МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет компьютерных наук

Кафедра программирования и информационных технологий

Медицинский ассистент с интегрированным чат-ботом

Техническое задание

09.03.04 Программная инженерия

Информационные системы и сетевые технологии

|  |  |  |
| --- | --- | --- |
| Зав. кафедрой | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | С.Д. Махортов, д-р физ.-мат. наук, доцент |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Е.С. Труфанов*,* 3 курс, д/о |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Д.С. Ушаков*,* 3 курс, д/о |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Н.С. Масалкин*,* 3 курс, д/о |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Д.А. Сакун*,* 3 курс, д/о |
| Обучающийся | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Е.В. Вологжин*,* 3 курс, д/о |
| Руководитель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | В.С. Тарасов, ст. преподаватель |
| Руководитель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Е.Д. Проскуряков, ассистент |

Воронеж 2025

содержание

[1 Терминология 5](#_Toc194059814)

[2 Общие сведения 7](#_Toc194059815)

[2.1 Полное наименование системы и название приложения 7](#_Toc194059816)

[2.2 Заказчик и разработчик 7](#_Toc194059817)

[2.3 Перечень документов, на основании которых создается система 7](#_Toc194059818)

[2.4 Плановые сроки начала и окончания работы 8](#_Toc194059819)

[3 Цели и назначение создания автоматизированной системы 9](#_Toc194059820)

[3.1 Цели создания АС 9](#_Toc194059821)

[3.2 Назначение АС 9](#_Toc194059822)

[4 Требования к автоматизированной системе 10](#_Toc194059823)

[4.1 Требования к структуре АС в целом 10](#_Toc194059824)

[4.1.1 Требования к способам и средствам обеспечения информационного взаимодействия компонентов АС 10](#_Toc194059825)

[4.1.2 Перспективы развития, модернизации АС 10](#_Toc194059826)

[4.2 Требования к функциям, выполняемым АС 10](#_Toc194059827)

[4.2.1 Вход в приложение 11](#_Toc194059828)

[4.2.2 Медицинская карта 11](#_Toc194059829)

[4.2.3 Расписание приема лекарств 12](#_Toc194059830)

[4.2.4 Медицинский чат-ассистент 12](#_Toc194059831)

[4.2.5 Экстренная кнопка (SOS) 13](#_Toc194059832)

[4.2.6 Настройки приложения 13](#_Toc194059833)

[4.3 Общие требования к оформлению и верстке страниц 14](#_Toc194059834)

[4.3.1 Общие элементы вёрстки 15](#_Toc194059835)

[4.3.2 Экран входа 15](#_Toc194059836)

[4.3.3 Экран регистрации 15](#_Toc194059837)

[4.3.4 Экран заполнения медицинской карты 16](#_Toc194059838)

[4.3.5 Экран чат-ассистента 16](#_Toc194059839)

[4.3.6 Экран просмотра медицинской карты 17](#_Toc194059840)

[4.3.7 Экран редактирования медицинской карты 17](#_Toc194059841)

[4.3.8 Экран расписания приёма лекарственных препаратов 17](#_Toc194059842)

[4.3.9 Экран редактирования расписания приёма лекарственных препаратов 17](#_Toc194059843)

[4.3.10 Экран настроек 17](#_Toc194059844)

[4.3.11 Экран изменения пароля 17](#_Toc194059845)

[4.3.12 Экстренная кнопка (SOS) 18](#_Toc194059846)

[4.4 Требования к видам обеспечения АС 18](#_Toc194059847)

[4.4.1 Лингвистическое обеспечение АС 18](#_Toc194059848)

[4.4.2 Программное обеспечение АС 18](#_Toc194059849)

[4.5 Общие технические требования к АС 20](#_Toc194059850)

[4.5.1 Требования к численности и квалификации персонала и пользователей АС 20](#_Toc194059851)

[4.5.2 Требования по безопасности 20](#_Toc194059852)

[4.5.3 Дополнительные требования 20](#_Toc194059853)

[4.6 Функциональные и нефункциональные требования 21](#_Toc194059854)

[4.6.1 Функциональные требования 21](#_Toc194059855)

[4.6.2 Нефункциональные требования 21](#_Toc194059856)

[5 Состав и содержание работ по созданию автоматизированной системы 23](#_Toc194059857)

[6 Порядок разработки автоматизированной системы 24](#_Toc194059858)

[7 Порядок контроля и приемки автоматизированной системы 25](#_Toc194059859)

[8 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие 26](#_Toc194059860)

[8.1 Общие положения 26](#_Toc194059861)

[8.2 Подготовка инфраструктуры 26](#_Toc194059862)

[8.2.1 Серверная инфраструктура 26](#_Toc194059863)

[8.2.2 Клиентская часть 26](#_Toc194059864)

[8.2.3 Интеграция с внешними сервисами 26](#_Toc194059865)

[8.3 Подготовка данных 27](#_Toc194059866)

[8.3.1 Перенос данных (при наличии) 27](#_Toc194059867)

[8.3.2 Начальное наполнение системы 27](#_Toc194059868)

[8.4 Обучение пользователей 27](#_Toc194059869)

[8.5 Тестирование перед вводом в эксплуатацию 27](#_Toc194059870)

[8.5.1 Функциональное тестирование 27](#_Toc194059871)

[8.5.2 Нагрузочное тестирование 28](#_Toc194059872)

[8.5.3 Тестирование безопасности 28](#_Toc194059873)

[8.6 План ввода системы в действие 28](#_Toc194059874)

[8.6.1 Пилотное внедрение 28](#_Toc194059875)

[8.6.2 Полномасштабный запуск 28](#_Toc194059876)

[8.6.3 Поддержка и сопровождение 28](#_Toc194059877)

[8.7 Документирование процесса внедрения 28](#_Toc194059878)

[8.8 Ответственные лица 29](#_Toc194059879)

[9 Требования к документированию 30](#_Toc194059880)

[9.1 Перечень подлежащих разработке документов 30](#_Toc194059881)

[9.2 Вид представления и количество документов 30](#_Toc194059882)

[10 Источники разработки 31](#_Toc194059883)

[10.1 Системы-аналоги 31](#_Toc194059884)

[10.1.1 Your.MD 31](#_Toc194059885)

[10.1.2 Medisafe 32](#_Toc194059886)

[ПРИЛОЖЕНИЕ А Диаграмма прецедентов 33](#_Toc194059887)

[ПРИЛОЖЕНИЕ Б Диаграмма классов 34](#_Toc194059888)

1. Терминология

|  |  |
| --- | --- |
| Термин | Определение |
| Back-end | часть программного обеспечения, отвечающая за обработку данных и взаимодействие с сервером. |
| Dart | язык программирования, созданный компанией Google. |
| Flutter | комплект средств разработки и фреймворк с открытым исходным кодом для создания мобильных приложений под Android и iOS, веб-приложений. |
| Front-end | часть программного обеспечения, отвечающая за визуальное представление данных и взаимодействие с пользователем. |
| MVP | продукт, обладающий минимальными, но достаточными для удовлетворения первых потребителей функциями. |
| REST API | архитектурный стиль веб-служб, который использует протокол HTTP для передачи данных между клиентом и сервером. |
| Spring | фреймворк для разработки приложений на языке Java. |
| Автоматизированная система (АС) | программа или набор программ, предназначенных для выполнения задач без прямого участия человека. |
| Авторизация | процесс предоставления пользователю или группе пользователей определенных разрешений, прав доступа и привилегий в компьютерной системе. |
| Аккаунт | персональная учетная запись пользователя, которая позволяет ему получить доступ к определенным ресурсам или функциям в рамках системы или сервиса. |
| Аутентификация | процесс проверки подлинности пользователя, предоставляющего учетные данные (логин и пароль) для доступа к системе или сервису. |
| База данных (БД) | организованное совокупность данных, обычно хранящихся и обрабатываемых с использованием компьютерных систем. |
| Библиотека | набор функций или классов, предназначенных для решения определенной задачи или облегчения разработки программного обеспечения. |
| Искусственный интеллект (ИИ) | это комплекс технологических решений, позволяющий имитировать когнитивные функции человека (включая самообучение, поиск решений без заранее заданного алгоритма и достижение инсайта) |
| Метрика | числовое значение, которое используется для измерения или оценки определенного параметра или процесса. |
| Мониторинг | процесс систематического наблюдения за состоянием системы, процессов или ресурсов с целью выявления и контроля изменений, а также оперативного реагирования на возможные проблемы или улучшения. |
| Отладка | процесс поиска и исправления ошибок в программном коде. |
| Программное обеспечение (ПО) | совокупность программных инструкций, данные и документации, предназначенных для работы компьютерной системы или устройства. |
| Сервер | компьютер или программа, предоставляющая ресурсы или услуги другим компьютерам или программам. |
| Система управления базами данных (СУБД) | программное обеспечение для управления базами данных. |
| Фреймворк | набор библиотек и инструментов, облегчающих разработку программного обеспечения. |

1. Общие сведения
   1. Полное наименование системы и название приложения

Полное наименование системы: «Мобильное приложение медицинского ассистента PocketHealth».

Условное обозначение приложения: «PocketHealth».

* 1. Заказчик и разработчик

Заказчик: Старший преподаватель Тарасов Вячеслав Сергеевич, Воронежский государственный университет, факультет компьютерных наук, кафедра программирования и информационных технологий.

Представитель заказчика: Ассистент Проскуряков Егор Дмитриевич, Воронежский государственный университет, факультет компьютерных наук, кафедра программирования и информационных технологий.

«4» команда группы «8». Состав команды разработчика:

* + - Сакун Дарья Александровна;
    - Труфанов Егор Сергеевич;
    - Ушаков Денис Сергеевич;
    - Масалкин Никита Сергеевич;
    - Вологжин Егор Владимирович.
  1. Перечень документов, на основании которых создается система
     + Закон РФ от 07.02.1992 N 2300-1 (ред. От 11.06.2021) «О защите прав потребителей»;
     + федеральный закон «О персональных данных» от 27.07.2006 N 152-ФЗ.
  2. Плановые сроки начала и окончания работы

Работы по созданию системы должны начаться 24.02.2025 и завершиться 10.06.2025.

1. Цели и назначение создания автоматизированной системы
   1. Цели создания АС

Целями создания системы являются:

* + - позволить пользователям узнать возможные заболевания по имеющимся у них симптомам на основе ИИ;
    - помочь пользователям не забывать о приеме лекарств;
    - сделать хранение медицинских данных пользователя более удобным.
  1. Назначение АС

Система предназначена для:

* + - получения информации о возможных заболеваниях на основе указанных симптомов;
    - установки расписание приема лекарственных препаратов;
    - формирования и редактирования интерактивной медицинской карты;
    - осуществления редактирования данных своего аккаунта после авторизации или регистрации в системе.

1. Требования к автоматизированной системе

Автоматизированная система должна соответствовать ряду требований, обеспечивающих ее надежность, масштабируемость и удобство использования.

* 1. Требования к структуре АС в целом

Структура автоматизированной системы должна обеспечивать устойчивую и эффективную работу всех ее компонентов. Важно предусмотреть возможность интеграции новых функций, а также обеспечить гибкость и масштабируемость для будущих модернизаций.

* + 1. Требования к способам и средствам обеспечения информационного взаимодействия компонентов АС

Система должна иметь архитектуру, соответствующую модели клиент-серверного взаимодействия на основе REST API для обмена данными между клиентом и сервером.

* + 1. Перспективы развития, модернизации АС

В долгосрочной перспективе возможны следующие направления модернизации системы:

* + - интеграция возможности записи к врачам в платные поликлиники;
    - база данных лекарственных препаратов;
    - введение подписки на неограниченное количество обращений к чат-ассистенту.
  1. Требования к функциям, выполняемым АС

В системе пользователь будет иметь одну из ролей:

* + - неавторизованный пользователь;
    - авторизованный пользователь.
    1. Вход в приложение

Система должна предоставлять пользователю возможность:

* + - зарегистрировать профиль;
    - войти по почте и паролю;
    - восстановить пароль;
    - заполнить данные медицинской карты во время регистрации.
    1. Медицинская карта

Для авторизованного пользователя система должна предоставлять доступ к личной медицинской карты:

* + - просмотр медицинской карты;
    - редактирование и удаление медицинской карты.

При создании и редактировании медицинской карты система должна позволять учитывать следующие сведения:

* + - ФИО;
    - рост;
    - вес(кг);
    - группа крови;
    - аллергии;
    - заболевания.

Как результат, эта информация должна сохраниться в системе и быть доступной для авторизованного пользователя.

* + 1. Расписание приема лекарств

Каждому пользователю система должна предоставлять доступ к расписанию приема лекарственных препаратов:

* + - просмотр расписания;
    - редактирование и удаление расписания.

При создании и редактировании расписания лекарств система должна позволять учитывать следующие сведения:

* + - название лекарственного препарата;
    - день приема;
    - время приема;
    - настройка уведомлений.

Как результат, эта информация должна сохраниться в системе и быть доступной для каждого пользователя приложения.

* + 1. Медицинский чат-ассистент

Для авторизованного пользователя система должна предоставлять доступ к медицинскому ассистенту в виде чат-бота, в котором реализован следующий функционал:

* + - просмотр предыдущих сообщений;
    - определение возможного заболевания, по введенным пользователем симптомам.

При получении симптома от пользователя система должна предоставлять список возможных заболеваний.

Выбранное заболевание должно включать в себя следующие пункты:

* + - описание болезни;
    - виды заболевания;
    - причины возникновения заболевания.

Как результат, эта информация должна сохраниться в системе и быть доступной для всех зарегистрированных пользователей.

* + 1. Экстренная кнопка (SOS)

Система должна предоставлять авторизованному пользователю возможность воспользоваться виджетом, открывающим медицинскую карту.

Как результат, эту информацию можно использовать при обращении в скорую медицинскую помощь.

* + 1. Настройки приложения

Система должна предоставлять авторизованному пользователю возможность воспользоваться настройками, включающие в себя:

* + - изменение пароля;
    - выход из аккаунта;
    - включение/выключение push-уведомлений.

При изменении пароля система должна учитывать:

* + - старый пароль;
    - новый пароль;
    - подтверждение нового пароля.
  1. Общие требования к оформлению и верстке страниц

Экраны мобильного приложения должны быть оформлены в едином стиле с использованием ограниченного набора шрифтов.

Необходимо корректное и одинаковое отображение экранов мобильного приложения на устройствах с операционной системой Android 11 и выше.

Рисунок 1 демонстрирует прототип цветового оформления дизайна мобильного приложения:



Рисунок 1 — Цветовое оформление дизайна мобильного приложения

Рисунок 2 демонстрирует прототипы основных типов экранов мобильного приложения:

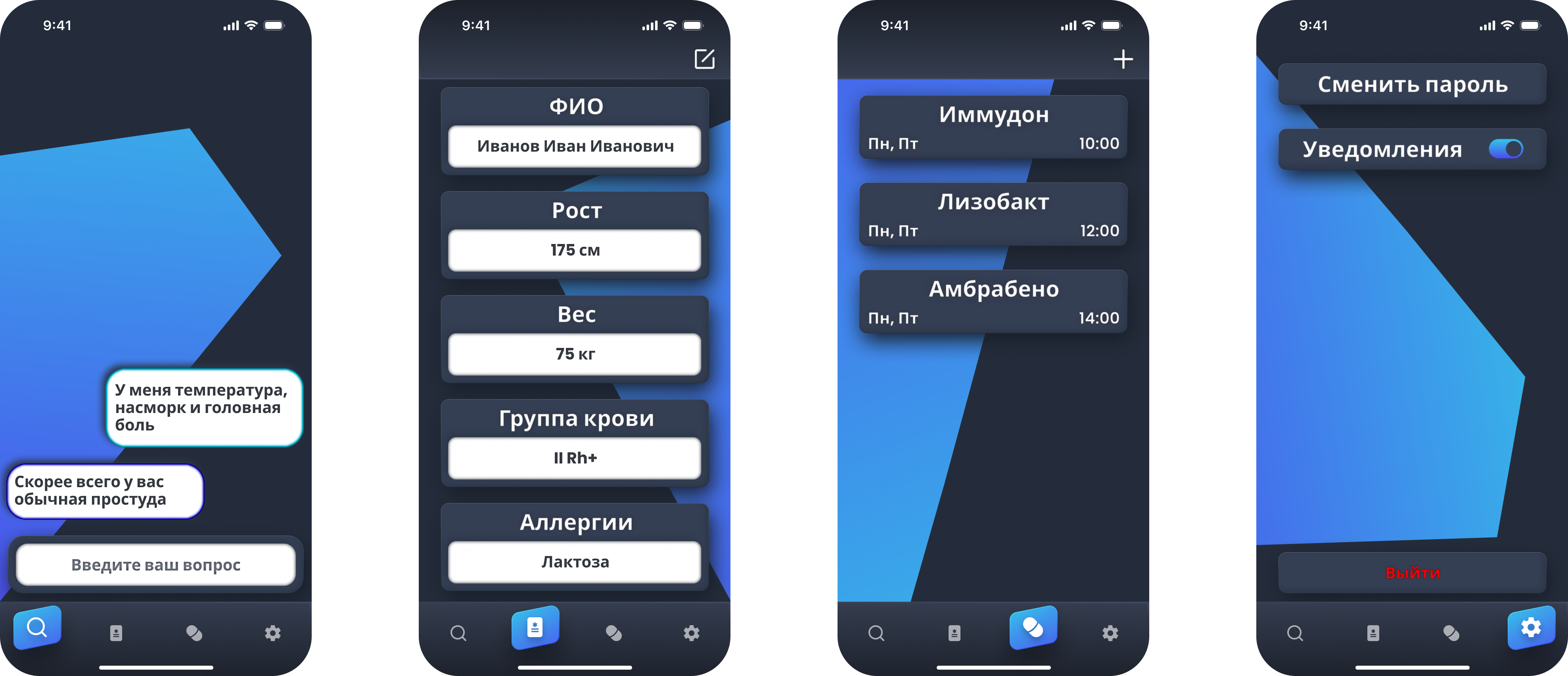


Рисунок 2 — Прототипы экранов с чатом, медицинской картой, расписанием приёма лекарственных препаратов и настройками

* + 1. Общие элементы вёрстки

TabBar содержит кнопки перехода на экран с чат-ассистентом, экран для просмотра и редактирования медицинской карты, экран расписания приёма лекарственных препаратов и экран настроек приложения.

* + 1. Экран входа

Экран входа открывается при запуске приложения. После экран должен содержать форму для ввода логина(почта) и пароля, кнопку входа, кнопка «Забыли пароль?» и кнопку регистрации.

После успешного входа должен появиться медицинский чат-ассистент.

* + 1. Экран регистрации

Экран регистрации открывается при нажатии на кнопку «Зарегистрироваться», содержит форму для ввода логина(почта), пароля и подтверждения пароля.

После успешного ввода данных при регистрации должна открываться форма для заполнения медицинской карты, которая также является частью регистрации.

* + 1. Экран заполнения медицинской карты

Данный экран (см. рисунок 3) содержит поля для ввода основной информации для медицинской карты.

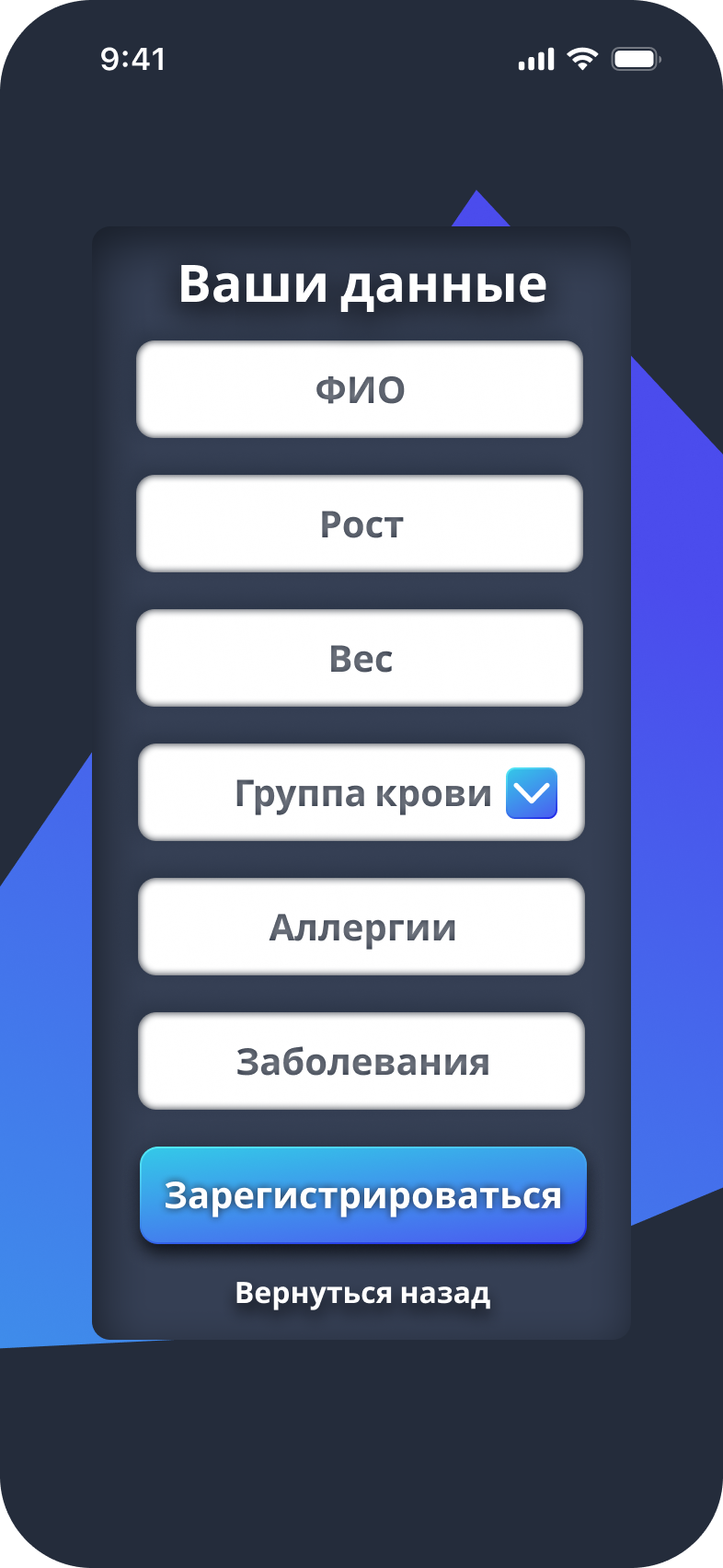


Рисунок 3 — Экран с заполнением медицинской карты

* + 1. Экран чат-ассистента

Данный экран предоставляет доступ к чат-ассистенту, в котором система принимает сообщения от пользователя и выводит ответы от обученного искусственного интеллекта.

* + 1. Экран просмотра медицинской карты

Данный экран имеет форму для просмотра данных медицинской карты, описанных в пункте 4.2.2, а также кнопку для редактирования медицинской карты.

* + 1. Экран редактирования медицинской карты

Данный экран содержит форму для ввода, описанную в пункте 4.2.2, которые позволяют редактировать пользователю данные медицинской карты, а также кнопки подтверждения изменений и отмены изменений.

* + 1. Экран расписания приёма лекарственных препаратов

Данный экран содержит форму для просмотра расписания, описанные в пункте 4.2.3, а также кнопки создания, удаления и редактирования записей.

* + 1. Экран редактирования расписания приёма лекарственных препаратов

Данный экран содержит форму, позволяющую редактировать данные расписания, описанные в пункте 4.2.3, а также кнопки подтверждения изменений и отмены изменений.

* + 1. Экран настроек

Данный экран содержит форму для изменения настроек пользователя, включающая обновление пароля, управление push-уведомлениями с помощью переключателя, а также кнопку выхода из аккаунта.

* + 1. Экран изменения пароля

Данный экран содержит форму, описанную в пункте 4.2.6, которая должна учитывать:

* + - старый пароль;
    - новый пароль;
    - подтверждение нового пароля.
    1. Экстренная кнопка (SOS)

Виджет (см. рисунок 4), расположенный на главном экране устройства, по нажатию на который пользователь может сразу попасть на страницу просмотра медицинской карты.



Рисунок 4 — Виджет для экстренной кнопки

* 1. Требования к видам обеспечения АС

Для полноценного функционирования автоматизированной системы необходимо учитывать требования к различным видам обеспечения, включая лингвистическое, программное и инструментальное обеспечение. Эти аспекты определяют доступность системы для пользователей, удобство ее эксплуатации и возможности для дальнейшего развития.

* + 1. Лингвистическое обеспечение АС

В системе для интерфейса пользователя должен использоваться русский язык.

* + 1. Программное обеспечение АС

Приложение должно иметь архитектуру, соответствующую модели Клиент-Серверного взаимодействия на основе REST API.

Для реализации серверной части будут использоваться следующие средства:

* + - язык программирования Java версии 17;
    - фреймворк Spring Boot версии 3.4.3;
    - СУБД PostgreSQL версии 17.

Для реализации клиентской части мобильного приложения и сервисного веб-приложения будут использоваться следующие средства:

* + - язык программирования Dart версии 3.7.0;
    - Flutter SDK версии 3.29.0.

Инструменты для введения документации:

* + - YouTrack — это средство для управления проектами и задачами, которое обеспечивает удобный интерфейс, гибкую настройку и возможность отслеживать прогресс работы над проектом;
    - Miro — это инструмент для совместной работы и визуализации идей, который позволяет создавать диаграммы, макеты, схемы и другие элементы проекта;
    - PlantUML — это инструмент с открытым исходным кодом, позволяющий пользователям создавать диаграммы на обычном текстовом языке;
    - Figma — инструмент для дизайна интерфейсов, который обеспечивает возможность создания прототипов, макетов и дизайнов веб-приложений.

Дополнительные инструменты:

* + - Git — распределенная система управления версиями, которая обеспечивает контроль изменений в программном коде, возможность ветвления и слияния программного кода;
    - GitHub — это платформа для хостинга проектов на базе Git, которая обеспечивает возможность хранения программного кода, управления задачами, рецензирования программного кода и совместной работы над проектами;
    - платформа Docker — открытая платформа для разработки, доставки и эксплуатации приложений. Позволяет создавать среды разработки и развертывать приложения с минимальными затратами на конфигурацию и совместимость.
  1. Общие технические требования к АС

Помимо архитектурных и функциональных требований, автоматизированная система должна соответствовать ряду общих технических требований. Эти требования направлены на обеспечение безопасности, удобства эксплуатации и оптимального использования ресурсов.

* + 1. Требования к численности и квалификации персонала и пользователей АС

Приложение основано на обученном искусственном интеллекте, поэтому для его работы не требуется постоянное участие медицинского персонала. Рекомендуется привлечение специалиста в области медицины для валидации и обновления алгоритмов на основе актуальных медицинских данных.

* + 1. Требования по безопасности
* обмен данных между клиентом и сервером должен осуществлять по протоколу HTTPS;
* пароли пользователей должны хранится в базе данных в хешированном виде;
* для хеширования должен использоваться BCrypt Password Encoder и JWT Token.
  + 1. Дополнительные требования

При первом запуске мобильного приложения пользователь должен иметь возможность пропустить этап регистрации, тем самым получить доступ только к функционалу по составлению расписания приёма лекарственных препаратов.

* 1. Функциональные и нефункциональные требования
     1. Функциональные требования

Полный перечень функций системы приведен в п. 4.2 «Требования к функциям, выполняемым АС». Ключевые возможности:

* + - управление учетными записями (регистрация, вход, восстановление пароля – п. 4.2.1);
    - работа с медицинской картой (просмотр, редактирование – п. 4.2.2);
    - управление расписанием приема лекарств (п. 4.2.3);
    - взаимодействие с чат-ассистентом (п. 4.2.4);
    - использование экстренной кнопки SOS (п. 4.2.5);
    - настройки приложения (п. 4.2.6).
    1. Нефункциональные требования

**Производительность и надежность**:

* + - время отклика сервера: ≤500 мс (на основе архитектуры REST API, п. 4.1.1);
    - поддержка ≥1000 одновременных пользователей (п. 4.5.1);
    - доступность 99.9% (требования к серверной инфраструктуре, п. 4.4.2).

**Безопасность**:

* + - обязательное использование HTTPS (п. 4.5.2);
    - хеширование паролей (BCrypt + JWT, п. 4.5.2);
    - защита персональных данных в соответствии с ФЗ-152 (п. 2.3).

**Пользовательский интерфейс**:

* + - единый стиль интерфейса (цветовая схема, шрифты – п. 4.3, рис. 1–2);
    - поддержка Android 11+ (п. 4.3).

**Масштабируемость**:

* + - возможность интеграции новых модулей (планы развития – п. 4.1.2);
    - поддержка роста базы данных (PostgreSQL, п. 4.4.2).

1. Состав и содержание работ по созданию автоматизированной системы

Состав и содержание работ по созданию мобильного приложения включают в себя следующие этапы:

* сбор необходимой информации, постановка целей, задач системы, которые в будущем должны быть реализованы;
* анализ предметной области, анализ конкурентов и построение структуры требований, ведущих к решению поставленных задач и целей;
* построение модели программы, описание спецификаций данных, определение связей между сущностями, разработка модели БД;
* разработка рабочего проекта, состоящего из написания программного кода, вёрстки страниц, отладки и корректировки кода программы;
* проведение тестирования программного обеспечения, развёртывание рабочей системы на хостинге и сбор аналитических данных.

1. Порядок разработки автоматизированной системы

Таблица 1 — Перечень документов, предъявляемых по окончании соответствующих этапов работ

|  |  |  |
| --- | --- | --- |
| Этап работы | Срок окончания этапа | Предъявляемые документы |
| 1 аттестация | 30 марта 2025 | Техническое задание |
| 2 аттестация | 30 апреля 2025 | Промежуточный курсовой проект |
| 3 аттестация | 6 июня 2025 | Готовый курсовой проект |

1. Порядок контроля и приемки автоматизированной системы

Предварительные отчёты по работе будут проводиться во время рубежных аттестаций:

* + - 1 аттестация (30.03.2025) — предоставлены ссылки и доступы к проекту на GitHub, YouTrack, Miro и Figma, предоставлены промежуточные результаты по курсовому проекту и готовое техническое задание, создан прототип дизайна;
    - 2 аттестация (30.04.2025) — реализовано MVP, реализована верстка и дизайн приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка программного кода, проведено тестирование по работе системы;
    - 3 аттестация (06.06.2025) — разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

1. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие
   1. Общие положения

Перед вводом автоматизированной системы (АС) «PocketHealth» в эксплуатацию необходимо выполнить комплекс работ, направленных на подготовку инфраструктуры, пользователей и данных к корректному функционированию системы. Данный раздел определяет порядок и содержание этих работ с учетом требований, изложенных в предыдущих разделах технического задания.

* 1. Подготовка инфраструктуры

Для успешного развертывания системы необходимо обеспечить выполнение следующих условий:

* + 1. Серверная инфраструктура
    - развертывание серверного окружения с поддержкой Java 17, Spring Boot 3.4.3 и PostgreSQL 17;
    - настройка HTTPS для безопасного обмена данными между клиентом и сервером (п. 4.5.2);
    - обеспечение резервного копирования базы данных и отказоустойчивости.
    1. Клиентская часть
    - подготовка мобильных устройств с операционной системой Android 11 и выше для тестирования и эксплуатации (п. 4.3);
    - настройка push-уведомлений для напоминаний о приеме лекарств (п. 4.2.6).
    1. Интеграция с внешними сервисами
    - проверка работоспособности REST API для взаимодействия между клиентской и серверной частями (п. 4.1.1);
    - настройка Docker-контейнеров для развертывания серверных компонентов (п. 4.4.2).
  1. Подготовка данных
     1. Перенос данных (при наличии)
     + если система внедряется взамен существующего решения, необходимо обеспечить миграцию данных пользователей (медицинских карт, расписаний приема лекарств) в новую базу данных;
     + проверка целостности и корректности данных после миграции.
     1. Начальное наполнение системы
     + загрузка справочной информации о заболеваниях и симптомах для работы чат-ассистента (п. 4.2.4);
     + тестовые данные для отладки функционала (например, примеры медицинских карт и расписаний).
  2. Обучение пользователей
     + разработка инструкций по регистрации, работе с медицинской картой, чат-ассистентом и расписанием приема лекарств (п. 4.2);
     + проведение обучающих сессий или создание видеоуроков для пользователей.
  3. Тестирование перед вводом в эксплуатацию
     1. Функциональное тестирование
     + проверка всех сценариев использования: регистрация, вход, работа с медицинской картой, чат-ботом, расписанием и настройками (пп. 4.2.1–4.2.6);
     + тестирование экстренной кнопки (SOS) и виджета (п. 4.2.5, п. 4.3.12).
     1. Нагрузочное тестирование

Проверка стабильности системы при одновременной работе множества пользователей.

* + 1. Тестирование безопасности

Проверка корректности хеширования паролей (BCrypt, JWT Token) и защиты данных (п. 4.5.2).

* 1. План ввода системы в действие
     1. Пилотное внедрение
     + ограниченный запуск для тестовой группы пользователей (например, студентов или сотрудников университета);
     + сбор обратной связи и устранение выявленных проблем.
     1. Полномасштабный запуск
     + развертывание системы для всех пользователей;
     + мониторинг работы в первые недели эксплуатации.
     1. Поддержка и сопровождение
     + назначение ответственных за техническую поддержку пользователей;
     + планирование обновлений и доработок (п. 4.1.2).
  2. Документирование процесса внедрения
     + составление отчета о проведенных работах, включая результаты тестирования и список выполненных задач;
     + обновление технической документации (инструкции, руководства пользователя).
  3. Ответственные лица
     + заказчик (Тарасов В.С., Проскуряков Е.Д.) – контроль за соблюдением требований;
     + разработчики (команда «4») – выполнение работ по подготовке и вводу системы.

1. Требования к документированию
   1. Перечень подлежащих разработке документов

* техническое задание;
* курсовой проект.
  1. Вид представления и количество документов

Документы должны быть представлены в электронном виде и опубликованы на сайте github.com в репозитории команды разработчика, а также в печатном виде.

1. Источники разработки
   1. Системы-аналоги

В ходе исследования рынка приложений медицинских чат-ассистентов было выявлено 2 прямых конкурента. Таблица 2 содержит результаты проведённого конкурентного исследования.

Таблица 2 — Результаты конкурентного исследования

|  |  |  |  |
| --- | --- | --- | --- |
| Характеристика | Your.MD | Medisafe | PocketHealth |
| Наличие искусственного интеллекта | + | - | + |
| Диагностика симптомов | + | - | + |
| Напоминания о приеме препаратов | + | + | + |
| Возможность заполнения собственной медицинской карты | - | - | + |

* + 1. Your.MD

Your.MD — это приложение, которое помогает пользователям управлять своим здоровьем с помощью искусственного интеллекта. Система включает диагностику симптомов, напоминания о приеме лекарств, онлайн-запись к врачу и персональные рекомендации. Пользователь может быстро получить анализ своего состояния, советы по лечению и профилактике, а также удобный доступ к медицинским услугам.

* + 1. Medisafe

Medisafe — это приложение, которое помогает пользователям не забывать о приеме лекарств. Оно отправляет напоминания, отслеживает дозировки и позволяет делиться расписанием с близкими или врачами. Приложение также анализирует совместимость препаратов и предоставляет персонализированные советы по их приему.

#### Диаграмма прецедентов

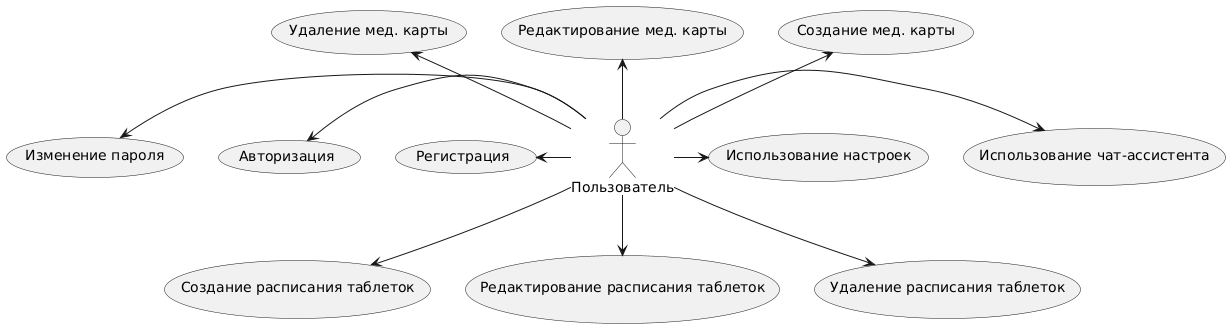


Рисунок А.1 — Диаграмма прецедентов

#### Диаграмма классов

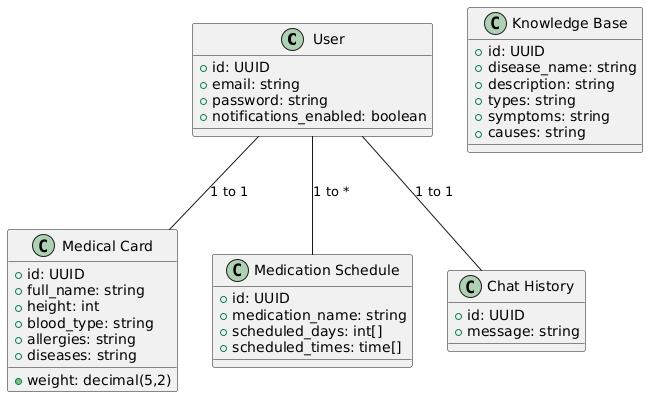


Рисунок Б.1 — Диаграмма классов