГЛАВА 3. РАЗРАБОТКА ЦЕЛЕВОЙ СИСТЕМЫ.	32
3.1 Архитектура системы.....	32
3.2 Разработка программного обеспечения.....	35
3.3 Выводы	54
ГЛАВА 4. ПРИМЕНЕНИЕ СИСТЕМЫ ДЛЯ РЕШЕНИЯ ПРИКЛАДНОЙ ЗАДАЧИ	55
4.1 Постановка задачи	55
4.2 Решение задачи на основе методики	56
4.3 Выводы	59
ЗАКЛЮЧЕНИЕ.....	60
ЛИТЕРАТУРА	61
ПРИЛОЖЕНИЯ	62
Приложение А.....	62
Приложение Б.....	68
Приложение В.....	72

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ

API	интерфейс программного взаимодействия (Application Programming Interface),
AI	искусственный интеллект (Artificial Intelligence), технология, обеспечивающая автоматизированный анализ на основе обученных моделей,
CI/CD	непрерывная интеграция и доставка (Continuous Integration / Continuous Delivery), практика автоматизации сборки, тестирования и развертывания программного обеспечения,
IoT	интернет вещей (Internet of Things), концепция объединения физических устройств, сенсоров и программного обеспечения в сеть с возможностью обмена данными и управления в реальном времени,
СКМ	система корпоративной медицины, цифровая платформа, обеспечивающая мониторинг здоровья сотрудников, управление медицинскими данными и координацию взаимодействия между пациентами, врачами и организациями.
БД	база данных, организованное хранилище информации, обеспечивающее структурированное хранение, быстрый поиск, обновление и удаление данных, используемых в программных системах,
UI	пользовательский интерфейс (User Interface).

РЕФЕРАТ

Дипломная работа, 73 с., 29 рис., 3 приложения.

Ключевые слова: МЕДИЦИНСКАЯ ИНФОРМАТИКА, ЗДРАВООХРАНЕНИЕ, УМНАЯ КОМПАНИЯ, МОДЕЛИРОВАНИЕ, СИСТЕМЫ МОНИТОРИНГА.

Актуальность выбранной темы обусловлена растущей потребностью в цифровизации корпоративного здравоохранения и отсутствием унифицированных мобильных решений для управления медицинскими данными, взаимодействия между пациентами и врачами, а также мониторинга состояния сотрудников в организациях.

Объект исследования: процесс цифровизации системы мониторинга здоровья различных групп работников в крупных и средних компаниях.

Предмет исследования: технические и архитектурные подходы к построению мобильных приложений в сфере корпоративного здравоохранения.

Цели работы: изучить и описать методы и алгоритмы цифровизации систем мониторинга здоровья.

Методы исследования: анализ и сравнение существующих методик мониторинга здоровья, разработка моделей систем мониторинга.

Полученные результаты: разработка системы мониторинга здоровья для крупных и средних корпораций.

Степень внедрения и рекомендации по внедрению: разработанная система может быть внедрена в рамках корпоративных медицинских программ как средство автоматизации процессов записи на приём, хранения и анализа медицинской информации, а также мониторинга состояния сотрудников. Рекомендуется для использования в крупных и средних компаниях с развитой внутренней инфраструктурой.

Достоверность материалов и результатов дипломной работы: все этапы проектирования, разработки и тестирования выполнены лично автором. Полученные результаты и выводы достоверны.

Область применения: интеграция результатов исследования в практические системы, включая здравоохранение и управление корпорациями.

РЭФЕРАТ

Дыпломная работа, 73 с., 29 мал., 3 дадатка.

Ключавыя словы: МЕДЫЦЫНСКАЯ ІНФАРМАТЫКА, АХОВА ЗДАРОЎЯ, РАЗУМНАЯ КАМПАНІЯ, МАДЭЛЯВАННЕ, СІСТЭМЫ МАНІТОРЫНГУ.

Актуальнасць абранай тэмы абумоўлена ростам патрэбнасці ў цыфравізацыі карпаратыўных сістэм аховы здароўя і адсутнасцю уніфікаваных мабільных рашэнняў для кіравання медыцынскімі дадзенымі, узаемадзеяння паміж пацыентамі і ўрачамі, а таксама маніторынгу стану супрацоўнікаў у арганізацыях.

Аб'ект даследавання: працэс цыфравізацыі сістэмы маніторынгу здароўя розных груп работнікаў у буйных і сярэдніх кампаніях.

Прадмет даследавання: тэхнічныя і архітэктурныя падыходы да пабудовы мабільных дадаткаў у сферы карпаратыўнай аховы здароўя.

Мэта працы: вывучыць і апісаць метады і алгарытмы цыфравізацыі сістэм маніторынгу здароўя.

Метады даследавання: аналіз і параўнанне існуючых метадых маніторынгу здароўя, распрацоўка мадэляў сістэм маніторынгу.

Атрыманыя вынікі: распрацавана сістэма маніторынгу здароўя для буйных і сярэдніх карпарацый.

Ступень укаранення і рэкамендацыі па ўкараненні: распрацаваная сістэма можа быць скарыстана ў рамках карпаратыўных медыцынскіх праграм як сродак аўтаматызацыі працэсаў запісу на прыём, захоўванні і аналізу медыцынскай інфармацыі, а таксама маніторынгу стану супрацоўнікаў. Рэкамендуецца для выкарыстання ў буйных і сярэдніх кампаніях з развітой унутранай інфраструктурай.

Дакладнасць матэрыялаў і вынікаў дыпломнай працы: усе этапы праектавання, распрацоўкі і тэсціравання выкананы асабіста аўтарам. Атрыманыя вынікі і высновы дакладныя.

Вобласць ужывання: інтэграцыя вынікаў даследавання ў практычныя сістэмы, уключаючы ахову здароўя і кіраванне карпарацыямі.

ABSTRACT

Graduate Work, 73 pages, 29 figures, 3 appendices.

Keywords: MEDICAL INFORMATICS, HEALTHCARE, SMART COMPANY, MODELING, MONITORING SYSTEMS.

The relevance of the chosen topic is due to the growing need for digitalization of corporate healthcare and the lack of unified mobile solutions for managing medical data, interaction between patients and doctors, and monitoring the condition of employees in organizations.

Object of the research: the process of digitalization of the health monitoring system for various groups of employees in large and medium-sized companies.

Subject of the research: technical and architectural approaches to building mobile applications in the field of corporate healthcare.

Methods of the research: to study and describe the methods and algorithms for digitalization of health monitoring systems.

The purpose of the research: analysis and comparison of existing health monitoring methods, development of monitoring system models.

Results: development of a health monitoring system for large and medium-sized corporations.

Degree of implementation and recommendations for implementation: the developed system can be implemented within the framework of corporate medical programs as a means of automating the processes of making appointments, storing and analyzing medical information, as well as monitoring the condition of employees. Recommended for use in large and medium-sized companies with a developed internal infrastructure.

Reliability of materials and results of the thesis: all stages of design, development and testing were performed personally by the author. The results and conclusions obtained are reliable.

Scope: integration of research results into practical systems, including healthcare and corporate management.

ВВЕДЕНИЕ

В результате фундаментальных технологических, экономических, политических и социальных перемен в XXI в. появились и быстро развиваются новые типы организационно-технических систем (ОТС) [1,2]. Они отличаются от традиционных ОТС глубокой интеграцией информационных технологий во все области жизни населения, включая: производство, транспорт, энергоснабжение, здравоохранение, управление, образование, досуг и т.д. К важнейшей группе новых ОТС относятся так называемые умные корпорации (Smart Corporations) [3].

Построение умной корпорации – сложная, многогранная и масштабная проблема, которую решают ведущие страны мира [4]. Важнейшее преимущество умной корпорации заключается в контроле важнейших составляющих жизненного цикла компании с помощью компьютерных технологий и принятии эффективных оперативных решений в случае возникновения проблемных ситуаций.

Одна из первоочередных задач при построении умной корпорации заключается в цифровой трансформации процессов обращения за медицинской помощью сотрудников корпорации, т.е. построении компьютеризованных коммуникаций и интеллектуальных систем для контроля здоровья и повышения качества жизни различных групп работников компаний [5, 6, 7]. Актуальность этой задачи вызвана рядом объективных факторов: ростом средней продолжительности жизни человека, нехваткой квалифицированного медицинского персонала и высокой стоимостью медицинского наблюдения и лечения. В современном мире нехватка времени стала насущной проблемой для всех, человек не может регулярно заботиться о своем здоровье. Выявление ранних признаков заболевания возможно с помощью регулярных медицинских осмотров. Построение системы мониторинга здоровья может помочь нам регулярно проверять свое здоровье непосредственно на рабочем месте. Она позволяет получать информацию о различных параметрах здоровья, таких как пульс, температура, артериальное давление и энергия. Целью этого медицинского мониторинга является контроль за состоянием здоровья сотрудника, профилактика возможных заболеваний и, как следствие, минимизация количества дней отпуска по болезни.

Традиционные методы медицинского обслуживания во многом не соответствуют ни масштабу, ни сложности новых проблем здравоохранения. Кроме того, администрация компании должна оперативно оценивать эффективность работы клиник и постоянно оказывать им помощь для улучшения здоровья работников. Таким образом, актуальными являются три задачи:

- автономная оценка состояния пациента, принятие соответствующего решения и отправка соответствующего сообщения в клинику;
- обеспечение приема сотрудника в клинике (назначение медицинского персонала для оказания медицинской помощи), накопление статистики;
- анализ статистики о деятельности клиник администрацией компании,
- оказание им помощи при необходимости (например, во время эпидемий).

Комплексное решение этих задач обеспечит синхронизацию отношений пациентов, медицинского персонала клиник и администрации компании [6, 7].

Один из наиболее эффективных способов решения заключается в разработке специализированных интеллектуальных систем, обеспечивающих фиксацию проблемных ситуаций у пациента, их оценку, вызов соответствующего медицинского персонала, ежедневный анализ работы клиник и оценку их деятельности административными органами (рисунок 1.1).

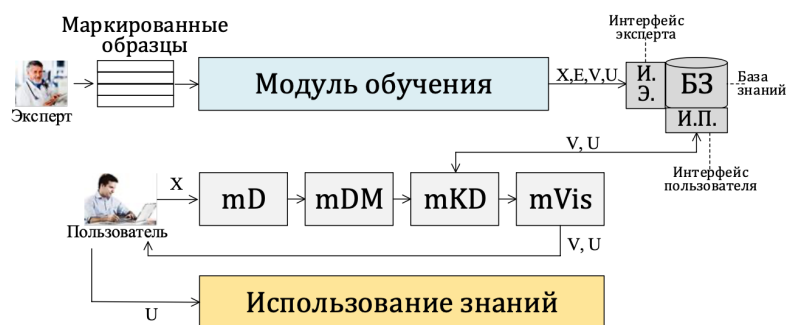


Рисунок 1.1 – Макро-архитектура интеллектуальных систем (ИС)

В данной работе рассматривается вариант построения системы мониторинга здоровья работников корпорации, особенность которой заключается в том, что она может разворачиваться поверх существующих клинических систем по принципу HLA (High Level Architecture) [8] и не требует изменения принципов их работы.

Рассматриваемая система мониторинга здоровья работников корпораций представляет собой инновационное и высокотехнологичное решение, нацеленное в основном на крупные и средние предприятия. В малых компаниях, где часто действуют более простые структуры управления и бюджетные ограничения, внедрение подобных систем может оказаться финансово и организационно неосуществимым в настоящее время.

В контексте малых предприятий, где ресурсы и организационные возможности могут быть ограничены, внедрение высокотехнологичных систем мониторинга здоровья может столкнуться с рядом препятствий:

- **финансовые ограничения.** Малые компании часто имеют ограниченные бюджеты на внедрение новых технологий. Реализация дорогостоящих систем мониторинга здоровья может выходить за рамки их финансовых возможностей;

- **организационные сложности.** Внедрение сложных технологических систем может требовать значительных изменений в организационной структуре и бизнес-процессах компании. Малые предприятия могут испытывать трудности в адаптации к таким изменениям;

- **необходимость специализированных ресурсов.** Внедрение и поддержка систем мониторинга здоровья требует наличия высококвалифицированных специалистов, которые могут быть недоступны для малых компаний.

С развитием технологий и ростом интереса к интеллектуальным решениям можно ожидать, что в будущем подобные системы станут более доступными и для малого бизнеса. Тенденции в области "умных" корпоративных структур и систем мониторинга здоровья сотрудников создают предпосылки для появления решений, адаптированных под потребности компаний различного масштаба.

На данный момент основное внимание при разработке и внедрении таких систем сосредоточено на средних и крупных предприятиях. Это объясняется тем, что именно в таких организациях существует необходимая база — как организационная, так и финансовая — для эффективного внедрения и сопровождения подобных инициатив.

Концепция «умной корпорации» охватывает широкий спектр направлений: от производственных процессов и управления до сферы образования и охраны здоровья персонала. Одним из ключевых компонентов этой модели выступает корпоративная медицина — специализированная система, направленная на сохранение здоровья сотрудников и поддержание стабильности рабочих процессов.

Такая система, как правило, представляет собой интеграцию информационных технологий, современных диагностических средств и квалифицированного медицинского персонала. Это позволяет не только оперативно выявлять потенциальные проблемы со здоровьем, но и своевременно принимать меры, ориентируясь на индивидуальные особенности каждого работника.

Реализация подобных подходов уже активно осуществляется в ряде крупных компаний. Так, например, технологические гиганты вроде Google и Apple создают на территории своих кампусов полноценные медицинские центры. Эти центры оснащены современным оборудованием, включающим электронные медицинские карты, системы телемедицины и автоматизированные средства контроля состояния здоровья. Благодаря этому снижается время ожидания помощи и минимизируются перерывы в работе.

Одной из основных причин активного развития систем корпоративной медицины является длительное ожидание приема у врачей-диагностов и узких специалистов в традиционной медицинской системе. Данные, представленные на рисунке 1.2, иллюстрируют продолжительность ожидания медицинской помощи в различных странах по состоянию на 2023 год. Например, в США среднее время ожидания консультации у врача-специалиста составляет 21 день, в Великобритании и Италии – 10 дней, во Франции – 6 дней, а в Швейцарии – всего 2 дня. Таким образом, очевидна необходимость оптимизации доступа к медицинским услугам, что возможно посредством внедрения адаптивных корпоративных систем.

Страна	Количество дней ожидания (2023 г.)
США	21
Великобритания, Италия	10
Франция	6
Швейцария	2

Рисунок 1.2 – Среднее время ожидания врачей в 2023 г.

Длительное ожидание медицинской помощи не только снижает общую продуктивность персонала из-за потери рабочего времени, но и повышает вероятность перехода острых заболеваний в хронические формы. Это, в свою очередь, ведёт к увеличению затрат на лечение и к более длительным периодам нетрудоспособности. В рамках корпоративной медицины подобные риски значительно снижаются за счёт систематического наблюдения за состоянием здоровья сотрудников и своевременного реагирования на возникающие проблемы.

Кроме того, корпоративные медицинские системы особенно важны в условиях чрезвычайных ситуаций, таких как эпидемии или пандемии. Опыт глобального кризиса, вызванного COVID-19, показал, что наличие налаженной внутренней медицинской инфраструктуры позволяет компаниям более эффективно проводить профилактические мероприятия, быстро выявлять

случаи заболевания и снижать риск массового распространения инфекции. Это способствует сохранению стабильной работы организации даже в условиях внешних потрясений.

Таким образом, развитие адаптивных систем управления корпоративной медициной является важной и актуальной задачей, позволяющей решать целый комплекс проблем: от оперативного выявления рисков для здоровья сотрудников до стратегического управления медицинскими ресурсами и повышения общей эффективности организации.

ГЛАВА 1.

АНАЛИЗ ЛИТЕРАТУРЫ И ПОСТАНОВКА ЗАДАЧИ

1.1 Основные проблемы корпоративной медицины

В сфере корпоративной медицины организации сталкиваются с разнообразными вызовами и проблемами, которые требуют системного подхода и глубокого анализа для их эффективного решения. Один из ключевых аспектов этой проблематики – обеспечение оптимального здоровья и благополучия сотрудников, что непосредственно связано с повышением производительности и эффективности труда. Этот аспект охватывает не только доступность качественного медицинского обслуживания для работников, но и внедрение проактивных программ по предупреждению профессиональных заболеваний и травм.

На состояние здоровья сотрудников влияет широкий спектр факторов, что делает задачу его сохранения особенно сложной. Речь идёт не только о физической рабочей среде, но и о психологических аспектах — таких как уровень стресса, эмоциональное выгорание и общее удовлетворение работой. Немаловажную роль играют и индивидуальные привычки: рацион питания, уровень физической активности, умение справляться с нагрузками.

Неблагоприятные условия труда — например, неэргономичное оборудование, недостаточное освещение, повышенный уровень шума — могут стать причиной хронических заболеваний или травм. В то же время психологические перегрузки способствуют развитию расстройств сна, сердечно-сосудистых заболеваний и депрессии. Хронический стресс признан одним из наиболее значимых факторов, отрицательно влияющих как на психическое, так и на физическое состояние.

В этой связи всё более актуальным становится использование современных технологий для мониторинга здоровья сотрудников. Одним из перспективных направлений является применение решений на базе Интернета вещей (IoT). Носимые устройства позволяют отслеживать в реальном времени такие параметры, как уровень физической активности, стресс, качество сна и другие показатели. Полученные данные могут использоваться для разработки персонализированных программ поддержки здоровья и раннего выявления отклонений.

Помимо IoT, всё большую роль в корпоративной медицине играют технологии анализа больших данных и искусственный интеллект. Они позволяют обрабатывать значительные массивы информации о состоянии

сотрудников и выявлять скрытые закономерности, указывающие на потенциальные риски. Это открывает возможности для внедрения точечных профилактических мер, а также для улучшения условий труда на основании объективных данных.

Однако внедрение таких технологий требует как технической экспертизы для их разработки и внедрения, так и глубокого понимания медицинских аспектов для эффективного использования и достижения поставленных целей. Важно обеспечить, чтобы медицинские данные были защищены и использовались в соответствии с законодательными нормами о защите персональной информации.

Несмотря на существенные преимущества корпоративной медицины и её растущую популярность, современные системы корпоративной медицины (СКМ) обладают рядом недостатков, которые ограничивают эффективность их применения и усложняют внедрение таких решений в корпоративную среду.

Одной из распространённых проблем является узкая специализация и сфокусированность многих существующих СКМ на решении единственной задачи. Например, в Индии, где корпоративная медицина ещё находится на ранних этапах развития, системы часто ограничены лишь указанием на ближайший медицинский пункт или предоставлением базовой консультации без интеграции комплексного подхода к мониторингу и управлению здоровьем сотрудников [9]. Подобная узкая направленность не позволяет в полной мере реализовать потенциал СКМ и обеспечить интегрированное управление здоровьем сотрудников.

Также серьёзным препятствием для широкого распространения СКМ является высокая сложность и избыточность их инфраструктуры, особенно в крупных компаниях. Масштабные корпоративные медицинские системы часто требуют интеграции множества программных компонентов, аппаратного оборудования и сетевых решений, что приводит к увеличению сложности и существенному росту затрат на эксплуатацию и поддержку [10].

Не менее важным недостатком является высокий уровень финансовых и временных затрат, связанных с разработкой и внедрением таких систем «с нуля». Согласно исследованиям компании Deloitte, средний срок запуска полноценных корпоративных медицинских программ в крупных компаниях варьируется от 1 до 3 лет и требует значительных инвестиций, что может препятствовать их внедрению в организациях с ограниченными ресурсами [11].

Дополнительным фактором, затрудняющим эффективную работу СКМ, является сложность модернизации и адаптации существующих систем. Часто корпоративные медицинские системы базируются на устаревших программных и аппаратных платформах, что усложняет их дальнейшее развитие и

обновление. Например, в отчёте компании McKinsey отмечается, что многие компании сталкиваются с необходимостью полной перестройки медицинских систем при появлении новых технологических решений, что сопряжено со значительными затратами и операционными рисками [12].

Кроме того, управление СКМ при большом количестве персонала становится чрезвычайно трудоёмкой задачей. Сложности связаны с масштабируемостью таких систем, необходимостью оперативного сбора и обработки больших объёмов медицинских данных, а также постоянным взаимодействием с медицинским персоналом и сотрудниками компании. Отсутствие централизованного и эффективного управления может привести к ошибкам, задержкам в обслуживании и снижению общего качества медицинских услуг [12].

Наконец, одной из ключевых проблем современных СКМ является сложность их адаптации к новым заболеваниям и эпидемическим вызовам. Пандемия COVID-19 наглядно продемонстрировала необходимость гибких медицинских систем, способных быстро перестраиваться под меняющиеся эпидемиологические условия. Однако, согласно данным Всемирной организации здравоохранения, большинство существующих корпоративных систем здравоохранения не были готовы к оперативному реагированию на новые вызовы, что привело к перебоям в работе компаний и увеличению затрат на здравоохранение [12].

Таким образом, несмотря на широкое внедрение СКМ в корпоративную среду и их очевидные преимущества, решение указанных проблем требует дальнейшего совершенствования технологий, внедрения гибких подходов к проектированию и разработке, а также повышения уровня управления медицинскими ресурсами и процессами в компаниях.

1.2 Анализ литературы

Анализ литературы по теме "умная корпорация" выделяет существование полисемии [1, 3, 4], то есть множественного значения или толкования данного понятия. Для того чтобы исключить неоднозначное понимание терминов и создать единое понятийное представление, предлагается сформировать понятийный каркас, который бы отражал суть умной корпорации в контексте постиндустриального мира и был бы однозначно понятен заказчикам, проектировщикам и программистам [2].

Понятийный каркас умной корпорации:

Умная корпорация - Иерархическая крупномасштабная гетерогенная организационно-техническая система, основанный на информационных технологиях. Жизненный цикл этой системы определяется в значительной степени использованием и интеграцией современных информационных технологий во все аспекты ее деятельности.

Актор умной корпорации - Лица, организации, программные системы, технические устройства, процессы и другие сущности, которые интегрированы в инфраструктуру умной корпорации. В этом контексте акторы считаются субъектами, обладающими способностью обмениваться информацией через Интернет. Классификация акторов включает три основные группы:

- люди (n_i): физические лица, обладающие естественным интеллектом и способностью взаимодействия с системой;
- системы искусственного интеллекта (a_i): программные системы и алгоритмы, обладающие возможностью принятия решений и адаптации на основе анализа данных;
- технические устройства, объекты и процессы (po_i): включает в себя различные технические устройства, объекты и процессы, чьи параметры фиксируются с помощью датчиков и обрабатываются в рамках умной корпорации.

Этот понятийный каркас предоставляет основу для единообразного понимания умной корпорации и ее акторов в контексте современных технологий и информационного общества. Он помогает установить общий язык и понимание среди участников процесса проектирования, разработки и эксплуатации систем умных корпораций (рисунок 1.3).

$X = f(\text{сигналы от датчиков}) \in [0.00 \dots 1.00]$									
adr id ni	ai	Входной вектор							
		V	E	X1	X2	X3	...	Xn	U
		Эталоны							
		Состояния							Управления
		V1	E1 =	E11	E12	E13	...	E1n	U1
		V2	E2 =	E21	E22	E23	...	E2n	U2
		...	...	...	...	...	...	...	...
		Vm	Em =	Em1	Em2	Em3	...	Emn	Um
$X_1, \dots, X_j$ - бифуркационные показатели									

Рисунок 1.3 – Универсальная модель актора в задачах мониторинга

(n , m , X , X , V , U , E – задает эксперт с учетом специфики объекта наблюдения и среды)

Датчик (сенсор) – техническое устройство, фиксирующее определенные параметры актора, выполняющее их первичную обработку и отправку результата другим акторам. Датчик состоит из сенсора, фиксирующего значение

параметров, и контроллера, выполняющего их первичную обработку и отправку по определенному адресу.

Администрация (центр) – актер верхнего уровня иерархии, ответственный за управление жизненным циклом умного города.

Искусственный интеллект (ИИ) – специализированная самообучаемая программа (ai), решающая интеллектуальные задачи быстрее и лучше, чем человек. Как правило, ИИ включает функциональную часть, data base (DB) и knowledge base (KB).

Клиника – медицинское учреждение, обеспечивающие дистанционное наблюдение и медицинское обслуживание различных групп населения.

Пациент – актер наблюдаемой группы работников, находящийся под дистанционным медицинским наблюдением.

Система мониторинга группы работников – система наблюдения и синтеза управляющих решений для сотрудников, требующих оперативной медицинской помощи на основе данных, носимых или имплантированных пациентам датчиков (рисунок 1.4).

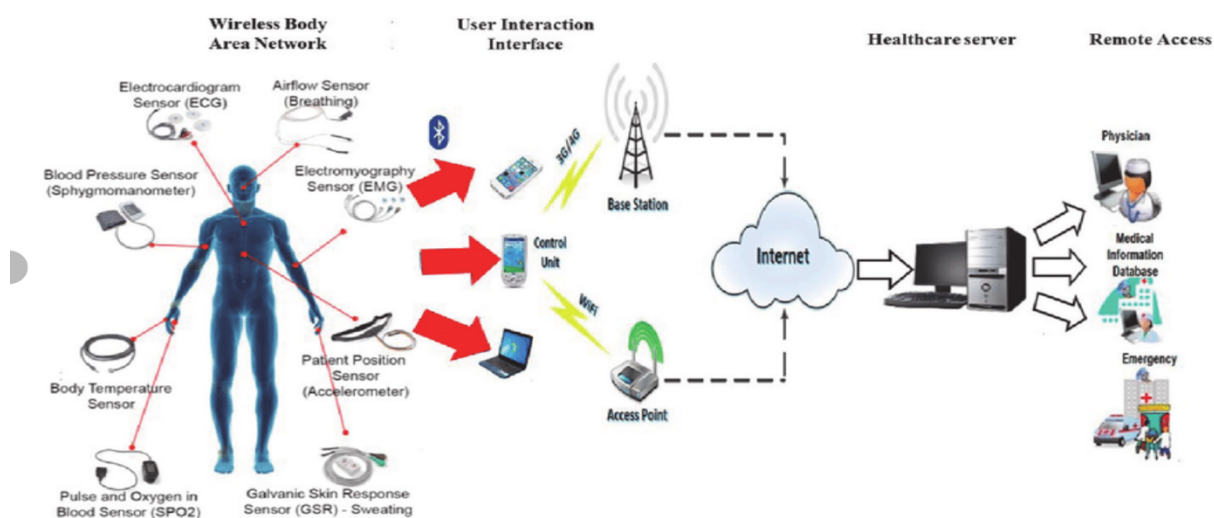


Рисунок 1.4 – Схема взаимодействия системы мониторинга здравоохранения

На основе данного набора определений формулируется задача, описывающая цель и способы решения задачи в контексте умной корпорации.

Парадигма решения задачи предусматривает рассмотрение компании в рамках двух сфер: сферы лечения врачами и сферы оперативного консультирования специализированным искусственным интеллектом (AI). Подобное совмещение человеческого и искусственного интеллекта позволяет

повысить качество диагностики и обеспечить своевременную адаптацию медицинских систем к появлению новых заболеваний и эпидемиологических вызовов [15].

Вместе с тем, внедрение и развитие таких систем сопряжено с рядом значительных трудностей. В частности, сложность изучения и обобщения литературы по корпоративной медицине обусловлена её относительной новизной и недостаточностью систематизированных данных и исследований [16]. Дополнительная проблема возникает из-за необходимости согласования требований менеджеров, медицинского персонала и сотрудников компании, что требует междисциплинарного подхода и эффективной коммуникации [17].

Также необходимы технологические решения, включающие разработку консольной версии системы для отработки взаимодействия с базами данных, что требует высокой квалификации разработчиков и аналитиков данных [18]. Кроме того, для обеспечения удобства использования и прозрачности процессов важна разработка интуитивно понятных графических модулей визуализации, доступных и понятных для менеджеров, врачей и персонала. Такие модули способствуют быстрому и точному восприятию данных и поддерживают принятие своевременных решений [19].

Современные исследования в области управления человеческими ресурсами, организационного поведения и корпоративного здравоохранения демонстрируют прямую зависимость между здоровьем персонала и производственными результатами компании. Исходя из этого, формула (рисунок 1.5) выражает функциональную зависимость эффективности компании (EffComp) от уровня здоровья персонала (HealthPersonal). В этой зависимости здоровье работников выступает ключевым медиатором, через который реализуются многие элементы корпоративной политики: снижение абсентеизма, рост производительности, повышение лояльности сотрудников и снижение текучести кадров.

$$\text{EffComp} = f_1 (\text{HealthPersonal})$$

Рисунок 1.5 – Формула зависимости эффективности компании от здоровья персонала

Здоровье персонала, в свою очередь, зависит от качества и уровня развития корпоративной медицинской системы (рисунок 1.6)

$$\text{HealthPersonal} = f_2 (\text{CorpMedSys})$$

Рисунок 1.6 – Формула зависимости здоровья от системы корпоративной медицины

Корпоративная медицинская система (СКМ) – это институционализированный механизм внутри компании, обеспечивающий профилактику, диагностику, лечение и сопровождение состояния здоровья работников. Это может включать как базовые программы (медосмотры, вакцинации, страховки), так и более развитые инициативы, включая ИИ-диагностику.

Однако сама корпоративная система медицины не является изолированным субъектом. Её эффективность определяется тремя основными составляющими (рисунок 1.7):

$$\text{CorpMedSys} = f_3(\text{TopMen}, \text{Doctors}, \text{AI})$$

Рисунок 1.7 – Система зависит от топ-менеджера, врачей и доступа к AI

где: TopMen – управленческое звено, принимающее стратегические решения, включая инвестиции в здоровье сотрудников; Doctors – квалифицированный медицинский персонал, обеспечивающий практическую реализацию программ: диагностику, консультирование, медицинское сопровождение; AI (искусственный интеллект) – современные цифровые инструменты, обеспечивающие поддержку в виде анализа больших данных о здоровье, предиктивной аналитики, автоматизации диагностики, управления записями и взаимодействия с пользователями. Таким образом, мы получаем логическую цепочку (1.1):

$$\text{EffComp} = f_1(f_2(f_3(\text{TopMen}, \text{Doctors}, \text{AI}))) \quad (1.1)$$

Иначе говоря, стратегические решения топ-менеджмента, качество и доступность врачебной помощи, а также уровень технологической интеграции (в том числе на основе ИИ) последовательно определяют состояние здоровья сотрудников, которое, в свою очередь, определяет эффективность всей компании.

Компании, игнорирующие необходимость построения СКМ, сталкиваются с целым рядом рисков:

- повышенный уровень заболеваемости;
- рост затрат на временную замену ключевых специалистов;
- снижение вовлечённости и морального состояния команды;
- потеря репутации как работодателя, заботящегося о сотрудниках;

– повышение текучести кадров и снижение привлекательности на рынке труда.

На фоне глобальных трендов – роста числа хронических заболеваний, стресса, ухудшения ментального здоровья и старения рабочей силы – наличие полноценной и адаптивной СКМ становится не просто желательной, а критически необходимой для поддержания конкурентоспособности.

Концептуальная модель показывает, что инвестиции в здоровье сотрудников – это не затраты, а стратегически важные вложения. Комплексный подход, включающий управленческое видение, профессиональную медицинскую поддержку и использование современных технологий (в частности ИИ), формирует устойчивую СКМ. Такая система становится фундаментом для поддержания высокой продуктивности, инновационности и устойчивости бизнеса в условиях современной экономики.

Если компания не создаёт или игнорирует развитие СКМ, она теряет не только внутреннюю устойчивость, но и рыночные преимущества в долгосрочной перспективе.

В дополнение к изложенному концептуальному базису целесообразно рассмотреть визуализированные схемы, раскрывающие логику функционирования и архитектуру корпоративной системы медицины (СКМ) в компании. Эти схемы позволяют более точно понять, как реализуется взаимодействие между ключевыми участниками процесса и какие технические компоненты лежат в основе функционирования системы (рисунок 1.8).

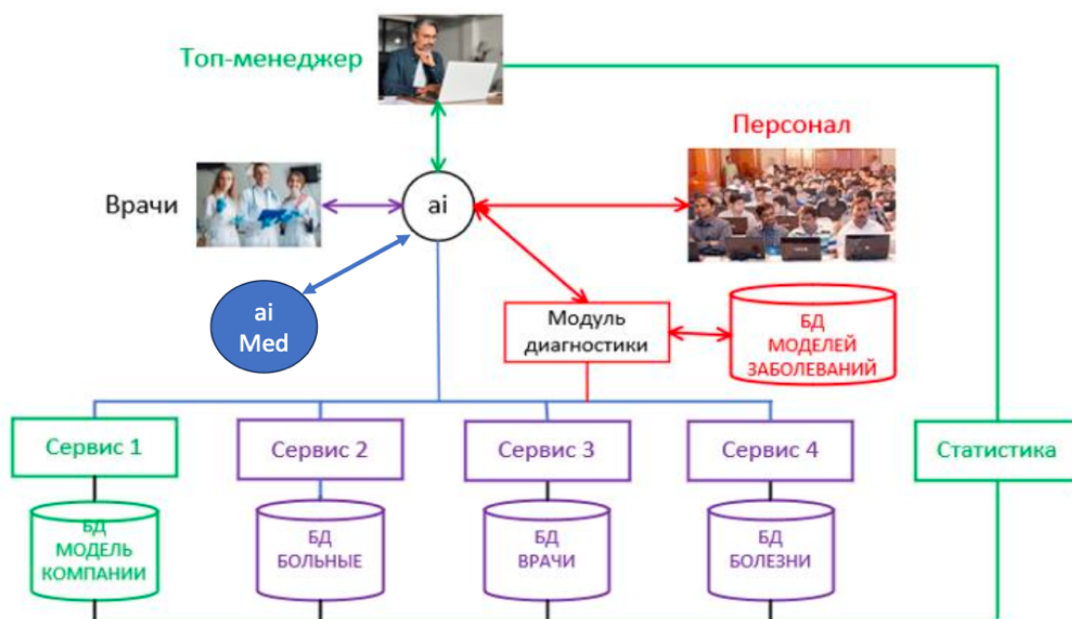


Рисунок 1.8 – Схема функционирования СКМ

На данной схеме представлена логика взаимодействия основных субъектов внутри СКМ: топ-менеджмента, персонала, врачей и искусственного интеллекта (AI), а также различных сервисов и баз данных, обслуживающих систему.

Ключевые элементы:

AI – центральный аналитический и координирующий узел, обеспечивающий обработку информации, принятие решений, интеграцию данных и передачу их соответствующим участникам.

Топ-менеджер – формирует стратегические установки, влияет на параметры, по которым система определяет приоритетность и бюджетирование медицинских инициатив.

Врачи – получают доступ к аналитике и инструментам, предоставляемым AI, и обогащают систему клиническими данными и экспертными заключениями. Связь двусторонняя: врачи взаимодействуют с ИИ и получают обратную связь, что позволяет уточнять диагнозы и корректировать лечение.

Персонал – ключевой объект мониторинга. Их данные о состоянии здоровья передаются в модуль диагностики, который с помощью AI и базы моделей заболеваний анализирует состояние и риски.

aiMed – специализированный медицинский ИИ-модуль, направленный на глубокий анализ медицинских показателей, предиктивную диагностику и управление случаями заболеваний.

БД моделей заболеваний – содержит алгоритмы, шаблоны и описания наиболее распространенных или опасных заболеваний. Используется для раннего выявления и диагностики.

Сервис 1 (БД модели компании) – содержит информацию о структуре, рисках и внутренних процессах компании.

Сервис 2 (БД больных) – хранит персонализированную историю здоровья сотрудников.

Сервис 3 (БД врачей) – включает квалификации, специализации и расписание врачей.

Сервис 4 (БД болезни) – обобщённая база о болезнях, их симптомах, прогнозах и методах лечения.

Статистика – результирующий компонент, получающий агрегированные данные, на основе которых можно формировать отчёты, выявлять тренды и принимать стратегические решения.

Таким образом, эта схема демонстрирует, что AI действует как центральный элемент между управлением, медицинским персоналом и сотрудниками, поддерживая непрерывный поток информации и обратной связи.

Схема (рисунок 1.9) более детально фокусируется на структуре данных и агентно-ориентированной архитектуре системы. Основной акцент сделан на сборе, хранении и обработке информации из разных источников.

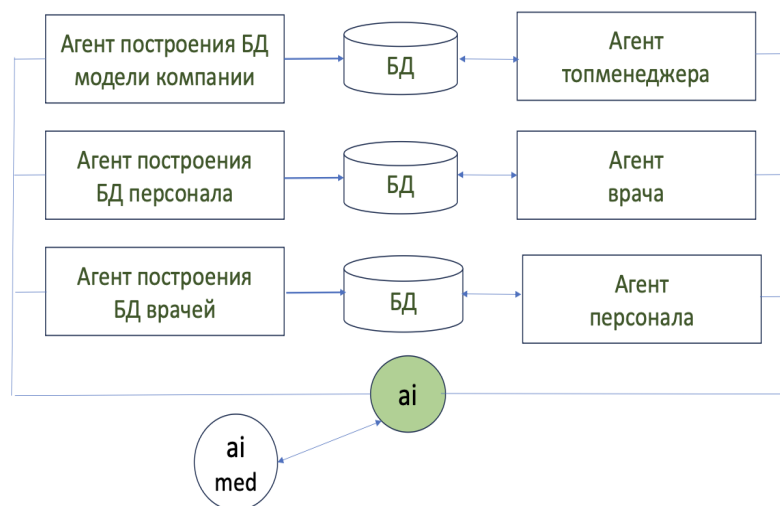


Рисунок 1.9 – Архитектура СКМ

Компоненты схемы:

- агенты построения баз данных: программные или организационные элементы, которые отвечают за сбор, верификацию и пополнение соответствующих баз данных;
- агент модели компании: анализирует структуру компании, её стратегические показатели, внутренние процессы, инициализирует связанные данные;
- агент персонала: собирает и актуализирует информацию о сотрудниках: медицинская история, участие в профилактических программах, реакции на лечение и т.п.;
- агент врачей: создаёт базу о врачах, их компетенциях, специализации, эффективности и распределении нагрузки;
- агенты субъектов: интерфейсы или алгоритмические посредники, представляющие интересы:
 - Топ-менеджера – взаимодействует с ИИ на стратегическом уровне, управляет приоритетами;
 - Врача – участвует в обмене данными с базами и ИИ;
 - Персонала – представляет интересы сотрудников, передаёт данные в систему и получает обратную связь.

Центральный элемент – AI – объединяет данные из всех источников, используя как собственные аналитические возможности, так и специализированный медицинский модуль aiMed для углублённого анализа. AI агрегирует информацию и выдаёт оптимальные стратегии воздействия, прогнозы, предиктивные модели и управленческие рекомендации.

Принцип функционирования архитектуры: каждый агент формирует и обменивается данными через свои БД, а центральный ИИ осуществляет агрегацию и интерпретацию информации, обеспечивая системную обратную связь и принятие решений.

Обе схемы функциональная и архитектурная являются взаимодополняющими. Первая демонстрирует динамическую логику взаимодействия субъектов и технологий, в то время как вторая фокусируется на структурной организации СКМ и логике построения базы знаний. Вместе они формируют целостную модель, в которой здоровье персонала становится не только объектом внимания, но и управляемым ресурсом, обеспечивающим рост и устойчивость всей компании.

Интеграция ИИ и агентно-ориентированных компонентов позволяет СКМ не просто обслуживать текущее состояние здоровья, но и действовать проактивно: предотвращать кризисы, адаптироваться к новым вызовам и поддерживать долгосрочную конкурентоспособность бизнеса.

1.3 Постановка задачи

Целью настоящей дипломной работы является исследование, моделирование и разработка системы мониторинга и поддержки здоровья персонала для средних и крупных компаний. Работа направлена на создание интегрированной технологической платформы, которая позволит компаниям в реальном времени отслеживать, анализировать и улучшать состояние здоровья сотрудников, тем самым обеспечивая устойчивое функционирование бизнес-процессов и повышение производственной эффективности.

Современная компания — это социально-экономическая система, зависящая от качества и продуктивности труда её сотрудников. В условиях глобализации, высокой конкуренции и постоянных изменений во внешней среде (включая эпидемиологические риски, стрессогенные факторы, удалённые форматы работы и пр.), здоровье персонала становится одним из ключевых факторов устойчивости бизнеса.

В такой динамической среде угроза ухудшения здоровья сотрудников напрямую сопряжена с риском снижения производительности, нарушениями в цепочках поставок, ростом затрат на замещение персонала и потерей

конкурентных преимуществ. Следовательно, поддержание здоровья сотрудников требует системного и технологически обоснованного подхода.

На основе вышеизложенного, задача настоящего исследования формулируется следующим образом:

1. Разработать научно обоснованную модель системы мониторинга и поддержки здоровья сотрудников компании, которая включает в себя:

- *анализ существующих методик мониторинга здоровья.*

На первом этапе проводится обзор текущих решений в области медицинской информатики, корпоративной медицины, телемедицинских сервисов и ИИ-систем, применяемых в крупных компаниях для мониторинга состояния здоровья работников. Особое внимание уделяется преимуществам и ограничениям этих подходов в корпоративном контексте.

Работники компании не являются однородной группой. Их потребности в медицинском сопровождении различаются в зависимости от профессии, условий труда, возраста, пола, хронических заболеваний и других факторов. Задача — выделить эти группы и определить, какие параметры здоровья необходимо мониторить для каждой из них;

- *формализация архитектуры системы мониторинга.*

На основе анализа существующих решений предлагается собственная архитектурная модель, включающая: агентов сбора данных, модули анализа, базы данных (сотрудников, врачей, заболеваний), диагностические алгоритмы и интерфейсы взаимодействия с пользователями. Архитектура должна предусматривать возможность масштабирования и адаптации под конкретные задачи бизнеса.

В рамках работы формируются алгоритмы диагностики и прогнозирования заболеваний на основе данных, полученных от медицинских осмотров, носимых устройств, анкетирования и цифровых платформ. Используются методы машинного обучения, вероятностного моделирования и экспертных систем.

Проектируемая система должна быть неотъемлемой частью общей ИТ-инфраструктуры компании. Это предполагает тесную интеграцию с кадровыми и управленческими платформами, корпоративными мессенджерами, внутренними порталами и системами корпоративной отчетности. Также важна возможность взаимодействия с внешними системами здравоохранения: клиниками, лабораториями, страховыми компаниями;

- *разработка прототипа системы.*

После формального моделирования разрабатывается прототип, который проходит апробацию в рамках пилотного проекта. В процессе тестирования

собираются данные о реальных сценариях использования, выявляются узкие места, и производится адаптация модели под реальные условия эксплуатации.

Для количественной оценки эффективности системы используются показатели: снижение числа больничных дней, сокращение расходов на временное замещение сотрудников, рост удовлетворенности и вовлечённости персонала, снижение текучести кадров и улучшение ключевых бизнес-метрик.

2. Сформировать техническое задание: разработать методологию и архитектуру для внедрения системы мониторинга и поддержки здоровья, включающей оперативное консультирование, автоматизированную диагностику, передачу рекомендаций врачам и менеджерам, и возможность быстрого реагирования в случае угроз состоянию здоровья персонала.

1.4 Выводы

В первой главе было сделано следующее:

- проведён комплексный анализ проблем корпоративной медицины, включая физические, психологические и организационные факторы, влияющие на здоровье персонала. Были выявлены ключевые вызовы, связанные с внедрением и эксплуатацией корпоративных медицинских систем (СКМ), включая сложности масштабирования, избыточность инфраструктуры и недостаточную гибкость при эпидемиологических угрозах;

- изучена и систематизирована литература по теме «умная корпорация», с акцентом на понятийный каркас, включающий классификацию акторов и их роли в инфраструктуре умной компании. Особое внимание было уделено интеграции ИИ, технических устройств и медицинских учреждений в единую цифровую среду;

- построены концептуальные модели, отражающие логические взаимосвязи между здоровьем персонала, эффективностью корпоративной медицины и общей продуктивностью компании. Были представлены визуальные схемы, раскрывающие архитектуру и принципы функционирования СКМ, а также роли ключевых участников: топ-менеджмента, врачей и ИИ;

- сформулирована цель дипломной работы и чётко определена исследовательская задача, включающая разработку модели системы мониторинга здоровья сотрудников, описание её компонентов, методов диагностики, интеграции с ИТ-инфраструктурой компании и критериев оценки эффективности.

Таким образом, проведенный анализ и формулировка задачи ставят перед проектом четкие направления для успешной разработки и внедрения системы мониторинга здоровья персонала крупной корпорации.

ГЛАВА 2. МОДЕЛИ И АЛГОРИТМЫ

2.1 Модели

2.1.1 Модель компании

Пусть имеется корпорация, осуществляющая медицинское сопровождение своих сотрудников в рамках внутренней Системы Корпоративной Медицины (СКМ). Управление процессами диагностики, консультирования и лечения осуществляется администрацией компании при поддержке интеллектуальной системы принятия решений, использующей методы анализа больших данных и искусственного интеллекта.

Внутренняя модель компании, как актора умной инфраструктуры, может быть представлена в виде следующего кортежа (2.1):

$$SCM = (idc, nP, nV, nZ, nVr, nAi) \quad (2.1)$$

где: idc : уникальный идентификатор компании, под управлением которой функционирует СКМ; nP : общее количество сотрудников, находящихся в зоне ответственности СКМ; nV : количество врачей, закреплённых за корпоративной системой (как внутренних, так и по договорам); nZ : текущее количество сотрудников, находящихся на лечении или под медицинским наблюдением (например, с активными случаями tZ); nVr : количество зарегистрированных обращений к врачам по всем каналам (личный приём, телемедицина, чат-боты); nAi : количество завершённых консультаций, проведённых с участием ИИ, включая автоматические рекомендации и диагностику низкой тяжести.

Каждый из указанных параметров играет ключевую роль в формировании управленческой отчётности и позволяет оценивать:

- степень укомплектованности корпоративной медицины;
- реальную загруженность врачей;
- уровень цифровизации процессов (через показатель nAi/nVr);
- потери компании из-за болезни персонала (среднее количество дней, которые сотрудник проводит на больничном);
- эффективность внедрённых профилактических программ.

Для полноценного функционирования СКМ необходима поддержка нескольких компонентов:

- централизованной базы данных сотрудников, включая медицинскую историю, хронические заболевания (hZ) и текущие жалобы (tZ);
- системы мониторинга обращений и симптомов;
- модуля статистического анализа и отчётности (для расчёта динамики ΔnZ , ΔnAi).

Таким образом, СКМ-модель компании в виде кортежа (idc , nP , nV , nZ , nVr , nAi) представляет собой верхнеуровневое описание корпоративной медицинской системы, обеспечивая агрегированный контроль за состоянием здоровья сотрудников и эффективность процессов диагностики, лечения и цифрового консультирования.

2.1.2 Унифицированная модель акторов

Каждый сотрудник корпорации в рамках СКМ рассматривается как индивидуальный актор (R), для которого осуществляется постоянный мониторинг состояния здоровья. Такой подход необходим для обеспечения персонализированной медицины, корректного назначения профилактических и лечебных мероприятий, а также для расчёта групповых рисков. Модель работника можно формализовать следующим образом (2.2):

$$R = (idp, kR, fioR, nV, pL, sP, hZ, tZ) \quad (2.2)$$

где: idp – уникальный идентификатор работника в рамках медицинской системы. Используется для привязки всех записей, включая назначения, обращения, историю болезней;

kR – внутренний код сотрудника, необходимый для интеграции с кадровыми и ERP-системами;

$fioR$ – фамилия, имя, отчество, используемые при формировании медкарты, регистрации визитов и взаимодействия с врачом;

nV – возраст сотрудника, критически важный параметр для анализа рисков, определения групп диспансеризации и расчёта индивидуальных индикаторов (например, норм давления, ЧСС и пр.);

pL – пол, применяемый в корректировке диагностических алгоритмов и при составлении прогнозов;

sP – должность или роль в компании, отражающая профессиональные нагрузки, график и вредные условия;

hZ – хронические заболевания, зарегистрированные в медкарте (например, диабет, гипертония);

tZ – текущие жалобы или диагностированные заболевания, зафиксированные на момент обращения или наблюдения.

Цели использования:

- формирование групп риска по возрасту, полу, нагрузке и хроническим заболеваниям;

- динамическая адаптация профилактических программ;

- оценка уровня индивидуального здоровья для КРІ отдела персонала;

- прогноз вероятности развития заболеваний с использованием AI-моделей.

Такая модель позволяет интегрировать медицинские показатели в общую систему управления человеческими ресурсами, связывая здоровье с производственной эффективностью.

2.1.3 Модель врача

В рамках СКМ каждый врач (активный участник медицинского обслуживания) представляется как специализированный актор (V), обладающий медицинской квалификацией, опытом и специализацией. Его модель описывает не только демографические параметры, но и профессиональный профиль, позволяющий автоматизировать назначение приёмов, анализ эффективности и распределение нагрузки. Онтология врача формализуется следующим образом (2.3):

$$V = (idv, kV, fioV, nS, mS, sR, kb, vP, sB, dB) \quad (2.3)$$

где: idv – уникальный идентификатор врача, однозначно определяющий специалиста в рамках СКМ;

kV – внутренний код врача, используемый в ИТ-инфраструктуре;

fioV – ФИО специалиста, отображаемое при назначении, в расписании, карточках и при обратной связи;

nS – научная степень (например, кандидат медицинских наук), используемая при классификации компетентности;

mS – медицинская специализация (терапевт, невролог, хирург и т. д.), которая определяет круг принимаемых пациентов;

sR – стаж работы, отображающий опыт специалиста и применяемый в системах расчёта приоритетов или нагрузки;

kb – номер кабинета, используется для маршрутизации внутри клиники;
vP – время приёма, доступное для записи пациентов;
sB – список принятых больных, агрегируемый по дням/неделям для расчёта загрузки;

dB – список выставленных диагнозов, используемый для аналитики, контроля качества и предиктивных моделей.

Функции модели:

- автоматизированное назначение врача на основании специализации и доступности;
- анализ эффективности врачей на основе статистики приёмов и исходов лечения;
- поддержка составления расписаний;
- формирование отчётов по загруженности, специализациям и востребованности.

Таким образом, модель врача служит основой для медицинского планирования, цифровой маршрутизации пациентов и аналитики качества оказания медицинских услуг. Наличие данной модели позволяет формализовать графики работы, определить степень нагрузки, соотнести квалификацию врача с профилем обращений, а также повысить прозрачность оказания медицинских услуг.

2.2 Алгоритмы последовательности действий на разных уровнях

Эффективное функционирование СКМ реализуется через логически взаимосвязанные действия, распределённые по трём ключевым уровням: работник (пациент), клиника и администрация. Такая модульная архитектура обеспечивает гибкость, масштабируемость и управляемость всей медицинской системы.

Уровень работника (пациента)

На этом уровне осуществляется первичное взаимодействие сотрудника с системой.

Регистрация симптомов: через мобильное приложение или интерфейс самообслуживания сотрудник вводит данные о самочувствии либо позволяет датчикам зафиксировать физиологические параметры.

Первичный анализ состояния: система на основе введённых данных (учитывая hZ, tZ, nV, pL) формирует вероятностную модель состояния.

Формирование управляющего решения U:

U = "наблюдать" — при нормальных показателях;

U = "направить к врачу" — при отклонениях;

U = "назначить приём" — при повышенных рисках;

U = "вызов бригады" — при критических состояниях.

Передача данных в клинику: создаётся задача для соответствующего врача или направления.

Уровень клиники

Здесь осуществляется обработка поступившей информации и оказание помощи: Получение информации: врач получает карточку (idp, fioR, tZ, hZ) и предписание от AI.

Назначение врача: на основании специализации mS и доступного времени vP.

Планирование и проведение приёма: очно или удалённо.

Фиксация результатов: обновление полей sB и dB.

Обратная запись в систему: закрытие обращения и обновление истории болезни в базе.

Уровень администрации компании

На этом уровне осуществляется контроль и стратегическое управление:

Анализ активности: сбор статистики по обращениям (nVr), текущим случаям (nZ) и AI-консультациям (nAi).

Оценка загруженности: вычисление коэффициента nVr/nV .

Прогнозирование и балансировка ресурсов: например, командировка врача в подразделение с высокой частотой обращений.

Формирование отчётности и KPI: автоматическая генерация аналитических сводок для управленческого звена.

2.3 Выводы

В результате выполнения анализа и проектирования, представленных во второй главе, были достигнуты следующие ключевые теоретико-методологические результаты, положенные в основу дальнейшей реализации системы мониторинга здоровья в корпоративной среде:

- выявлены все участники процесса мониторинга здоровья в корпорации, а также определены их взаимосвязи;
- создано формализованное описание концепций, акторов и их взаимосвязей в контексте системы мониторинга здоровья в корпорации;
- созданы алгоритмы, определяющие процесс построения модели компании на основе данных, полученных от умных датчиков и других источников;
- описаны последовательности действий, которые осуществляются на разных уровнях системы мониторинга здоровья - уровне пациента, уровне клиники и уровне администрации компании.

Таким образом, вторая глава заложила фундамент для последующих этапов проектирования и разработки, сформировав целостное представление о логике функционирования системы корпоративного мониторинга здоровья, а также обеспечила теоретическое и алгоритмическое обоснование её ключевых компонентов.

ГЛАВА 3. РАЗРАБОТКА ЦЕЛЕВОЙ СИСТЕМЫ.

3.1 Архитектура системы

Одной из ключевых сложностей при проектировании архитектуры систем мониторинга является их высокая вариативность и сложная структура. Каждая система требует индивидуального проектирования, что существенно увеличивает стоимость разработки, усложняет её поддержку и затрудняет масштабирование. В качестве решения предлагается унифицированная архитектура, не зависящая от специфики конкретного региона или города, что позволяет существенно упростить реализацию и сократить затраты.

Унифицированная архитектура системы мониторинга включает следующие основные компоненты: интеллектуальные модули, централизованная база данных клиник, коммуникационные каналы и умные датчики.

Интеллектуальные модули:

- aiP (Patient AI) — отвечает за первичную обработку данных от умных датчиков, установленных у пациента. Модуль анализирует физиологические параметры и оценивает текущее состояние здоровья;
- aiK (Clinic AI) — функционирует на уровне медицинского учреждения, обрабатывает поступающие данные от пациентов, формирует рекомендации и принимает решения о необходимых действиях;
- aiC (Company Administration AI) — координирует работу клиник и управляет ресурсами на уровне компании. Отвечает за анализ агрегированных данных и принятие стратегических решений.

Централизованная база данных клиник – это хранилище, в котором содержится медицинская информация о пациентах, результаты мониторинга, истории болезни и иные данные, необходимые для принятия обоснованных решений в клиниках и на уровне компании.

Коммуникационные каналы связывают все модули между собой и с базой данных, обеспечивая стабильный и оперативный обмен информацией в реальном времени.

Умные датчики являются основой всей системы. Они непрерывно собирают данные о состоянии здоровья пациентов и передают их в модуль aiP. После предварительного анализа информация направляется в aiK, где принимаются оперативные медицинские решения. В случае выявления системных проблем или перегрузки ресурсов в отдельных клиниках, модуль aiC вмешивается и перераспределяет нагрузку.

Таким образом, унифицированная архитектура позволяет создать гибкую и масштабируемую систему мониторинга, способную адаптироваться к условиям любой организации или региона без необходимости полного пересмотра её структуры.

Преимущества унифицированной архитектуры:

- снижение стоимости разработки.

Унифицированный подход позволяет значительно уменьшить затраты на разработку и внедрение систем мониторинга, так как одна и та же архитектура может быть использована в разных городах;

- упрощение модернизации.

Единая архитектура облегчает процесс обновления и модернизации программного кода, так как изменения могут быть применены к стандартной структуре без необходимости индивидуальных доработок для каждой системы;

- повышение надежности и эффективности.

Стандартизированные модули и коммуникационные каналы обеспечивают более высокую надежность системы и эффективность взаимодействия между различными компонентами системы мониторинга (рисунок 3.1).

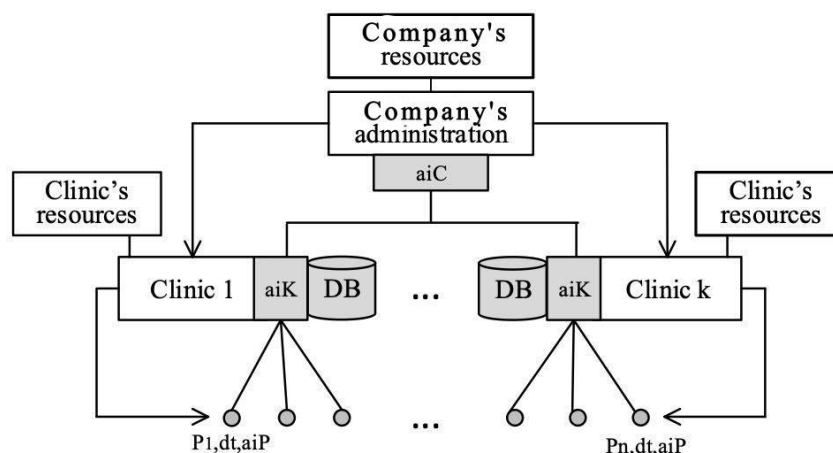


Рисунок 3.1 – Архитектура системы мониторинга здоровья компании

Данный вариант архитектуры обладает рядом ключевых свойств, которые делают его особенно привлекательным для внедрения в крупные корпорации и городские системы здравоохранения:

- универсальность.

Архитектура разработана таким образом, что она может быть применена в различных условиях, независимо от специфики конкретной компании или организации. Её внедрение не требует глубокой переработки под каждую

конкретную среду, что значительно упрощает масштабное распространение и снижает затраты на настройку;

- масштабируемость.

Система легко масштабируется в зависимости от размера и потребностей организации. Она может быть внедрена как на уровне отдельного офиса или учреждения, так и в рамках целой корпорации или городской инфраструктуры, при этом сохраняя стабильность и эффективность при росте объёма данных и количества пользователей;

- минимизация участия человека в принятии решений.

Одним из ключевых направлений развития архитектуры является постепенное снижение зависимости от человеческого фактора (niC, niK) в процессе принятия решений. По мере повышения точности и надёжности алгоритмов искусственного интеллекта, всё больше задач будет переходить под их контроль. Это позволяет оперативно реагировать на потенциальные угрозы, повышает точность и объективность медицинских решений, а также снижает издержки за счёт автоматизации процессов. Это имеет несколько важных преимуществ, таких как ускорение реакции на опасные ситуации, повышение качества медицинского обслуживания и экономия средств;

- оперативное управление ресурсами.

Особенность данной архитектуры заключается в возможности оценки деятельности клиник администрацией компании на основе "горячей" информации. Это позволяет оперативно маневрировать ресурсами. В случае эпидемий или других чрезвычайных ситуаций система позволяет быстро перераспределять ресурсы, направляя их в наиболее нуждающиеся районы или отделы.

Система предоставляет администрации компании или городским властям инструмент для оценки эффективности работы медицинских учреждений в режиме реального времени. Это позволяет оперативно принимать управленческие решения и способствует повышению качества медицинского обслуживания.

Несмотря на кажущуюся жёсткость архитектурной структуры, система сохраняет высокую степень гибкости за счёт возможности масштабируемого применения искусственного интеллекта на всех уровнях. ИИ может не только отслеживать состояние здоровья сотрудников, но и выполнять задачи по анализу эффективности работы клиник, отдельных подразделений и других организационных единиц. Такой подход позволяет сформировать более целостную картину функционирования системы здравоохранения.

Благодаря модульному принципу и технологической открытости, система может адаптироваться к новым требованиям и задачам. По мере развития

технологий и роста потребностей организаций, функциональность ИИ легко расширяется, обеспечивая устойчивость системы к внешним изменениям и повышая её долгосрочную ценность.

Таким образом, предложенная архитектура представляет собой гибкое, масштабируемое и эффективное решение, способное значительно улучшить управление здравоохранением в крупных корпорациях и городах, обеспечивая высокое качество медицинского обслуживания и экономическую эффективность.

3.2 Разработка программного обеспечения

Разработка и внедрение систем мониторинга компонентов умной компании отличается сложностью и высокой стоимостью. Для уменьшения стоимости на первой стадии используется имитационное моделирование, в процессе которого определяются и устраняются обнаруженные ошибки [12]. В данной работе представлен вариант, разработанный на языке программирования Swift с использованием фреймворков Foundation, SwiftUI, SwiftData, Charts, CryptoKit, XCTest.

Фреймворк Foundation предоставляет основные функции и API для работы с базовыми задачами программирования.

Фреймворк SwiftUI используется для реализации пользовательского интерфейса (UI), что обеспечивает декларативный подход к его описанию и гарантирует высокую производительность и отзывчивость при взаимодействии с системой.

Фреймворк SwiftData применяется для хранения и обработки данных, обеспечивающий безопасность, стабильность и целостность информации, а также интеграцию со SwiftUI.

Фреймворк Charts используется для создания графиков и визуализации данных в приложении.

Фреймворк CryptoKit используется для выполнения криптографических операций и обеспечения безопасности данных.

Фреймворк XCTest используется для написания и выполнения тестов в приложении.

Мобильное приложение представляет собой комплексную систему для корпоративной медицины, предоставляя доступ к ключевым данным и функциям в зависимости от роли пользователя.

Экран авторизации является ключевым элементом системы с точки зрения обеспечения безопасности и разграничения доступа к функциональности мобильного приложения. Его назначение — предоставить пользователю удобный

и защищённый способ входа в систему с учётной записью, соответствующей его роли (пациент, врач или администратор).

LoginView реализует идентификацию и аутентификацию пользователей системы. Компоненты экрана создаются на основе реактивной модели данных, а все изменения отображаются в реальном времени благодаря механизму @State и @ObservedObject (рисунок 3.2)

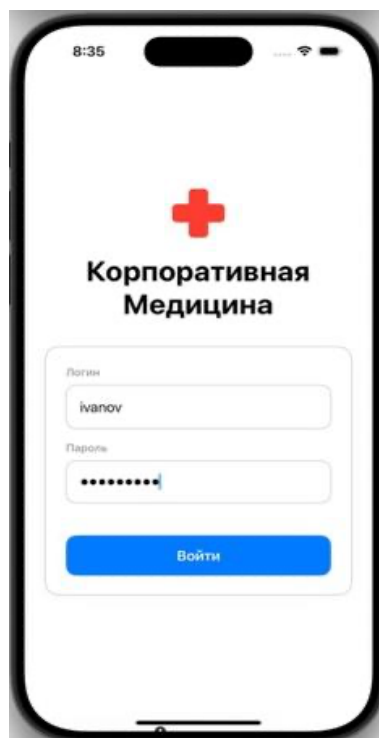


Рисунок 3.2 – Вход в систему

Интерфейс включает в себя два поля ввода: поле для логина (TextField), поле для пароля (SecureField). Поле логина использует стиль RoundedBorderTextFieldStyle и отключает автокапитализацию, чтобы предотвратить случайные ошибки при вводе адресов электронной почты или логинов. Пароль вводится в SecureField, который маскирует символы и предотвращает визуальный перехват данных.

Каждое поле снабжено валидацией: осуществляется базовая проверка на пустое значение, а также на корректность формата логина. Эти проверки проводятся до попытки авторизации, что снижает нагрузку на обработку и предотвращает ложные попытки входа (Приложение В).

Нажатие на кнопку «Войти» инициирует вызов метода, в котором происходит сверка введённых учётных данных с локальным хранилищем.

Если введённые логин и пароль соответствуют существующей записи, то система:

- устанавливает роль пользователя;
- инициирует переход на соответствующий интерфейс;
- сохраняет состояние аутентификации.

В случае неуспешной попытки входа система отображает информативное сообщение об ошибке, окрашенное в красный цвет. Это сообщение управляется через реактивное состояние, которое обновляется в зависимости от результата проверки.

Пароли в системе хранятся в захешированном виде. Для этого используется встроенная реализация криптографических хеш-функций. При авторизации введённый пароль хешируется и сверяется с хранимым значением, что обеспечивает соответствие базовым требованиям безопасности хранения данных.

Переход к другому представлению реализован через `NavLink` или программный переход, в зависимости от установленной роли. Таким образом, архитектура логина реализует шаблон *role-based navigation*, что критически важно для корпоративных приложений с дифференцированными уровнями доступа

Экран панели администратора является ключевым интерфейсом для управления корпоративной медицинской системой. Разработанный с учётом современных принципов дизайна интерфейсов пользователя (UI/UX), он обеспечивает максимальную функциональность при минимальном количестве действий (рисунок 3.3). В верхней части экрана отображается заголовок «Корпоративная медицина», подчёркивающий сферу применения и специализацию системы. Ниже находится подзаголовок «Панель администратора», ясно указывающий на административные полномочия текущего пользователя и его возможности управления. Центральная область экрана разделена на логические блоки, представленные крупными иконками и наименованиями модулей системы:

«Компании» — управление структурой организаций, входящих в корпоративную систему.

«Персонал» — администрирование сотрудников и их аккаунтов, включая роли и привязку к компаниям.

«Заболевания» — управление базой медицинских данных, содержащей симптомы, заболевания и методы лечения.

«Врачи» — полный контроль и управление специалистами медицинской части системы, включая их расписания и профили. В нижней части представлены дополнительные модули аналитической направленности:

«Обзор компаний» — предоставляющий обобщённые статистические данные по зарегистрированным компаниям и департаментам.

«Аналитика» — обеспечивающий глубокий анализ текущего состояния системы и её элементов, основываясь на различных метриках и данных в режиме реального времени.

«Аналитические отчеты» — отчеты о состоянии системы оказания медицинской помощи, основанные на различных данных в режиме реального времени.

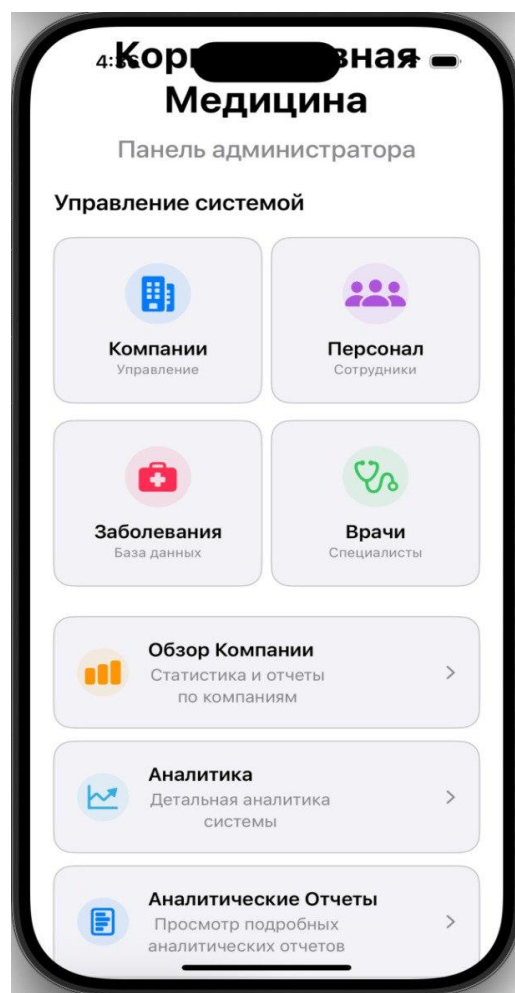


Рисунок 3.3 – Главное меню администрации компании

Экран «Управление компаниями» представляет собой важный элемент административного интерфейса (рисунок 3.4). Его основное назначение заключается в создании, отображении и редактировании организационной структуры юридических лиц, участвующих в экосистеме. Через данный экран администратор может управлять компаниями и их структурными подразделениями — департаментами.

Пользовательский интерфейс включает следующие основные компоненты:

Заголовок экрана — «Управление компаниями», оформленный с использованием крупного и жирного шрифта, что подчёркивает функциональную значимость интерфейса.

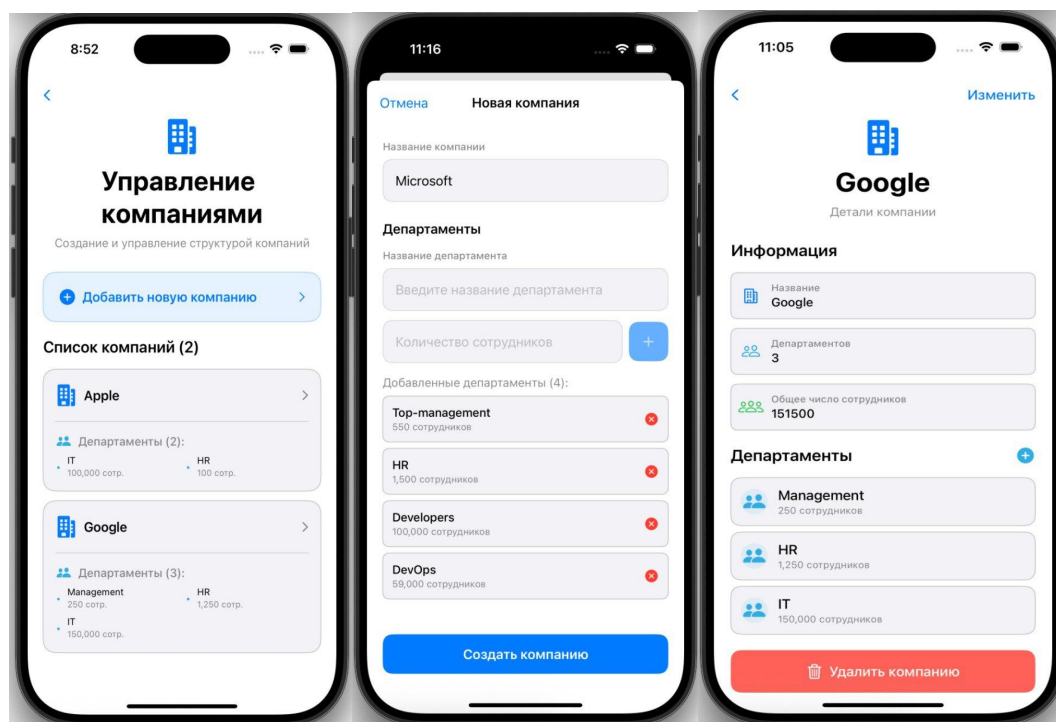


Рисунок 3.4 – Управление компаниями

Подзаголовок — поясняющий текст «Создание и управление структурой компаний», снижающий когнитивную нагрузку и способствующий лучшему пониманию пользователем функционального назначения интерфейса.

Кнопка действия — «Добавить новую компанию», реализованная как call-to-action элемент. При нажатии открывается модальная форма, позволяющая ввести наименование компании, определить количество и наименования департаментов, а также указать численность сотрудников в каждом из них.

Список компаний — представлен в виде карточек, каждая из которых содержит наименование организации, перечень её департаментов и численность сотрудников по каждому подразделению. Также предусмотрен элемент навигации, позволяющий перейти к подробной информации о конкретной компании.

Интерфейс построен с использованием компонентов VStack, HStack, NavigationLink и ForEach, что обеспечивает высокую гибкость, адаптивность и реактивность при изменении данных.

При загрузке экрана инициируется запрос @Query, получающий все записи из базы компаний. Данные автоматически сортируются по алфавиту, что

обеспечивает удобную навигацию при большом объёме информации. При добавлении новой компании используется модальная форма, в которой выполняется валидация входных данных.

Вся архитектура экрана построена на концепции однонаправленного потока данных (unidirectional data flow). Источниками истины (source of truth) выступают свойства `@Query` и `@State`. Любое изменение — добавление, редактирование или удаление данных — автоматически отображается в пользовательском интерфейсе благодаря реактивной природе SwiftUI.

Система предусматривает встроенные механизмы валидации на этапе создания и редактирования компаний: наименование компании не может быть пустым; численность сотрудников в департаментах должна быть ≥ 0 ; названия департаментов должны быть уникальны в пределах одной организации.

Экран проектировался с учётом возможного масштабирования. Возможности добавления фильтрации и сортировки по численности сотрудников, названию или другим мета-атрибутам, использования тегов или меток для классификации организаций по отраслям; подключение API-интерфейсов для импорта и экспорта данных из внешних корпоративных источников.

Раздел «Управление персоналом» представляет собой функциональный экран мобильного приложения системы корпоративной медицины, предназначенный для административной работы с данными сотрудников. Этот экран обеспечивает возможность централизованного взаимодействия с персональными сведениями зарегистрированных пользователей, предоставляя доступ к просмотру, добавлению и удалению записей о сотрудниках в рамках структуры компании.

Визуально экран оформлен в соответствии с единым стилем приложения и содержит заголовок, указывающий на назначение интерфейса, а также список сотрудников, представленных в виде отдельных карточек. Каждая карточка включает ключевую информацию о сотруднике, такую как фамилия, имя, отчество, пол, возраст, должность и адрес электронной почты. Данные элементы адаптируются под размер экрана и отображаются с учётом эстетических требований, включая скруглённые формы, светлый фон и контрастные надписи. В правой части каждой карточки располагается элемент управления, позволяющий инициировать процесс удаления сотрудника, что сопровождается сохранением изменений в локальной базе данных и обработкой возможных исключений.

С точки зрения архитектуры, экран основан на модели User (Приложение А), структура которой включает уникальный идентификатор, логин,

хешированный пароль, полные ФИО, дату рождения, пол, должность и электронную почту (рисунок 3.5).

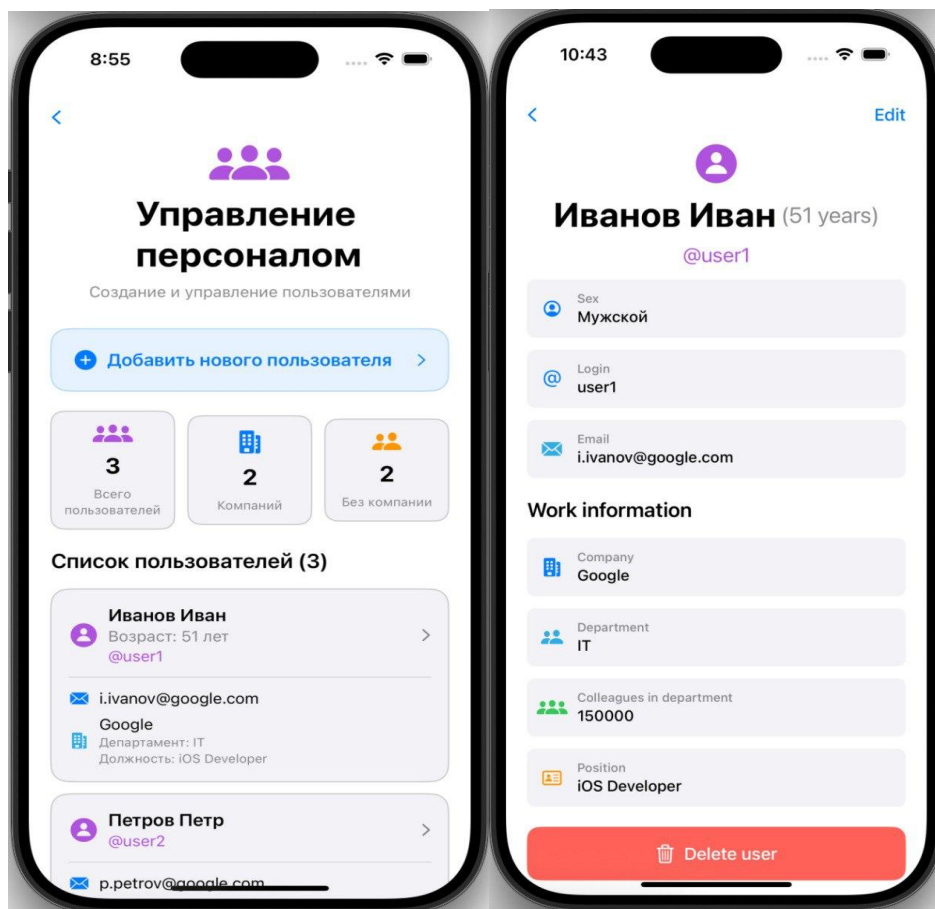


Рисунок 3.5 – Управление персоналом

Все данные обрабатываются в рамках привязки @Binding, обеспечивающей синхронизацию состояния между различными уровнями иерархии представления. Таким образом, любые изменения, произведённые на данном экране, автоматически распространяются на другие компоненты приложения, поддерживая целостность данных без необходимости принудительной синхронизации.

Внутренние вычисления, в частности, определение возраста пользователя, реализуются при помощи вычисляемого свойства, извлекающего значение из даты рождения. Валидация данных осуществляется непосредственно перед их визуализацией, что исключает вероятность отображения пустых или некорректных значений. Удаление записи сопровождается актуализацией хранилища посредством вызова соответствующего метода `saveUsers()`, который сохраняет изменения в модельном контексте `SwiftData`. Все действия, связанные

с изменением хранилища, обрачиваются в конструкции обработки исключений, что позволяет надёжно контролировать корректность операций.

Особое внимание в реализации экрана уделяется вопросам защиты персональных данных. В частности, чувствительные сведения, а именно пароли пользователей, обрабатываются в зашифрованном виде. Сама модель User проектируется с соблюдением требований по минимальному доступу и отсутствию прямого отображения конфиденциальной информации. Для шифрования паролей применяется функция хэширования SHA256 из фреймворка CryptoKit. Для промышленных реализаций может быть дополнительно использован алгоритм шифрования на базе AES-256.

Экран «База заболеваний» представляет собой специализированный модуль административного интерфейса системы корпоративной медицины, обеспечивающий централизованное хранение, редактирование и визуализацию информации о заболеваниях, их симптомах и методах лечения (рисунок 3.6).

Этот компонент системы направлен на формирование полноценных справочных данных, применимых как для целей внутреннего медицинского документооборота, так и для автоматизированного анализа при работе с пациентами и врачами. С точки зрения пользовательского взаимодействия, экран организован в виде навигационно-информационного пространства, в котором представлена вся доступная информация о заболеваниях. В верхней части интерфейса размещён заголовок, сопровождаемый кратким описанием его функционального назначения. Основная зона содержит кнопку для добавления нового заболевания, при активации которой открывается форма ввода на отдельном слое, реализованная с применением модального окна.

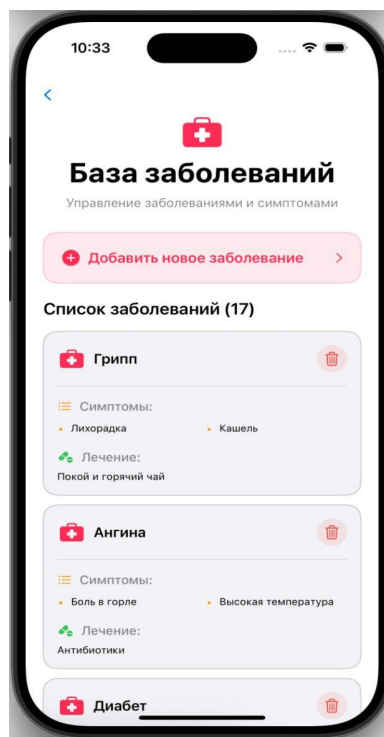


Рисунок 3.6 – База заболеваний

Хранилище данных использует модель Disease (Приложение А), содержащую уникальный идентификатор, строковые поля для наименования заболевания и лечения, а также массив строк для хранения списка симптомов. При добавлении нового заболевания осуществляется предварительная валидация всех полей. Система не позволяет сохранить запись, если не указано хотя бы одно симптоматическое проявление и не задана схема лечения. Таким образом, обеспечивается семантическая целостность справочника и исключается возможность появления неполных записей.

Для обеспечения устойчивости приложения к некорректным действиям предусмотрены методы обработки ошибок с логированием и защитой от критических сбоев. При ошибке сохранения информации в модельном контексте приложение продолжает работу в штатном режиме, а соответствующее сообщение регистрируется в отладочном журнале. Безопасность работы с медицинскими данными обеспечивается за счёт хранения всех записей локально в зашифрованном виде, при этом чувствительные данные не подвергаются передаче во внешние источники.

Экран отображения врачей представляет собой важный элемент административного интерфейса системы корпоративной медицины. Он обеспечивает доступ к актуальной информации о медицинских специалистах, зарегистрированных в системе, и предоставляет возможности для навигации, анализа и управления данными, связанными с профессиональной деятельностью

каждого врача (рисунок 3.7). Функционально данный экран решает задачу визуализации состава медицинского персонала, позволяя представить каждого специалиста в виде структурированной карточки. В карточке указываются имя и фамилия врача, его специальность, количество пациентов, находящихся у него на обслуживании, а также перечень заболеваний, с которыми данный специалист имеет опыт работы.

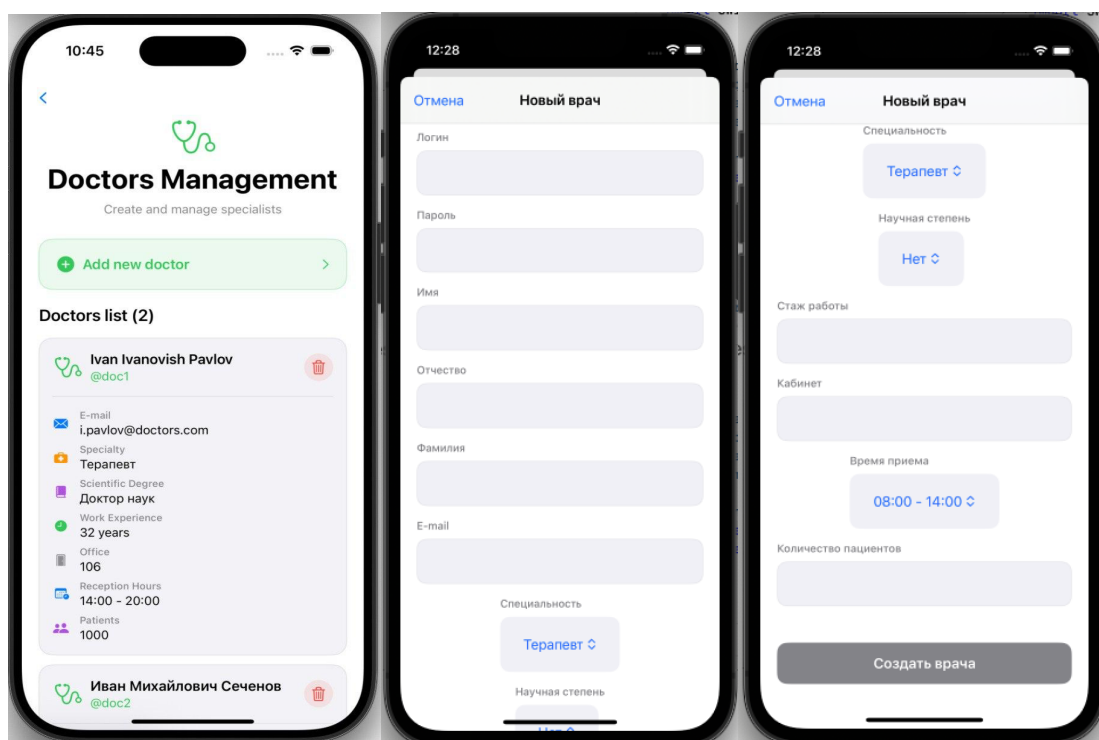


Рисунок 3.7 – Экран отображения врачей

Такая форма подачи информации повышает информативность представления, способствует быстрому принятию решений со стороны администратора и облегчает процессы маршрутизации пациентов в системе.

В разработанной системе каждый врач представлен как отдельный объект модели Doctor (Приложение А), содержащий идентификационные, контактные и профессиональные характеристики: уникальный идентификатор, логин и зашифрованный пароль, имя и фамилию, адрес электронной почты, область специализации, текущую нагрузку в виде количества обслуживаемых пациентов, а также перечень заболеваний, с которыми врач работает на регулярной основе. Модель также содержит связи с другими сущностями — в частности, с заявками на приём (Request), поступающими от пациентов, а также с персонализированным расписанием визитов.

Система предусматривает многокомпонентный интерфейс взаимодействия врача с платформой (рисунок 3.8).

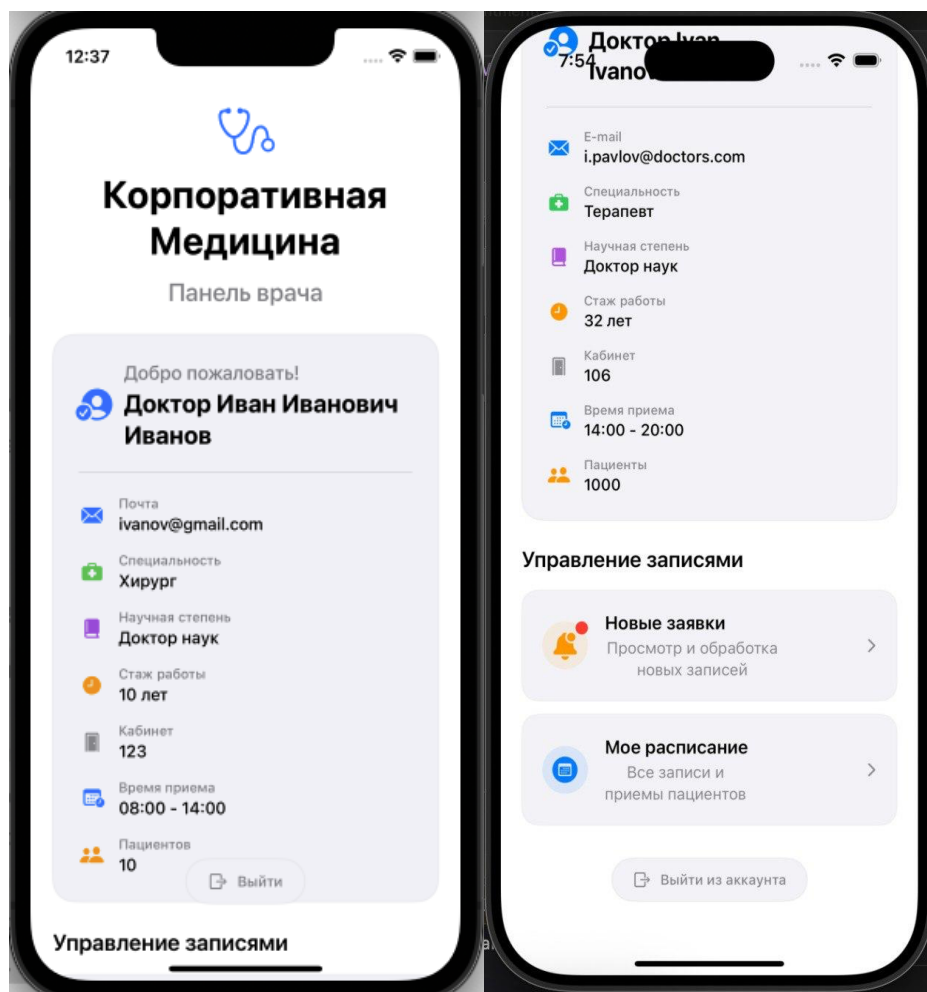


Рисунок 3.8 – Интерфейс врача

После прохождения аутентификации и выбора соответствующей роли пользователю открывается доступ к специализированному интерфейсу врача, в который включены следующие функциональные модули: просмотр и управление собственным расписанием, обработка новых заявок от пациентов, а также анализ завершённых визитов.

Расписание отображается в виде упорядоченного списка приёмов, каждый из которых содержит сведения о пациенте, дате визита, жалобах, диагнозе и назначениях. Такой формат способствует эффективной самоорганизации рабочего времени врача и позволяет вести непрерывное наблюдение за историей заболеваний.

Отдельную ценность представляет модуль обработки новых заявок. При поступлении запроса от пациента он отображается в интерфейсе врача как неподтверждённая заявка, требующая активного действия. Врач может принять, отклонить или отложить запрос, а при необходимости — инициировать обратную связь с пациентом.

Это обеспечивает гибкость маршрутизации в системе и позволяет учитывать реальную загруженность специалистов. После подтверждения заявка автоматически переходит в активное расписание, обеспечивая консистентность данных и единый источник истины для всех компонентов системы (рисунок 3.9).

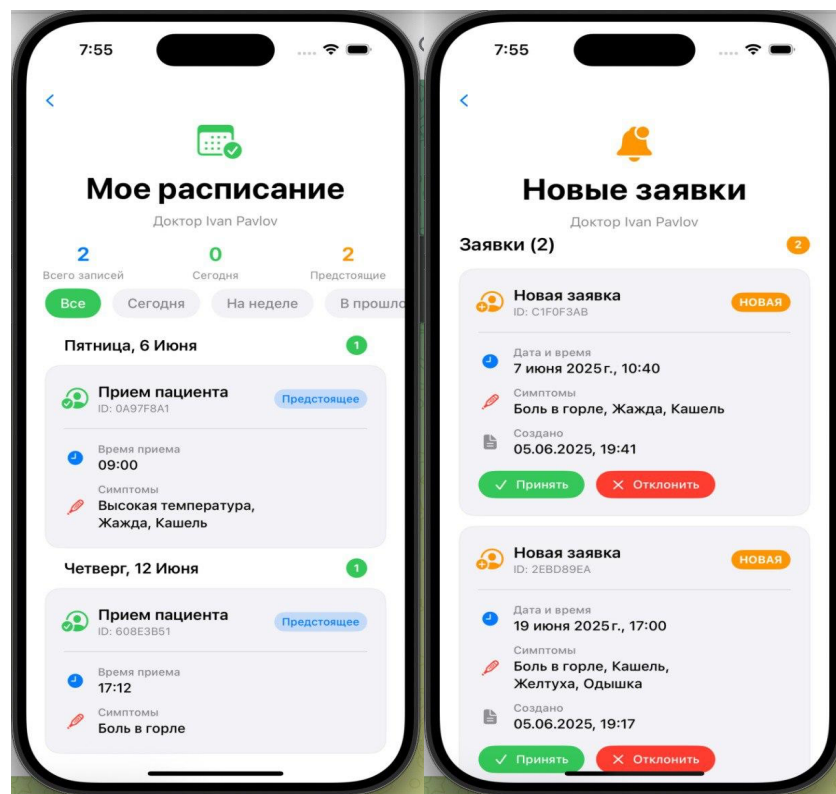


Рисунок 3.9 – Экран расписания врача, экран новых заявок

История приёмов формируется на основе завершённых заявок, что позволяет врачу анализировать свою деятельность, отслеживать повторные обращения и выявлять случаи хронических заболеваний.

В рамках разработанного программного обеспечения особое внимание уделено интерфейсу пациента, представляющему собой ключевую точку входа в функциональность мобильного приложения корпоративной медицины. Экран пользователя (UserView) реализует персонализированный подход, обеспечивая доступ к ряду медицинских сервисов в зависимости от учетной записи и текущего состояния пользователя, а также демонстрируя высокий уровень адаптивности и безопасности (рисунок 3.10).

При загрузке пользовательского интерфейса осуществляется автоматическое извлечение актуальных данных о заболеваниях и симптомах, потенциально доступных для диагностики, с применением реактивных механизмов отображения, что позволяет оперативно обновлять визуальное представление экрана без необходимости ручного вмешательства.

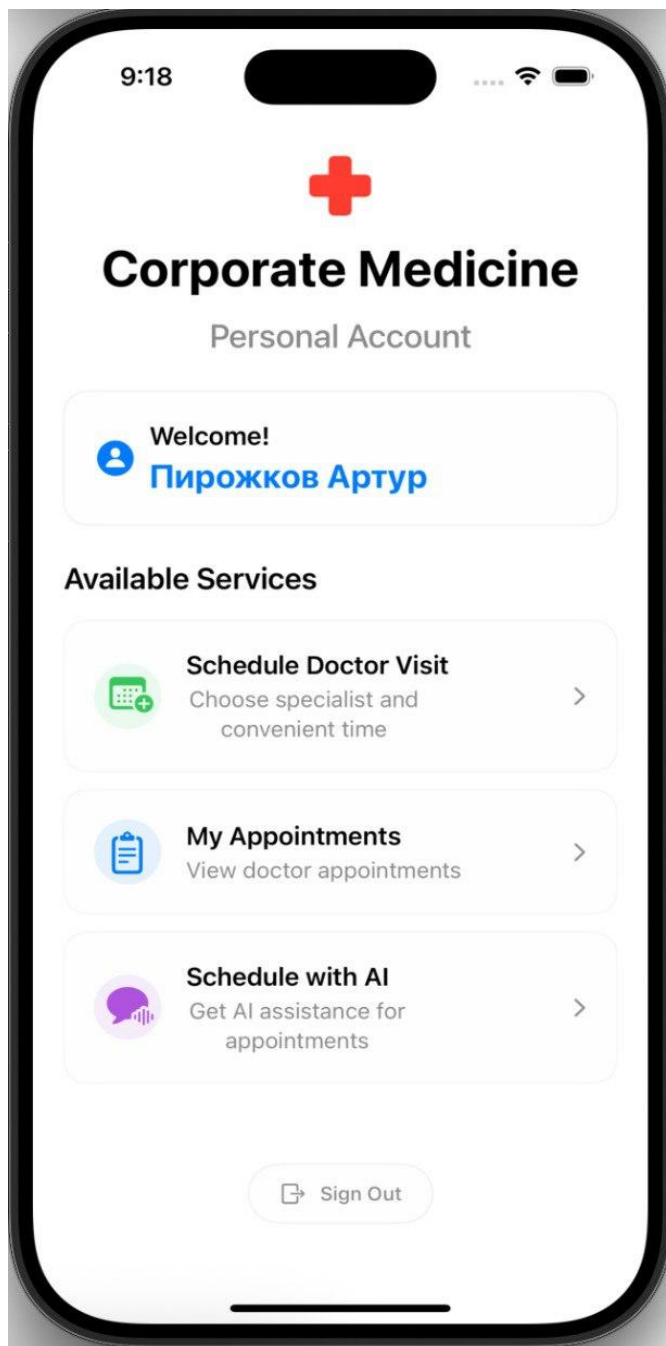


Рисунок 3.10 – Экран пользователя

Основной блок интерфейса включает приветственную панель с отображением имени пользователя, что способствует персонализации взаимодействия, а также информирует о текущей сессии авторизации.

Функционально экран предлагает несколько ключевых возможностей. Во-первых, реализован переход к модулю записи на приём к врачу (рисунок 3.11).

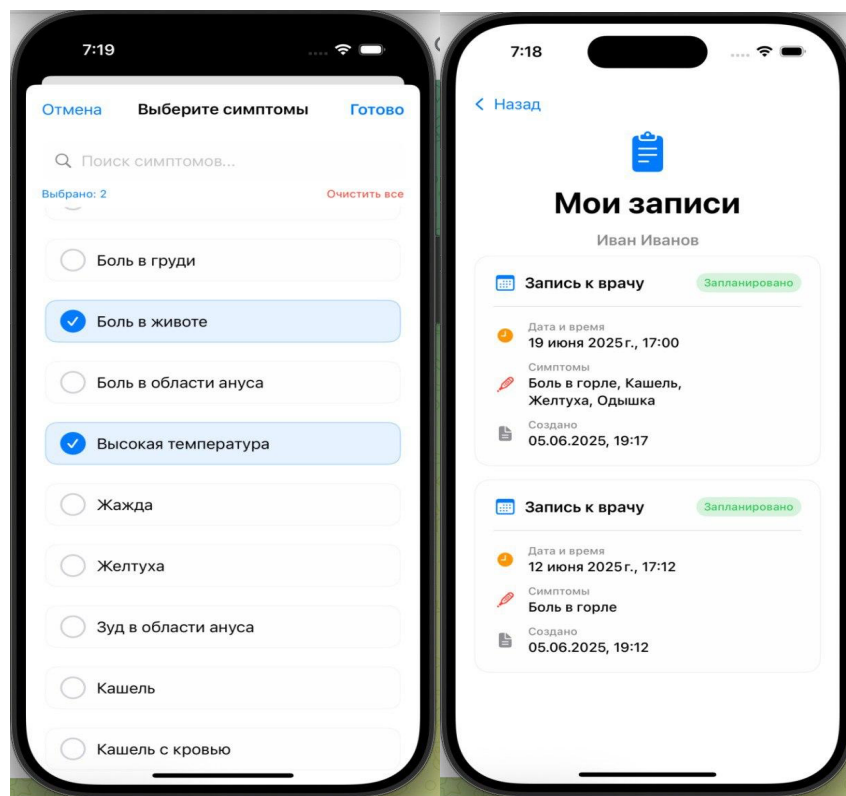


Рисунок 3.11 – Экран записи к врачу

Данный сценарий взаимодействия предоставляет пользователю интерфейс выбора специалиста и доступного времени приёма. Запись осуществляется через передачу соответствующих данных в систему, с последующей валидацией и сохранением в хранилище. Во-вторых, предусмотрен доступ к истории записей: пользователь может просматривать все прошедшие и запланированные визиты, а также видеть связанные с ними детали. Это обеспечивает прозрачность медицинского взаимодействия и способствует лучшему планированию.

Одним из ключевых элементов взаимодействия является выбор симптомов, реализованный через отдельный представленный экран `SymptomsSelectionView`. При этом пользователь может задать перечень наблюдаемых признаков заболевания, которые далее используются системой для подбора релевантных заболеваний. Логика реализована с применением двустороннего связывания `@Binding`, обеспечивающего синхронность между состоянием и пользовательским вводом. По завершении выбора пользователь получает результат — либо соответствующее заболевание, либо сообщение об отсутствии совпадений, с обработкой ошибок и уведомлением о возможных ограничениях модели.

В дополнение к основному сценарию, пользователю доступен модуль взаимодействия с виртуальным помощником на основе искусственного интеллекта (`ChatView`), в рамках которого возможно получение рекомендаций по

симптомам, ориентировочных предварительных заключений и направлений на приём (Приложение Б).

Это реализовано через компонент чата, поддерживающий двустороннюю коммуникацию, с возможностью расширения логики через интеграцию NLP-моделей (рисунок 3.12).

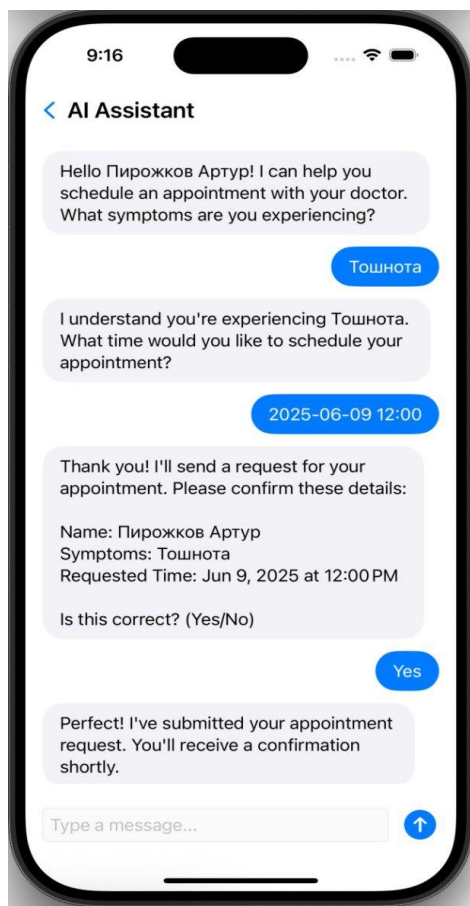


Рисунок 3.12 – Экран AI-ассистента

Был разработан отдельный модуль аналитики, реализующий ключевые функции мониторинга состояния системы и анализа статистических данных. Экран *AnalyticsView* служит центральной точкой доступа к различным видам аналитических отчётов, в том числе по корпоративной активности, медицинским показателям и динамике поведения пользователей. Отчёты представлены в виде интуитивно понятных визуализаций и карточек, каждая из которых содержит агрегированные метрики, отражающие состояние соответствующих компонентов системы.

Основной задачей данного экрана является обеспечение руководителей и администраторов достоверной и актуальной статистикой, позволяющей принимать обоснованные управленческие решения. Так, отчёт *CompanyImage* предоставляет сравнительный график по ключевым метрикам организаций,

включая общее число пользователей, среднее количество визитов к врачу, частоту использования ИИ-чата и среднюю продолжительность отсутствий на рабочем месте из-за болезни. Отчёты *Dynamics of the Sick* и *Dynamics of the Absence of Employees* визуализируют ежемесячное распределение обращений и среднюю продолжительность болезней соответственно (Приложение В). Все графики реализованы с использованием компонента *Charts* и адаптированы для отображения на мобильных устройствах (рисунок 3.13).

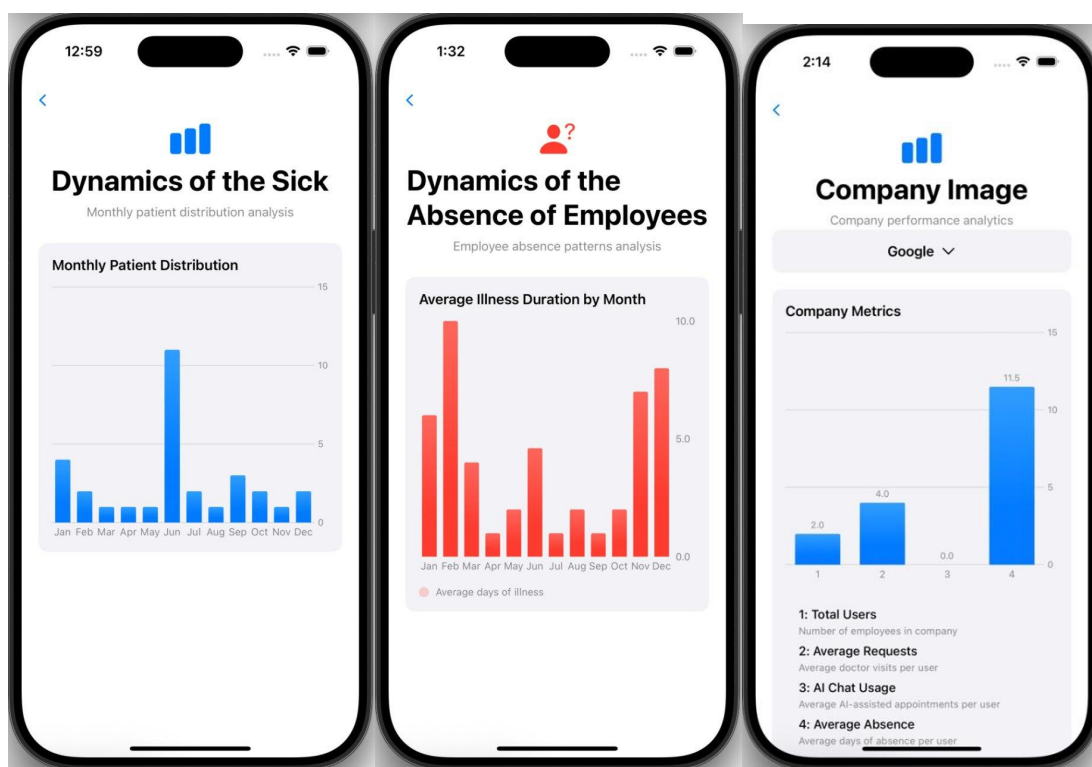


Рисунок 3.13 – Аналитика

В процессе разработки мобильного приложения для системы корпоративной медицины особое внимание было уделено вопросам масштабируемости, интернационализации, модульности и соответствию современным практикам промышленной разработки программного обеспечения. Приложение изначально было локализовано на три языка: русский, английский и французский, что обеспечивает его доступность для широкой аудитории, включая международные компании с многоязычным составом сотрудников (рисунок 3.14). Локализация охватывает все пользовательские интерфейсы, элементы управления, системные уведомления и диагностические сообщения. При этом архитектура приложения построена с учётом последующего масштабирования — внедрение новых языков возможно без модификации основной логики, путём добавления соответствующих *strings*-файлов и

подключаемых языковых пакетов, что соответствует принципам разработки с учётом интернационализации и локализации.

Графическая оболочка приложения реализована с поддержкой двух цветовых тем — светлой и тёмной, что не только повышает эстетическое восприятие интерфейса, но и соответствует требованиям доступности, позволяя адаптироваться к предпочтениям пользователя и условиям внешнего освещения. Переключение тем происходит автоматически, в зависимости от системных настроек операционной системы, благодаря использованию параметров среды @Environment(\.colorScheme).

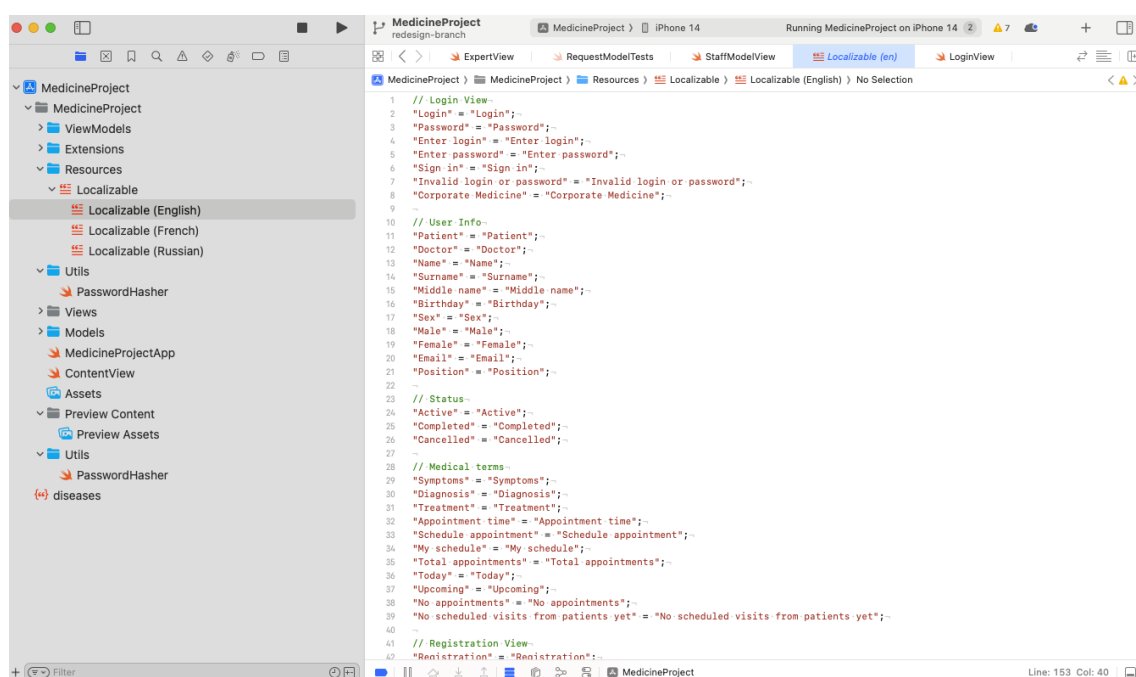


Рисунок 3.14 – Локализация приложения

С точки зрения обеспечения качества программного обеспечения, приложение покрыто модульными юнит-тестами, реализованными с использованием фреймворка XCTest. Покрывание охватывает критически важные участки бизнес-логики, включая модели данных, обработку пользовательского ввода, валидацию форм, маршрутизацию экранов и транзакции сохранения данных в локальном хранилище. Тестирование проводится в автоматическом режиме при каждом обновлении кода, что обеспечивает стабильность и контроль над регрессиями (рисунок 3.15).

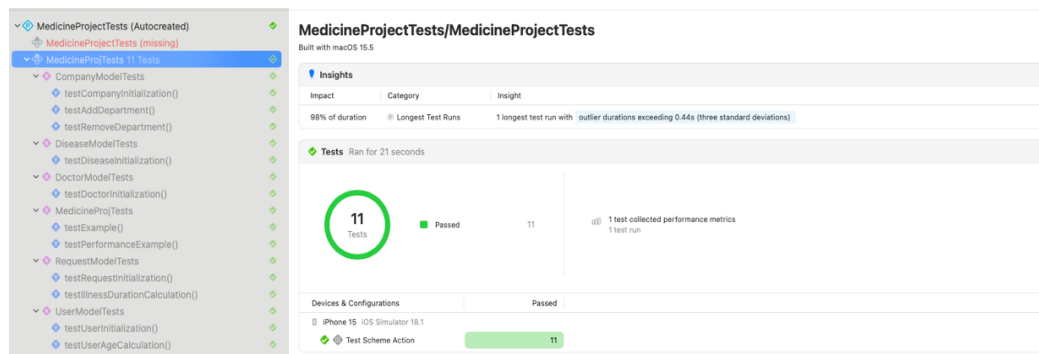


Рисунок 3.15 – Unit testing

Разработка проекта осуществлялась с использованием распределённой системы контроля версий Git и размещением основного репозитория на платформе GitHub, что позволило реализовать надёжную систему командной работы.

Проект имеет полностью настроенный CI/CD-пайплайн, автоматизирующий ключевые этапы жизненного цикла программного продукта. Инфраструктура CI/CD реализована с использованием GitHub Actions (рисунок 3.16), что обеспечивает гибкость и масштабируемость процесса автоматизации, а также упрощает интеграцию дополнительных этапов в будущем (например, автоматическое формирование changelog или статическую проверку безопасности кода).



Рисунок 3.16 – CI/CD

Разработка велась согласно современным практикам инженерной культуры: каждая новая функциональность реализовывалась в отдельной ветке (feature branch), затем проходила стадию ревью и слияния через pull request в основную ветку (main). Такой подход обеспечивает изоляцию изменений, облегчает ревью кода, а также повышает воспроизводимость и управляемость изменений (рисунок 3.17).

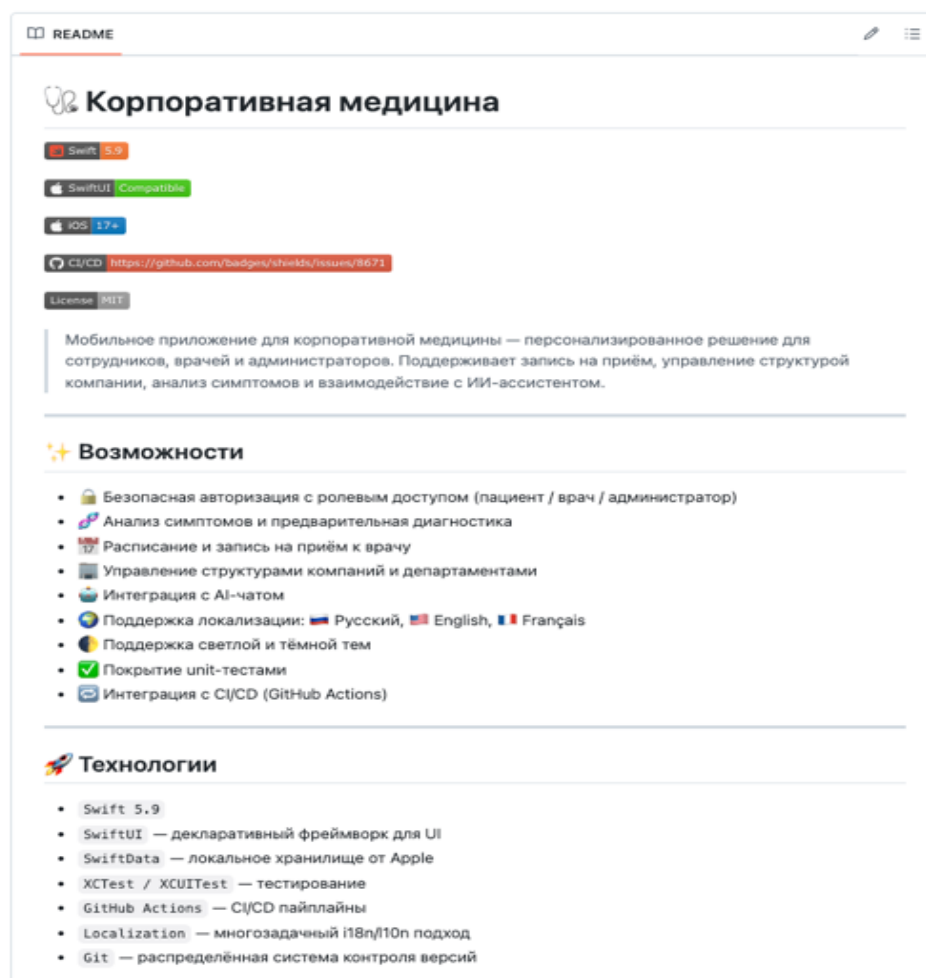


Рисунок 3.17 – GitHub репозиторий

Таким образом, данное приложение представляет собой не только функционально полноценную медицинскую платформу, но и пример зрелой инженерной реализации, соответствующей современным требованиям к мобильной разработке, включая интернационализацию, визуальную адаптацию, тестируемость, поддержку командной разработки и автоматизированный процесс доставки.

3.3 Выводы

В результате выполнения работ, представленных в данной главе, были достигнуты следующие значимые результаты, подтверждающие реализацию ключевых этапов проектирования и разработки системы мониторинга здоровья персонала в рамках концепции умной компании:

- описан визуальный интерфейс приложения;
- проведен детальный анализ и проектирование архитектуры системы мониторинга здоровья персонала умной компании;
- исследованы основные компоненты, их взаимосвязь и функциональные характеристики;
- осуществлена разработка программного обеспечения.

Таким образом, третья глава демонстрирует полную проработку как визуальной, так и технической стороны проекта, а также подтверждает соответствие созданного программного обеспечения современным требованиям к корпоративным медицинским системам.

ГЛАВА 4.

ПРИМЕНЕНИЕ СИСТЕМЫ ДЛЯ РЕШЕНИЯ ПРИКЛАДНОЙ ЗАДАЧИ

4.1 Постановка задачи

В современном корпоративном здравоохранении важным аспектом является обеспечение своевременной диагностики и предотвращение заболеваний среди сотрудников компании. В связи с этим, рассматриваем систему для автоматического направления сотрудников к врачам на основе выбранных симптомов. Эта система представляет собой технологию построения адаптивных систем управления корпоративной медициной, предназначенную для эффективного мониторинга состояния здоровья сотрудников, улучшения скорости реагирования на заболевания и оптимизации взаимодействия между сотрудниками, врачами и администраторами.

Сотрудники компании используют специализированное мобильное приложение, в котором они могут выбрать симптомы, с которыми они сталкиваются. Например, если сотрудник ощущает недомогание — головную боль, жар или другие симптомы, он может воспользоваться приложением, чтобы выбрать подходящие жалобы из предложенного списка. После ввода данных система анализирует симптомы с помощью алгоритмов искусственного интеллекта и встроенной базы медицинских знаний. В результате формируется перечень возможных диагнозов, соответствующих описанным признакам.

На основе этого предварительного анализа система автоматически направляет сотрудника к профильному специалисту — терапевту, пульмонологу или другому врачу, в зависимости от характера симптомов. Врач получает уведомление в своём мобильном приложении, где отображаются собранные данные о пациенте, его жалобах и предварительных рекомендациях. После ознакомления специалист может подтвердить назначение, скорректировать его при необходимости, назначить лечение и внести соответствующие изменения в электронную карту пациента.

Менеджер компании при этом получает доступ к аналитической панели, где отображается статистика обращений, загруженность медицинского персонала и другие ключевые показатели. Это позволяет не только оперативно отслеживать ситуацию, но и принимать обоснованные управленческие решения, направленные на повышение эффективности корпоративной медицины.

Таким образом, технологическая основа подобной системы обеспечивает автоматизацию процессов диагностики, маршрутизации и управления медицинскими обращениями. Это способствует более слаженной работе всех

участников — сотрудников, врачей и менеджмента — и напрямую влияет на улучшение качества медицинского обслуживания внутри организации.

4.2 Решение задачи на основе методики

Решение задачи автоматизации направления сотрудников к врачам на основе симптомов можно представить в виде последовательных этапов, каждый из которых выполняет ключевую роль в обеспечении оперативности и точности медицинского обслуживания.

Этап 1. Выбор симптомов

Первоначально сотрудник через мобильное приложение выбирает симптомы, которые он испытывает. Интерфейс разработан так, чтобы упростить навигацию и минимизировать ошибки: пользователь видит список распространённых симптомов (например, головная боль, кашель, повышение температуры), а также может получить уточняющие подсказки. Это позволяет точнее описать состояние и обеспечивает качественный входной сигнал для дальнейшего анализа (рисунок 4.1).

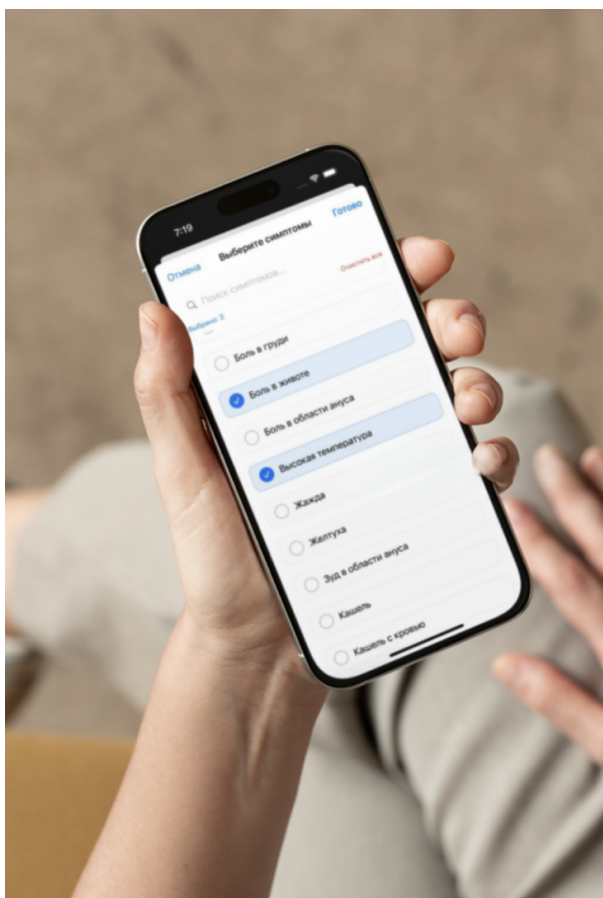


Рисунок 4.1 – Выбор симптомов

Этап 2. Анализ симптомов и формирование возможных диагнозов

На этом этапе система обрабатывает введённые данные с помощью алгоритмов машинного обучения и ИИ, сопоставляя их с медицинской базой знаний. Результатом становится список вероятных заболеваний, соответствующих описанным симптомам.

Если в системе уже имеются данные о хронических или ранее зафиксированных заболеваниях конкретного сотрудника, они также учитываются при формировании рекомендаций. Благодаря этому система может предложить персонализированные варианты диагностики и направить сотрудника к наиболее релевантному специалисту.

Этап 3. Автоматическое направление к врачу

После определения вероятного диагноза система автоматически выбирает подходящего врача и направляет к нему сотрудника. Медицинский специалист получает уведомление в своём приложении с подробной информацией о жалобах и предполагаемом диагнозе.

Врач может подтвердить или уточнить диагноз, назначить необходимые обследования, зафиксировать план лечения и обновить данные о пациенте в общей системе. Все действия документируются, обеспечивая прозрачность и возможность дальнейшего мониторинга (рисунок 4.2).

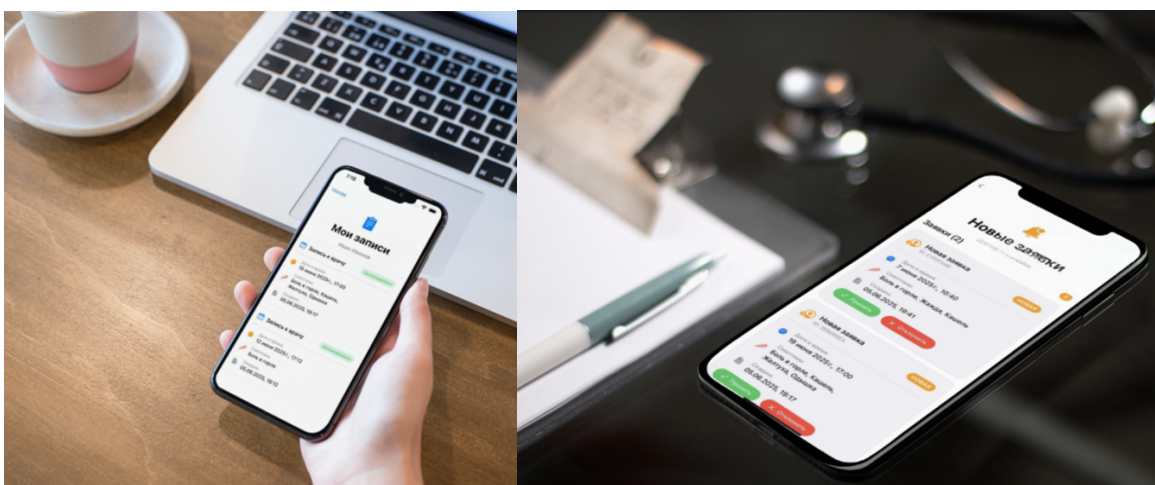


Рисунок 4.2 – Запись к доктору

Этап 4. Мониторинг и управление системой со стороны администратора

Завершающий этап — управление системой через административную панель. Менеджер компании получает доступ к аналитике: количество обращений, распределение по симптомам и диагнозам, загрузка врачей. Эти данные позволяют контролировать эффективность работы системы и оперативно реагировать на дисбаланс нагрузки или рост заболеваемости (рисунок 4.3).



Рисунок 4.3 – Анализ компании менеджером

Панель управления позволяет также обновлять базу данных заболеваний, устанавливать новые правила для направления к врачам и настраивать систему в зависимости от меняющихся потребностей компании.

Технология построения адаптивных систем управления корпоративной медициной позволяет создать интеллектуальную, самобалансирующуюся систему, которая в реальном времени реагирует на изменения состояния здоровья сотрудников, оперативно направляет их к врачам, а также предоставляет администраторам необходимые инструменты для управления ресурсами.

Основная цель системы — это интеграция процессов диагностики, лечения и управления медицинскими услугами в одном едином потоке, что способствует не только улучшению качества жизни сотрудников, но и повышению общей эффективности работы компании, снижению заболеваемости и уменьшению количества пропусков на рабочем месте из-за болезни.

4.3 Выводы

В четвертой главе получены следующие основные результаты:

- проведена постановка конкретной прикладной задачи, которую предполагается решить с использованием разработанной системы;
- четко определены цели и требования к решению данной задачи;
- подробно описан процесс решения поставленной задачи с применением разработанной системы;
- приведены шаги, методы и алгоритмы, использованные при решении задачи.

Таким образом, четвёртая глава наглядно продемонстрировала практическую применимость разработанной системы и её способность эффективно решать задачи цифровизации медицинских процессов в корпоративной среде, подтверждая её ценность как инженерного решения и как инструмента поддержки принятия решений.

ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломной работы была спроектирована, реализована и протестирована система мониторинга здоровья сотрудников в условиях корпоративной среды, ориентированная на интеграцию с современными мобильными платформами. На основании анализа полученных результатов, можно сформулировать следующие ключевые выводы.

Во-первых, разработанная концептуальная модель и архитектура системы обладают чёткой прикладной направленностью и обеспечивают однозначность интерпретации сущностей и процессов, тем самым устраняя типичную для информационных систем в области медицины полисемию и неоднозначность трактовок.

Во-вторых, использование онтологического подхода к описанию сущностей и их взаимодействий позволило сформировать целостное представление о системе мониторинга, охватывающее все уровни: от индивидуального пациента и врача до административных структур, ответственных за принятие решений и координацию ресурсов.

В-третьих, предложенная архитектура была изначально ориентирована на расширяемость и масштабируемость, а также на постепенное снижение степени участия человека в рутинных процессах принятия решений по мере внедрения более совершенных интеллектуальных алгоритмов. Такой подход обеспечивает высокую адаптивность системы к условиям будущего развития ИИ и автоматизации.

В-четвёртых, реализованная система обладает высокой прикладной ценностью в условиях эпидемиологических и кризисных ситуаций, позволяя клиникам оперативно перераспределять ресурсы, назначать приёмы, отслеживать симптоматику и формировать отчётность в реальном времени.

Наконец, открытая модульная структура архитектуры создаёт предпосылки для дальнейшего масштабирования системы за пределы медицинского направления. В частности, она может быть адаптирована к задачам управления другими сферами городской и корпоративной инфраструктуры, такими как транспорт, образование, экология и кадровая политика, что делает разработку универсальным инструментом цифровой трансформации.

Таким образом, выполненное исследование подтверждает как теоретическую состоятельность разработанных решений, так и их высокую практическую применимость в современных условиях цифровизации здравоохранения и корпоративного управления.

ЛИТЕРАТУРА

1. Hissman, S. Ultra-Large-Scale Systems: Socio-adaptive Systems / S.Hissman – CMU, 2016. – P.2-17.
2. Vissia, H. Decision making in the information society / H.Vissia, V.Krasnoproshin, A.Valvachev, A. Lan, – St. Petersburg, 2019. – 227 p.
3. Indrajit G., IoT Based Health Monitoring System / Indrajit God, Suyash Jagtap, Chirag Suryavanshi, Bhagyashri Pawar – International Journal for Research in Applied Science and Engineering Technology, 2024. – 240 p.
4. Herzberg, C. Smart Cities, Digital Nations: Building Smart Cities in Emerging Countries and Beyond / C.Gerzberg. – Roundtree Press, 2017. – 224 p.
5. Brown, G. Health Informatics: A Systems Perspective / G.Brown et al. – Health Administration Press, 2018. – 417 p.
6. Hersh, W. Health Informatics / W.Hersh, R. Hoyt – Lulu, 2018. – 486 p.
7. Hebda, T. Handbook of Informatics for Nurses & Healthcare Professionals / T.Hebda, P.Tzar. – Pearson, 2012. – 624 p.
8. 1516.1-2000 IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA). Federate Interface Specification 2000. – IEEE, 2000. – 480 p.
9. Braunstein, M. Health Informatics on FHIR: How HL7's New API is Transforming Healthcare / M.Braunstein. – Springer, 2018. – 314 p.
10. Arp, R. Building Ontologies with Basic Formal Ontology / R.Arp, B.Smith, A.Spear. – Cambridge: The MIT Press, 2015. – 248 p.
11. Zadeh, L. Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems / L.Zadeh, G.Klir, B.Yuan. – World Scientific Pub., 1996. – 840 p.
12. Kostin, A. Modeling and Simulation of Distributed Systems / A.Kostin, L.Ilushechkina – World Scientific, 2010. – 440

ПРИЛОЖЕНИЯ

Приложение А

```
@Model
class User: Identifiable {
    var id: UUID
    var login: String
    var password: String
    var surname: String
    var name: String
    var secondName: String?
    var birthday: Date?
    var sex: Sex?
    var mail: String
    var position: String?
    var listOfDiseases: [Disease]
    var companyID: UUID?
    var departmentID: UUID?

    @Transient
    var age: Int? {
        guard let birthday = birthday else { return nil }
        let calendar = Calendar.current
        let ageComponents = calendar.dateComponents([.year], from: birthday, to:
Date())
        return ageComponents.year
    }
}

import Foundation
import SwiftData

@Model
class Company: Identifiable {
    var id = UUID()
    var name: String
    var departments: [Department] = []

    init(name: String) {
        self.name = name
    }

    func addDepartment(_ department: Department) {
        departments.append(department)
    }
}
```

```

    }

    func removeDepartment(at index: Int) {
        departments.remove(at: index)
    }
}

@Model
class Department: Identifiable {
    var id = UUID()
    var name: String
    var employeeCount: Int

    init(name: String, employeeCount: Int) {
        self.name = name
        self.employeeCount = employeeCount
    }
}

import Foundation
import SwiftData

@Model
class Disease {
    var id: UUID
    var name: String
    var treatment: String
    var symptoms: [String]

    enum CodingKeys: String, CodingKey {
        case name, treatment, symptoms
    }

    // Инициализатор для декодирования, который игнорирует `id`
    init(name: String, treatment: String, symptoms: [String]) {
        self.id = UUID()
        self.name = name
        self.treatment = treatment
        self.symptoms = symptoms
    }
}

//
// Doctor.swift

```

```

import Foundation
import SwiftData

enum ScientificDegree: String, Codable {
    case doctor = "Доктор наук"
    case candidate = "Кандидат наук"
}

enum MedicalSpeciality: String, Codable {
    case ophthalmologist = "Офтальмолог"
    case surgeon = "Хирург"
    case therapist = "Терапевт"
}

enum ReceptionTime: String, Codable {
    case morning = "08:00 - 14:00"
    case evening = "14:00 - 20:00"
}

@Model
final class Doctor {
    var id: UUID
    var login: String
    var password: String
    var name: String
    var secondName: String
    var surname: String
    var mail: String
    var medicalSpeciality: MedicalSpeciality
    var scientificDegree: ScientificDegree?
    var workExperience: Int
    var roomNumber: String?
    var receptionTime: ReceptionTime
    var numberOfPatients: Int
    var listOfTreatedDiseases: [String]

    init(
        login: String,
        password: String,
        name: String,
        secondName: String,
        surname: String,
        mail: String,

```

```

        medicalSpeciality: MedicalSpeciality,
        scientificDegree: ScientificDegree? = nil,
        workExperience: Int,
        roomNumber: String? = nil,
        receptionTime: ReceptionTime,
        numberOfPatients: Int = 0,
        listOfTreatedDiseases: [String] = []
    ) {
        self.id = UUID()
        self.login = login
        self.password = PasswordHasher.hash(password)
        self.name = name
        self.secondName = secondName
        self.surname = surname
        self.mail = mail
        self.medicalSpeciality = medicalSpeciality
        self.scientificDegree = scientificDegree
        self.workExperience = workExperience
        self.roomNumber = roomNumber
        self.receptionTime = receptionTime
        self.numberOfPatients = numberOfPatients
        self.listOfTreatedDiseases = listOfTreatedDiseases
    }
}

```

// RequestModel.swift

```

import Foundation
import SwiftData

@Model
final class Request {
    var id: UUID
    var treatment: String?
    var symptoms: [String]
    var timestamp: Date
    var visitDateTime: Date?
    var userID: UUID
    var doctorID: UUID?
    var status: String?
    var notes: String?
    var beginIllnessDate: Date?
    var endIllnessDate: Date?
    var diseases: [Disease]
}

```

```

    init(treatment: String? = nil, symptoms: [String] = [], timestamp: Date = Date(),
        visitDateTime: Date? = nil, userID: UUID, doctorID: UUID? = nil, status: String? =
        "Активен", notes: String? = nil, beginIllnessDate: Date? = nil, endIllnessDate: Date?
        = nil, diseases: [Disease] = []) {
        self.id = UUID()
        self.treatment = treatment
        self.symptoms = symptoms
        self.timestamp = timestamp
        self.visitDateTime = visitDateTime
        self.userID = userID
        self.doctorID = doctorID
        self.status = status
        self.notes = notes
        self.beginIllnessDate = beginIllnessDate
        self.endIllnessDate = endIllnessDate
        self.diseases = diseases
    }
}

```

```

//
// UserModel.swift

```

```

import Foundation
import SwiftData

```

```

enum Sex: String, Codable {
    case male = "Мужской"
    case female = "Женский"
}

```

```

@Model
class User: Identifiable {
    var id: UUID
    var login: String
    var password: String
    var surname: String
    var name: String
    var secondName: String? // Отчество
    var birthday: Date? // Дата рождения
    var sex: Sex? // Пол
    var mail: String
    var position: String? // Должность
    var listOfDiseases: [Disease]
}

```

```

var companyID: UUID?
var departmentID: UUID?

@Transient
var age: Int? {
    guard let birthday = birthday else { return nil }
    let calendar = Calendar.current
    let ageComponents = calendar.dateComponents([.year], from: birthday, to:
Date())
    return ageComponents.year
}

init(login: String, password: String, surname: String, name: String, secondName:
String? = nil, birthday: Date? = nil, sex: Sex? = nil, mail: String, position: String? =
nil, listOfDiseases: [Disease], companyID: UUID? = nil, departmentID: UUID? = nil)
{
    self.id = UUID()
    self.login = login
    self.password = PasswordHasher.hash(password)
    self.surname = surname
    self.name = name
    self.secondName = secondName
    self.birthday = birthday
    self.sex = sex
    self.mail = mail
    self.position = position
    self.listOfDiseases = listOfDiseases
    self.companyID = companyID
    self.departmentID = departmentID
}
}

```

```

import Foundation
import SwiftUI
import SwiftData

class ChatViewModel: ObservableObject {
    @Published var messages: [ChatMessage] = []
    @Published var userInput: String = ""
    @Published var currentStep: ChatStep = .greeting
    @Published var appointmentInfo: AppointmentInfo
    @Environment(\.modelContext) private var modelContext
    @Query private var requestsList: [Request]
    @Query private var diseasesList: [Disease]

    enum ChatStep {
        case greeting
        case askSymptoms
        case askTime
        case confirmation
        case completed
    }

    struct AppointmentInfo {
        let userName: String
        let userSurname: String
        let userID: UUID
        var symptoms: [String] = []
        var appointmentTime: Date?
    }

    init(userName: String, userSurname: String, userID: UUID) {
        self.appointmentInfo = AppointmentInfo(userName: userName, userSurname:
userSurname, userID: userID)
    }

    func startChat() {
        messages.removeAll()
        currentStep = .greeting
        sendAIMessage(String(format: NSLocalizedString("chat_greeting", comment:
""), "\(appointmentInfo.userName) \(appointmentInfo.userSurname)"))
    }

    private func validateSymptoms(_ inputSymptoms: String) -> [String] {

```

```

    // Split input into individual symptoms
    let userSymptoms = inputSymptoms.components(separatedBy: ",").map {
    $0.trimmingCharacters(in: .whitespacesAndNewlines) }

    // Get all valid symptoms from diseases
    let validSymptoms = Set(diseasesList.flatMap { $0.symptoms })

    // Filter out invalid symptoms
    return userSymptoms.filter { symptom in
        validSymptoms.contains(symptom)
    }
}

func handleUserInput(_ input: String) {
    // Add user message
    messages.append(ChatMessage(content: input, isUser: true))

    // Process based on current step
    switch currentStep {
    case .greeting:
        let validatedSymptoms = validateSymptoms(input)

        if validatedSymptoms.isEmpty {
            sendAIMessage(NSLocalizedString("invalid_symptoms", comment: ""))
            // Stay in greeting step
        } else {
            appointmentInfo.symptoms = validatedSymptoms
            currentStep = .askTime
            let symptomsString = validatedSymptoms.joined(separator: ", ")
            sendAIMessage(String(format: NSLocalizedString("chat_ask_time",
comment: ""), symptomsString))
        }

    case .askTime:
        // Parse the date/time input and validate
        if let date = parseDateTime(input) {
            if isValidWorkingHours(date) {
                currentStep = .confirmation
                let symptomsString = appointmentInfo.symptoms.joined(separator: ", ")
                sendAIMessage(String(format:
NSLocalizedString("chat_confirmation", comment: ""),
                    "\\(appointmentInfo.userName)
\\(appointmentInfo.userSurname)",
                    symptomsString,

```

```

        formatDateTime(date)))
        appointmentInfo.appointmentTime = date
    } else {
        sendAIMessage(NSLocalizedString("working_hours_warning",
comment: ""))
    }
    } else {
        sendAIMessage(NSLocalizedString("invalid_datetime_format", comment:
""))
    }

    case .confirmation:
        if input.lowercased().contains("yes") || input.lowercased().contains("oui") ||
input.lowercased().contains("да") {
            if let visitDate = appointmentInfo.appointmentTime {
                scheduleAppointment(userID: appointmentInfo.userID, visitDate:
visitDate, symptoms: appointmentInfo.symptoms)
                currentStep = .completed
                sendAIMessage(NSLocalizedString("chat_appointment_confirmed",
comment: ""))
            }
        } else {
            currentStep = .greeting
            sendAIMessage(NSLocalizedString("chat_start_over", comment: ""))
        }

    case .completed:
        startChat()

    default:
        break
    }
}

private func sendAIMessage(_ message: String) {
    messages.append(ChatMessage(content: message, isUser: false))
}

private func scheduleAppointment(userID: UUID, visitDate: Date, symptoms:
[String]) {
    do {
        let newRequest = Request(
            treatment: nil,
            symptoms: symptoms,

```

```

        timestamp: Date(),
        visitDateTime: visitDate,
        userID: userID,
        status: "Активен",
        diseases: []
    )
    modelContext.insert(newRequest)
    try modelContext.save()
    print("Successfully saved appointment request")
} catch {
    print("Error saving appointment request: \\\(error)")
    sendAIMessage(NSLocalizedString("error_saving_appointment", comment:
""))
}
}

private func isValidWorkingHours(_ date: Date) -> Bool {
    let calendar = Calendar.current
    let hour = calendar.component(.hour, from: date)
    return hour >= 7 && hour < 18
}

private func parseDateTime(_ input: String) -> Date? {
    let dateFormatter = DateFormatter()
    dateFormatter.dateFormat = "yyyy-MM-dd HH:mm"
    return dateFormatter.date(from: input)
}

private func formatDateTime(_ date: Date) -> String {
    let formatter = DateFormatter()
    formatter.dateStyle = .medium
    formatter.timeStyle = .short
    return formatter.string(from: date)
}
}

```

```
//
// AnaliticsView.swift
// MedicineProject
//
// Created by Ksenya Chernyshova on 2024-10-02.
//

import SwiftUI
import SwiftData

struct AnaliticsView: View {
    @Query private var companiesList: [Company]
    @Query private var usersList: [User]
    @Query private var doctorsList: [Doctor]
    @Query private var requestsList: [Request]

    @Environment(\.modelContext) private var modelContext
    @Environment(\.dismiss) private var dismiss
    @Environment(\.colorScheme) private var colorScheme

    @Binding var companies: [Company]

    var body: some View {
        ZStack {
            // System background color
            Color(uiColor: .systemBackground)
                .ignoresSafeArea()

            VStack(spacing: 0) {
                // Кастомный навбар
                headerSection

                // Контент
                ScrollView {
                    VStack(spacing: 24) {
                        // Общая статистика
                        overallStatsSection

                        // Статистика по компаниям
                        companyStatsSection

                        // Медицинская статистика
                        medicalStatsSection
                    }
                }
            }
        }
    }
}
```

