



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«МИРЭА – Российский технологический университет»**  
**РТУ МИРЭА**

---

**Институт информационных технологий (ИИТ)**  
**Кафедра математического обеспечения и стандартизации информационных**  
**технологий (МОСИТ)**

**КУРСОВАЯ РАБОТА**

по дисциплине      «Проектирование и разработка мобильных приложений»

**Тема курсовой работы: «АРМ регистратуры в поликлинике»**

Студент группы ИКБО-62-23      Кокшарова Анна Алексеевна

*кокшарова*  
(подпись)

Руководитель  
курсовой работы

Доцент Синицын И.В.

*Синицын*  
(подпись)

Работа представлена к защите      «11» 06 2025 г.

«11» 06 2025 г.

Допущен к защите

Москва 2025 г.

*Судаков*



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«МИРЭА – Российский технологический университет»**  
**РТУ МИРЭА**

**Институт информационных технологий (ИИТ)**  
**Кафедра математического обеспечения и стандартизации информационных**  
**технологий (МОСИТ)**

Утверждаю  
И.О.Заведующего кафедрой МОСИТ  
Головин С.А.  
«17» февраля 2025 г.

**ЗАДАНИЕ**  
**на выполнение курсовой работы**  
по дисциплине «Проектирование и разработка мобильных приложений»

Студент Кокшарова Анна Алексеевна

Группа ИКБО-62-23

**Тема «АРМ регистратуры в поликлинике»**

**Исходные данные:** Разрабатываемый прототип мобильного приложения должен предоставлять возможности полностью интерактивной системы с понятным дружественным UI и обеспечивать необходимую функциональность, которая формируется в зависимости от заданной темы и предметной области изучаемых вопросов.

**Перечень вопросов, подлежащих разработке, и обязательного графического материала:**  
Установка и настройка мобильной ОС с применением виртуальных сред. Установка и настройка IDE для мобильной разработки. Установка и настройка эмуляторов мобильного девайса.

Анализ предметной области разрабатываемой программной системы, сбор и анализ требований, составление ТЗ согласно ГОСТ 19.201-78. Реализация ролевой модели безопасности.

Изучение жизненного цикла мобильных программ и их компонентов, а также создание мобильного программного комплекса с применением языков программирования высокого уровня согласно теме курсовой работы. Реализация в создаваемом программном комплексе визуальных элементов, сервисов, методов хранения данных.

Тестирование и отладка кода созданного программного продукта, контрольные примеры работы. Возможно портирование написанной программной системы на внешних хостах в сети Интернет. Отчет по курсовой работе в виде пояснительной записки.

**Срок представления к защите курсовой работы:**

до «30» мая 2025 г.

Синицын И.В.

Подпись руководителя

(ФИО руководителя)

«17» февраля 2025 г.

**Задание на курсовую работу выдал**

Кокшарова А.А.

(ФИО обучающегося)

«17» февраля 2025 г.

**Задание на курсовую работу получил**

код -

Подпись обучающегося

(ФИО обучающегося)

«17» февраля 2025 г.

## **Оглавление**

|   |    |
|---|----|
| Введение.....   | 2  |
| Цель курсового проектирования.....                      | 2  |
| Задачи .....  | 2  |
| Предметная область и актуальность .....                 | 2  |
| Актуальность .....                                      | 3  |
| Описание обильного приложения .....                     | 3  |
| Анализ и исследование .....                             | 4  |
| Анализ результатов исследования.....                    | 4  |
| Аналоги.....  | 4  |
| Требования к реализации проекта .....                   | 6  |
| Техническое задание на разработку АРМ регистратуры..... | 6  |
| Реализация проекта .....                                | 10 |
| Средства разработки .....                               | 10 |
| Пользователи и роли .....                               | 10 |
| Сценарий использования.....                             | 11 |
| Демонстрация работы приложения .....                    | 12 |
| Вывод.....  | 22 |
| Список используемых источников.....                     | 24 |

## **Введение**

Автоматизированное рабочее место (АРМ) регистратуры в поликлинике — это программное обеспечение, предназначенное для упрощения и ускорения процессов записи пациентов на приём, управления их данными и взаимодействия с медицинским персоналом. Внедрение таких систем позволяет сократить время обслуживания пациентов, минимизировать ошибки при записи и улучшить общую эффективность работы медицинского учреждения. Разработка АРМ регистратуры, обеспечит удобный интерфейс для работы администраторов, врачей и пациентов, а также интеграцию с базой данных для хранения информации.

## **Цель курсового проектирования**

Разработать АРМ регистратуры поликлиники для автоматизации процессов записи пациентов, управления их данными и формирования отчётности.

## **Задачи**

- Провести анализ предметной области и существующих решений.
- Определить функциональные и нефункциональные требования к системе.
- Разработать пользовательский интерфейс для администраторов и пациентов.
- Реализовать базу данных для хранения информации о пациентах, врачах и записях на приём.
- Обеспечить безопасность данных и разграничение прав доступа.
- Протестировать систему и подготовить документацию.

## **Предметная область и актуальность**

АРМ регистратуры включает следующие функции:

- Запись пациентов на приём к врачу.
- Управление расписанием врачей.
- Хранение и обработка данных пациентов.

- Интеграция с электронной медицинской картой (при наличии).

## **Актуальность**

Автоматизация процессов в поликлиниках позволяет сократить очереди и время ожидания пациентов. Также учет записей позволяет избежать ошибок. Автоматизация упрощает работу администратора и повышает прозрачность и контроль за расписанием врачей

## **Описание обильного приложения**

В данной курсовой работе разработано мобильное приложение, которое позволяет пользователям удобно записываться на прием к определенным врачам. Приложение предоставляет доступ к электронной мед карте, личным данным.

Чтобы получить доступ к возможности записываться, пользователю необходимо создать аккаунт.

## **Анализ и исследование**

### **Анализ результатов исследования**

В результате анализа текущих процессов работы регистратуры в поликлиниках можно заключить следующее:

Существующие способы записи пациентов на приём зачастую неудобны и требуют значительного времени. Большинство пациентов сталкиваются с трудностями при выборе врача, времени записи и получения информации о доступных слотах.

актуально разработать АРМ регистратуры , которое будет:

- Удобным в использовании
- Поддерживать мобильный доступ
- Объединять все функции: Запись к врачу, архив посещений, медкарта, рекомендации от врачей

### **Аналоги**

Был проведён поиск и анализ существующих решений, связанных с записью к врачу через мобильное приложение. На основе анализа были выбраны ключевые аналоги:

#### **1. «Госуслуги» (портал и мобильное приложение)**

Данное приложение позволяет записаться к врачу через ЕГИС.

#### **Основные возможности:**

- Выбор врача из доступных
- Запись на приём
- Получение направления
- Просмотр истории приёма

#### **Отличия от разрабатываемого АРМ:**

- Неудобный интерфейс
- Нет возможности работать с медкартой напрямую
- Требуется авторизация через госуслуги
- Ограниченнная функциональность на мобильном устройстве

## **2. Приложение «Поликлиника Online»**

Приложение предоставляет возможность записаться к врачу, получить направление и посмотреть историю посещений.

### **Основные особенности:**

- Возможность онлайн-записи
- Информация о врачах и свободных слотах
- Просмотр истории посещений

### **Отличия от разрабатываемого АРМ:**

- Требует регистрации в системе
- Нет функции медкарты с диагнозами и рекомендациями
- Не подходит для внутреннего использования в поликлинике
- Нет роли администратора

## **3. Приложение «ЕГИС.Здрав» (аналог в AppStore)**

Это государственная система электронной регистратуры.

### **Основные функции:**

- Онлайн-запись к врачу
- Просмотр расписания
- Получение номерка в очередь

### **Отличия от разрабатываемого АРМ:**

- Нет разделения ролей (администратор, врач, пациент)
- Нет хранения данных пациента внутри самого приложения
- Нет интеграции с медкартой
- Нет удобного интерфейса для врачей

Таким образом, предлагаемое АРМ регистратуры в поликлинике позволяет объединить все функции в одном месте, будет иметь простой и понятный интерфейс, поддержит разные роли доступа, позволит записаться к врачу, просмотреть медкарту, сохранить жалобы, назначения и рекомендации избежит распространённых проблем: реклама, подписка, региональные ограничения

## **Требования к реализации проекта**

Мобильное приложение, предназначенное для работы с записью пациентов и медкартами, должно обеспечивать удобный и надёжный доступ к информации о пациентах, врачах, записях и рекомендациях.

### **Основные экраны приложения:**

- Главный экран (регистратура)**

1. Список пациентов, записанных на приём
2. Кнопка "Начать приём"
3. Фильтр по дате и времени
4. Информация о текущем враче

- Медкарта пациента**

1. Группа крови, рост, вес
2. Жалобы пациента
3. Рекомендации врача
4. Заметки и диагнозы
5. История посещений

- Архив записей**

1. Возможность просмотра старых записей
2. Поиск по истории

- Профиль врача**

1. Просмотр личной информации
2. Возможность выхода из аккаунта
3. Редактирование данных профиля

- Добавление/редактирование записей**

1. Форма записи от администратора
2. Добавление слотов
3. Удаление или изменение существующих слотов

## **Техническое задание на разработку АРМ регистратуры**

Необходимо разработать мобильное приложение для Android, которое будет использоваться пациентами, врачами и администраторами поликлиники.

Приложение должно предоставлять следующие возможности:

- Запись к врачу через интерфейс
- Просмотр активных и завершённых приёмов
- Отображение жалоб пациента
- Работа с медкартой: заполнение диагноза, рекомендаций, анамнеза
- Хранение данных в облаке или локально

## **1. Общие сведения**

1.1. Наименование системы: АРМ регистратуры «MedReg»

1.2. Код документа по классификационному каталогу: необходимо указать конкретный код в соответствии с внутренними стандартами организации.

## **2. Основание для разработки**

2.1. Предмет, ставший основанием для разработки:

"Информационные системы в здравоохранении"

## **3. Назначение разработки**

3.1. Цель создания системы:

Обеспечить медицинский персонал и пациентов интуитивно понятным интерфейсом для записи, хранения данных и работы с медкартами.

3.2. Обоснование необходимости разработки:

Существующие способы записи к врачу требуют больших временных и человеческих ресурсов. Разрабатываемая система позволит:

- Автоматизировать процесс записи
- Упростить работу регистратуры
- Ускорить взаимодействие между пациентом и врачом
- Сохранять данные медкарт в едином месте

## **4. Требования к системе**

### **4.1. Функциональные требования:**

4.1.1. Возможность выбора свободного слота для записи

4.1.2. Просмотр списка записей у врача

4.1.3. Возможность начала приёма — переход к медкарте

4.1.4. Заполнение и сохранение данных медкарты

4.1.5. Отправка данных из медкарты пациенту

4.1.6. Поддержка ролей: пациент, врач, администратор

## **4.2. Нефункциональные требования:**

4.2.1. Совместимость с Android 8.0+ (API level 26 и выше)

4.2.2. Локализация: только на русский язык

4.2.3. Хранение данных в **SharedPreferences** или Room Database

4.2.4. Интерфейс должен соответствовать Material Design

4.2.5. Защита конфиденциальных данных пациентов и врачей

4.2.6. Высокая производительность: работа без зависаний при большом количестве записей

## **5. Требования к документации**

5.1. Перечень и состав документации:

- Руководство пользователя
- Техническое описание архитектуры
- Документация по API (если используется бэкенд)
- Руководство по установке и тестированию
- Примеры использования

5.2. Требования к оформлению и содержанию: Документация должна быть выполнена в соответствии с внутренними стандартами организации.

Включать:

- Диаграммы UML
- Структуру базы данных
- Сценарии использования
- Описание всех классов и их взаимодействия
- Скриншоты экранов

## **6. Стадии и этапы разработки**

6.1. Этап 1: Формирование технического задания — 1 неделя

6.2. Этап 2: Проектирование архитектуры и макетов — 2 недели

6.3. Этап 3: Реализация базового функционала — 3 недели

6.4. Этап 4: Тестирование и исправление ошибок — 2 недели

## **7. Порядок контроля и приемки**

### **7.1. Порядок проведения испытаний:**

- Проверка корректности записи пациента к врачу
- Проверка передачи данных в медкарту
- Тестирование отображения записей у врача
- Проверка удаления записей после приёма
- Тестирование пользовательского интерфейса на соответствие Material Design
- Проверка безопасности данных и ролей
- Тестирование производительности при множестве записей

### **7.2. Критерии приёмки:**

- Приложение соответствует техническому заданию
- Все функции работают корректно
- Нет критических ошибок и сбоев
- Положительная обратная связь от тестовой группы

# **Реализация проекта**

## **Средства разработки**

Приложение разработано на объектно-ориентированном языке Java , что обеспечивает надёжность и совместимость с Android-устройствами. Разметка интерфейса выполнена на XML. Для хранения данных пациентов, записей, медкарт и информации о врачах используется локальное хранение через **SharedPreferences** и потенциальная интеграция с Firebase для более серьёзной архитектуры. Также реализована возможность работы с анкетой пациента и отображением рекомендаций врача.

## **Пользователи и роли**

В системе определены следующие роли:

### **1. Пациент**

- Может:
  - Выбрать врача
  - Записаться на приём
  - Просмотреть свои записи
  - Получить рекомендации после приёма
  - Просмотреть свою медкарту

### **2. Врач**

- Может:
  - Посмотреть список записей
  - Начать приём
  - Заполнить медкарту пациента
  - Добавить рекомендации
  - Переместить запись в архив

### **3. Администратор**

- Может:
  - Создавать/удалять слоты приёма
  - Удалять врачей
  - Управлять расписанием

Права ограничиваются ролевой моделью:

- Пациент не может редактировать чужие записи
- Врач видит только своих пациентов
- Администратор управляет всей системой

### **Сценарий использования**

1. Пациент открывает приложение
2. Переходит в раздел "Запись к врачу"
3. Выбирает специалиста и свободный слот
4. Подтверждает запись
5. Запись сохраняется в PatientConfirmedAppointment
6. Слот удаляется из общего пула

Сценарий: Начало приёма

1. Врач открывает приложение
2. Видит список записей на сегодня
3. Нажимает на запись → открывается MedicalCardActivity
4. Врач заполняет анкету, диагноз, рекомендации
5. При закрытии формы запись удаляется из активных и добавляется в архив

Сценарий: Просмотр медкарты пациентом

1. Пациент открывает раздел "Моя медкарта"
2. Видит данные, введённые врачом: группа крови, вес, рост, жалобы, рекомендации
3. Может скачать или просмотреть историю посещений

Сценарий: Создание слотов администратором

1. Администратор переходит в раздел "Добавить запись"
2. Выбирает врача, дату, время
3. Сохраняет слот
4. Он появляется в списке доступных для пациентов

Сценарий: Выход из аккаунта

1. Пользователь находится в своём профиле

2. Нажимает "Выход"
3. Приложение очищает текущую сессию
4. Открывается экран входа

### Демонстрация работы приложения

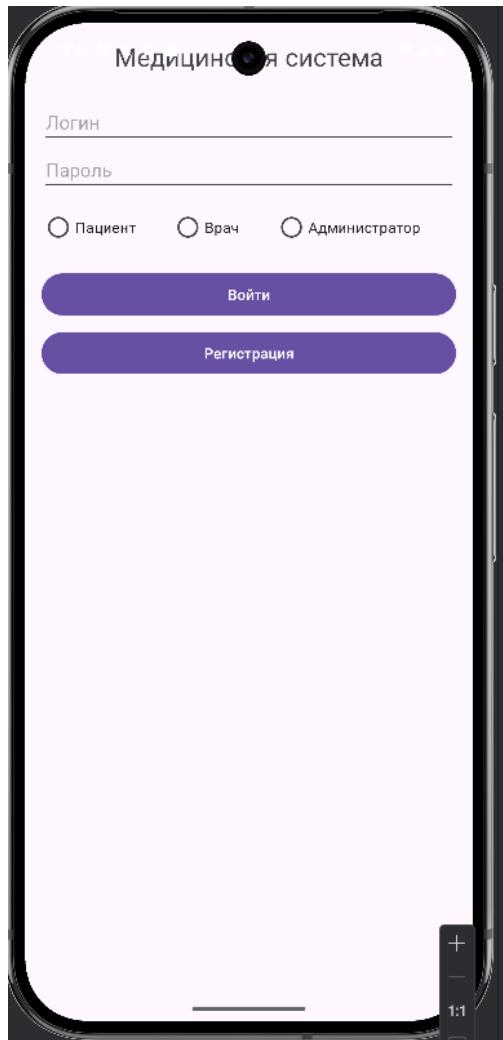


Рисунок 1 – Начальный экран

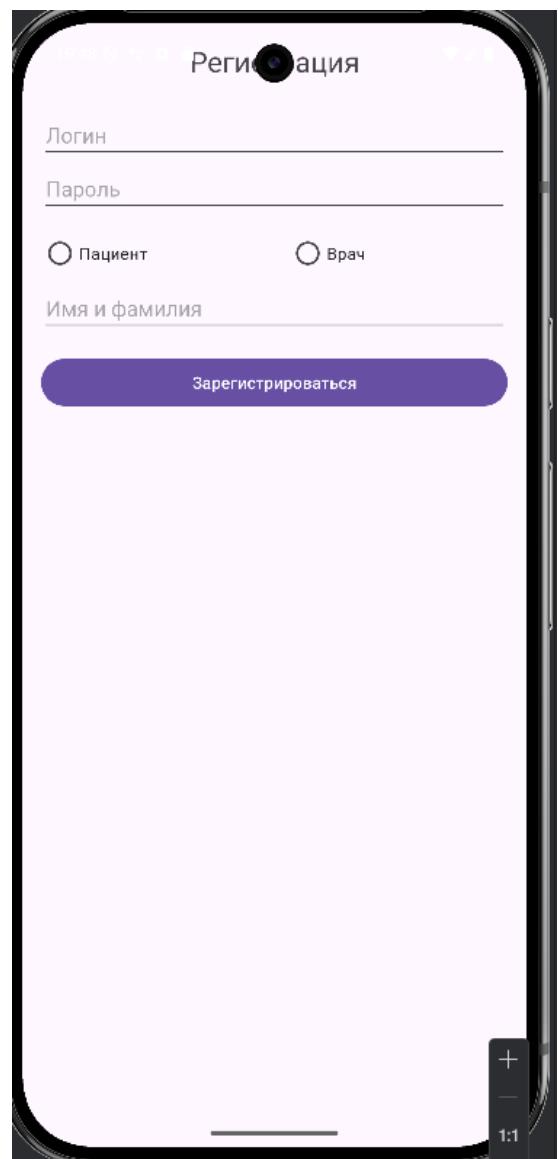


Рисунок 2 -Регистрация

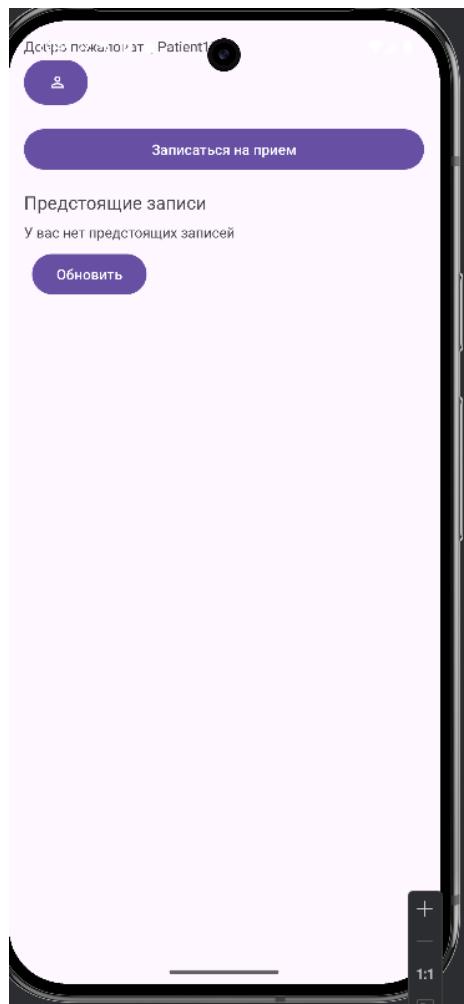


Рисунок 3 – Экран пациента

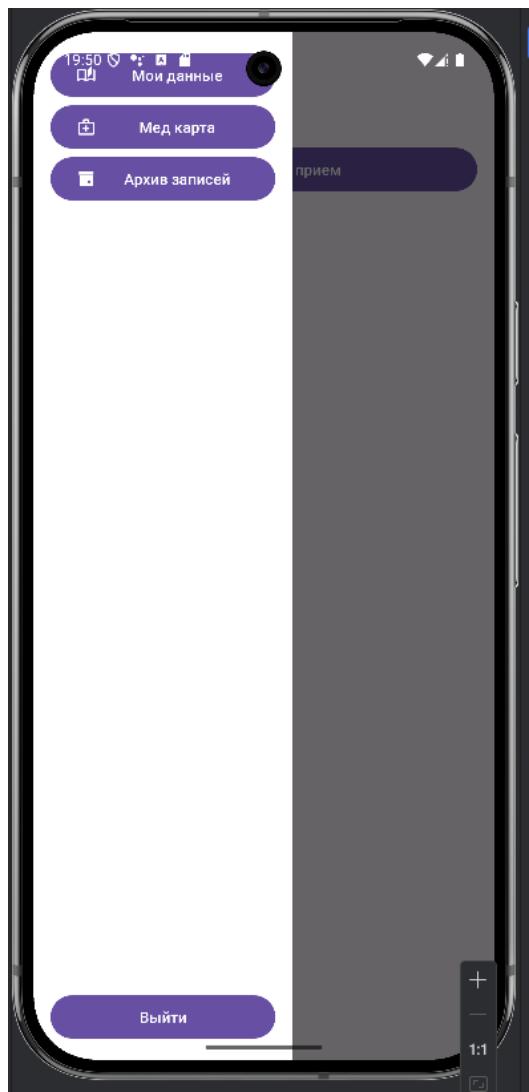


Рисунок 4 – Экран пациента

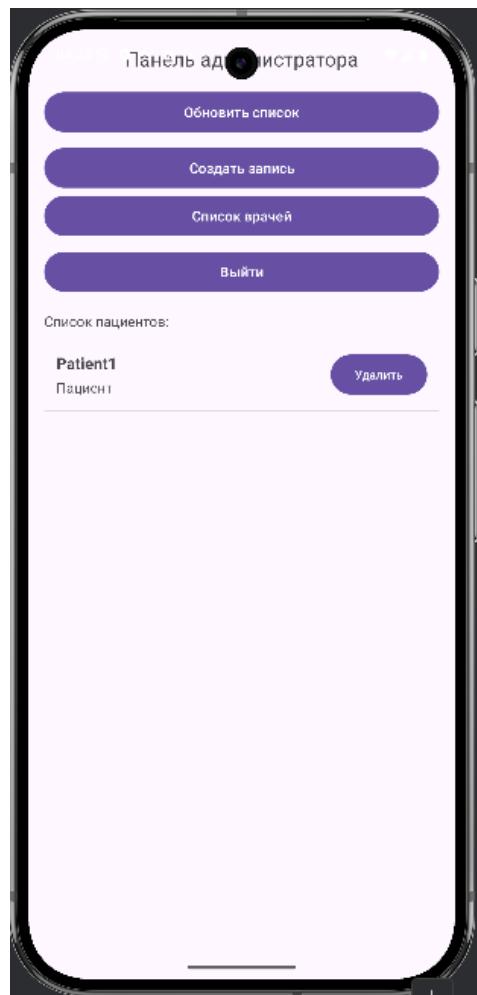


Рисунок 5 – Панель администратора

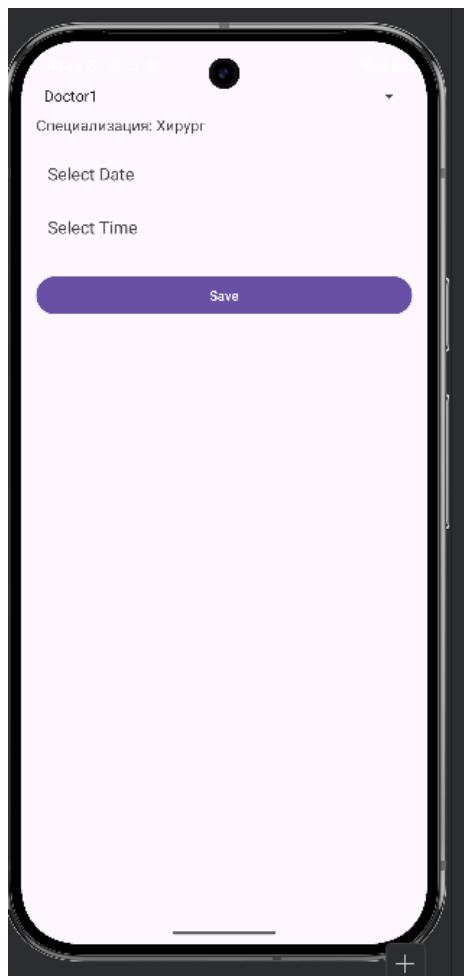


Рисунок 6 – Создание записи администратором

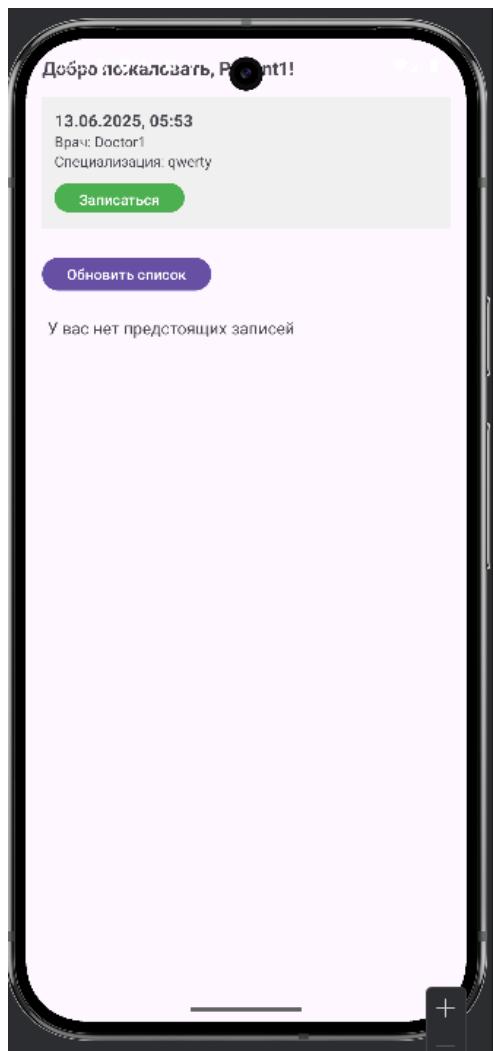


Рисунок 7 – После создания записи у пациента она становится доступна

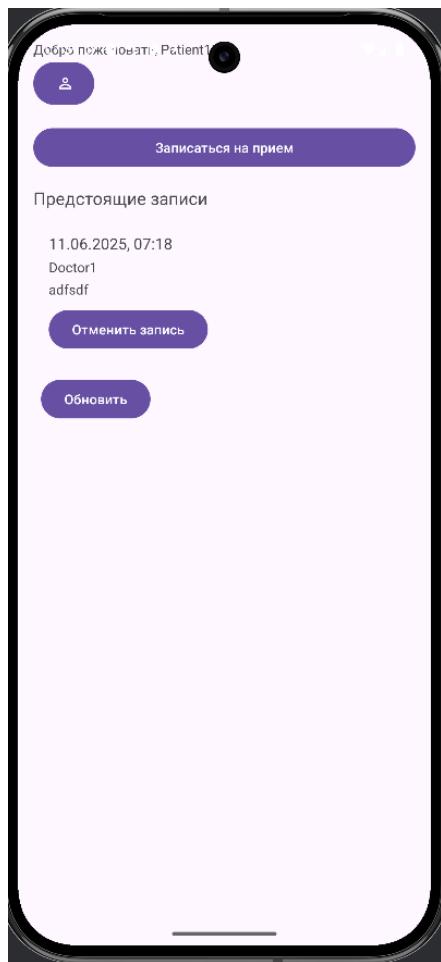


Рисунок 8 – Возможность просмотра и отмены записей

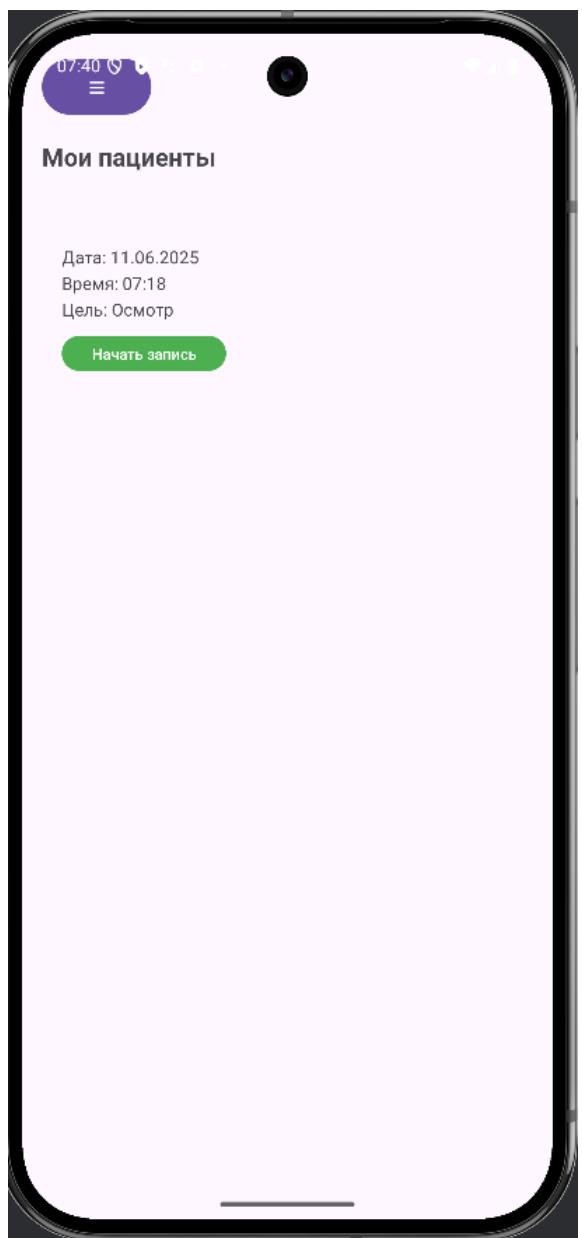


Рисунок 9 – Экран доктора

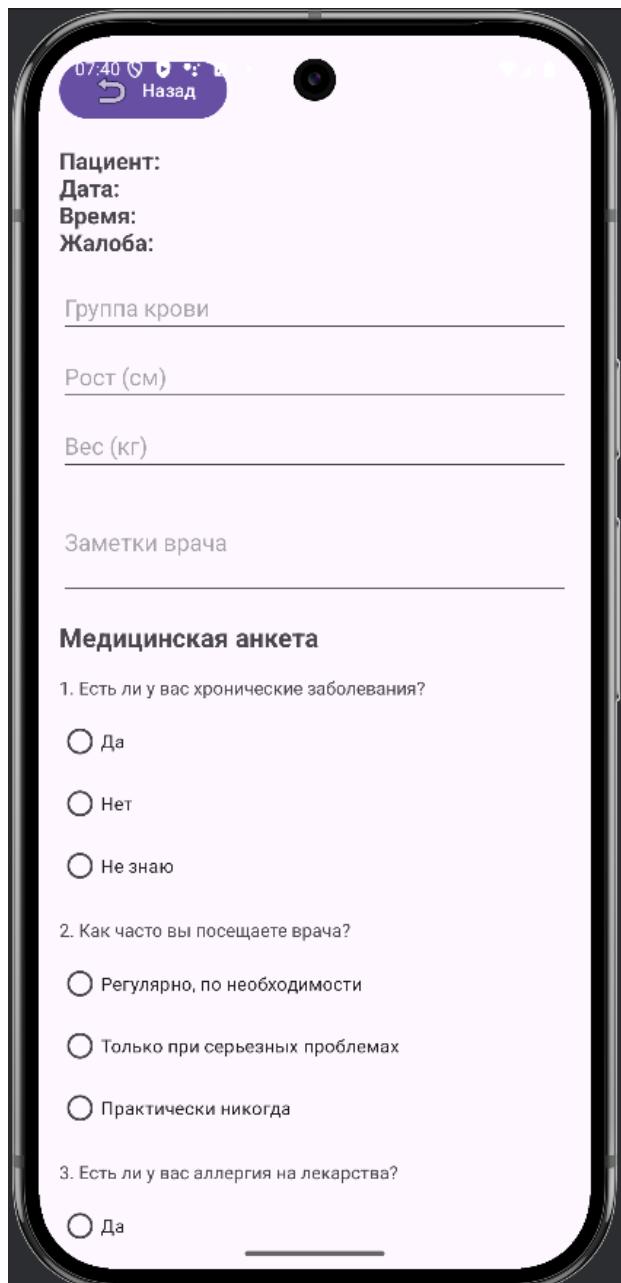


Рисунок 10 – Начало записи с анкетой

## **Вывод**

В рамках данной курсовой работы было разработано мобильное приложение, предназначенное для автоматизации процесса записи пациентов к врачам и упрощения взаимодействия между регистратурой, врачами и пациентами. Приложение реализовано на платформе Android с использованием языка программирования Java и локального хранения данных через SharedPreferences , что позволило создать простую, но функциональную систему без подключения к серверу.

Разработанная система предоставляет следующие возможности:

- Пациент может выбрать свободный слот записи к врачу
- Запись сохраняется под его именем
- После подтверждения запись удаляется из общего пула
- Врач видит список пациентов, записавшихся к нему
- Возможность просмотра истории записей (архива)
- Простое боковое меню с основными действиями: профиль, медкарта, выход

Приложение соответствует современным требованиям к интерфейсам мобильных приложений:

- Используется Material Design
- Поддерживается навигация через DrawerLayout
- Все данные структурированы и безопасно передаются между активностями
- Предусмотрены меры по предотвращению NullPointerException и других ошибок
- Обеспечено удобство использования за счёт интуитивно понятного интерфейса

Актуальность проекта обусловлена тем, что:

- Существующие системы записи к врачу часто требуют личного присутствия пациента или длительного ожидания
- Онлайн-порталы имеют ограничения по регионам и не всегда доступны

- Мобильные решения позволяют сократить время на запись и повысить уровень обслуживания
- Особенно важно — внедрение цифровых решений в условиях растущего спроса на электронные услуги

Данный проект может быть использован:

- В поликлиниках и амбулаториях
- Для внутреннего использования в регистратуре
- Как прототип будущей полноценной медицинской системы с интеграцией в государственные сервисы

Перспективы дальнейшего развития:

- Перенос всех данных в Room Database или Firebase для масштабирования
- Добавление онлайн-синхронизации
- Расширение функционала до полноценной электронной регистратуры
- Интеграция с Госуслугами или ЕГИС
- Реализация push-уведомлений о времени приёма
- Добавление формы жалоб и направлений от врача

Таким образом, разработанное приложение является актуальным и полезным решением, которое демонстрирует, как информационные технологии могут улучшить качество медицинских услуг и сделать их более доступными для населения.

## **Список используемых источников**

1. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО КУРСОВОМУ ПРОЕКТИРОВАНИЮ, 2024. – режим доступа: URL: <https://online-edu.mirea.ru/course/view.php?id=7400>, свободный (дата обращения 19.03.2024)
2. GITHUB glide, 2016. – режим доступа: URL: <https://github.com/bumptech/glide>, свободный (дата обращения 20.05.2024)
3. Firebase Documentation, 2022. – режим доступа: URL: <https://firebase.google.com/docs?hl=en>, свободный (дата обращения 09.05.2024)
4. The Movie Database (TMDB), 2024. – режим доступа: URL: <https://www.themoviedb.org/>, свободный (дата обращения 05.05.2024)
5. IMDb, 2022. – режим доступа: URL: <https://www.imdb.com/>, свободный (дата обращения 13.04.2024)