

Мобильный кроссплатформенный интерфейс интеллектуальной программной системы анализа ультразвуковых снимков щитовидной железы

И. М. Салахов, К. С. Зайцев, М. Е. Дунаев

НИЯУ МИФИ

imsalakhovv@mail.ru

Аннотация. Программный комплекс представляет собой кроссплатформенную информационную систему (Android, iOS, Web, Аврора ОС), предназначенную для автоматизированного анализа ультразвуковых изображений щитовидной железы с целью поддержки принятия врачебных решений и самоконтроля пациентов. Область применения охватывает сферы телемедицины и эндокринологии, обеспечивая удаленную обработку диагностических данных. Функциональные возможности продукта включают загрузку медицинских снимков, их передачу на сервер для обработки алгоритмами искусственного интеллекта, автоматическое выявление узловых образований с расчетом вероятности злокачественности по международной шкале EU-TIRADS, визуализацию результатов с наложением сегментационных масок, ведение персональной истории обследований, а также управление доступом к сервису через систему тарифных подписок и безопасное хранение персональных данных пользователей.

Ключевые слова: телемедицина; искусственный интеллект; ультразвуковая диагностика; автоматическая классификация; TIRADS; кроссплатформенная разработка; мобильное здравоохранение

I. ВВЕДЕНИЕ

Заболевания щитовидной железы представляют значимую проблему современного здравоохранения. Узловые образования щитовидной железы обнаруживаются у 20–76% населения при ультразвуковом исследовании, причём в 5–15% случаев требуется дифференциальная диагностика для исключения злокачественности [1]. В Российской Федерации, по данным аналитического обзора Национального медицинского исследовательского центра эндокринологии, заболевания щитовидной железы имеются почти у 40% населения, а распространенность гипотиреоза и тиреодита составляет соответственно 446 и 428 случаев на 100 000 населения [3].

Ультразвуковое исследование (УЗИ) является ключевым методом первичной диагностики и динамического наблюдения узловых образований щитовидной железы благодаря своей доступности, неинвазивности и экономической эффективности. Однако традиционная интерпретация ультразвуковых данных существенно зависит от опыта оператора, что создаёт риски субъективности и вариабельности заключений. Для стандартизации описания и стратификации риска злокачественности Европейская

тиреоидная ассоциация разработала систему классификации EU-TIRADS, которая определяет категории риска и показания к тонкоигольной аспирационной биопсии [1].

Параллельно с развитием стандартов визуализации активно внедряются технологии искусственного интеллекта (ИИ) для автоматизированного анализа ультразвуковых изображений. Систематический обзор и мета-анализ 2025 года, включивший 28 исследований и более 158 тысяч узлов, показал, что ИИ-модели на основе УЗИ демонстрируют высокую диагностическую точность: объединённая чувствительность 0,89, специфичность 0,84, площадь под кривой SROC 0,93 [2]. Эти данные подтверждают потенциал ИИ как инструмента поддержки принятия врачебных решений.

В условиях географического неравенства в доступе к медицинской помощи телемедицинские решения приобретают особую актуальность для Российской Федерации с её обширной территорией и неравномерным распределением медицинских кадров. Мобильные технологии позволяют преодолевать географические барьеры и обеспечивать доступ к специализированной диагностике для пациентов в удалённых и сельских регионах. Однако большинство существующих решений либо ориентированы на узкий круг платформ, либо не обеспечивают интеграцию с современными ИИ-сервисами анализа медицинских изображений.

Цель данной работы — разработка и описание архитектуры кроссплатформенного мобильного интерфейса интеллектуальной системы анализа ультразвуковых снимков щитовидной железы, обеспечивающего:

- единый пользовательский опыт на платформах Android, iOS, Web и отечественной ОС Аврора;
- интеграцию с разнородными бэкенд-сервисами ИИ через адаптивный слой абстракции;
- визуализацию результатов сегментации и классификации по шкале EU-TIRADS;
- безопасное хранение персональных данных и истории обследований.

Научная новизна заключается в применении архитектуры Kotlin Multiplatform (КМП/СМР) для создания медицинского приложения с поддержкой

импортзамещающей платформы Аврора ОС, а также в реализации механизма динамического переключения между различными ИИ-провайдерами без изменения клиентской логики.

Практическая значимость работы состоит в создании инструмента, который потенциально способен снизить нагрузку на врачей-эндокринологов за счёт предварительной автоматической оценки снимков и повысить доступность скрининга для жителей удалённых территорий Российской Федерации.

II. ФУНКЦИОНАЛЬНАЯ АРХИТЕКТУРА И СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ

A. Назначение системы

Разработанный программный комплекс UziMP представляет собой кроссплатформенную информационную систему, предназначенную для автоматизированного анализа ультразвуковых изображений щитовидной железы. Система функционирует в сфере телемедицины и решает две ключевые задачи:

- Поддержка принятия врачебных решений — предоставление врачу дополнительной информации о выявленных узловых образованиях с оценкой риска злокачественности по международной шкале EU-TIRADS;
- Самоконтроль пациентов — возможность для пациентов отслеживать динамику изменений в рамках преаналитического этапа, до очной консультации со специалистом.

Важно подчеркнуть, что система не заменяет очную консультацию врача и не предназначена для постановки окончательного диагноза. Результаты анализа следует рассматривать как предварительную оценку, требующую верификации квалифицированным специалистом.

Подобные мобильные решения в области здравоохранения (mHealth) демонстрируют потенциал для повышения доступности медицинской помощи, особенно в условиях ограниченного числа специалистов в удалённых регионах [1].

B. Основные сценарии использования

Архитектура приложения построена вокруг трёх функциональных блоков, соответствующих основным сценариям взаимодействия пользователя с системой:

Сценарий 1. Загрузка и отправка снимка на анализ: выбор файла ультразвукового снимка через кроссплатформенный компонент, ввод дополнительных метаданных (дата приёма, название клиники), отправка изображения на сервер для обработки ИИ, отображение прогресса обработки с промежуточными статусами.

Сценарий 2. Просмотр результатов диагностики: отображение обработанного снимка с наложенными масками сегментации, визуализация классификации по шкале EU-TIRADS с цветовой индикацией риска, переключение между исходным и обработанным изображением, доступ к рекомендациям для пациента.

Сценарий 3. Управление историей обследований и подпиской: ведение персональной истории всех диагностик, просмотр результатов прошлых обследований, управление доступом через систему

тарифных подписок, безопасное хранение персональных данных и токенов авторизации.

C. Модульная архитектура приложения

Приложение реализовано с использованием многомодульной архитектуры на базе Kotlin Multiplatform, что обеспечивает строгое разделение ответственности между компонентами системы. Точкой входа и сборщиком всех зависимостей выступает модуль composeApp, который содержит основную навигацию и инициирует работу остальных частей системы. Базовый уровень абстракции представлен модулем core, включающим общие доменные модели, сетевые клиенты для взаимодействия с API, а также библиотеку переиспользуемых UI-компонентов.

Бизнес-логика распределена между специализированными функциональными модулями:

- auth — аутентификация и регистрация пользователей, экраны ввода данных, взаимодействие с сервером авторизации;
- uzi — ключевая функциональность системы: загрузка медицинских снимков, отображение результатов диагностики, визуализация сегментации, история обследований;
- profile — управление профилем пользователя, отображение информации о подписке, тарифные планы;
- subscription — управление платным доступом и тарифными планами, интеграция с платёжными провайдерами.

Инфраструктурные задачи решаются посредством отдельных технических модулей:

- settings — локальное хранение чувствительных данных (токены авторизации, информация о профиле) с использованием технологии DataStore;
- room — кэширование данных в локальной базе данных SQLite для работы с историей диагностик в офлайн-режиме;
- logging — сбор телеметрии и отладочной информации, фиксация событий работы приложения и ошибок;
- core — общие доменные модели, сетевые клиенты, переиспользуемые UI-компоненты (кнопки, поля ввода, контейнеры).

Такая структура позволяет изолировать изменения в одном функциональном домене и минимизировать влияние на остальные части системы.

D. Требования к платформе

Для обеспечения максимальной доступности сервиса система функционирует на широком спектре устройств и операционных систем благодаря технологии Kotlin Multiplatform:

- Android — наиболее массовая мобильная ОС в Российской Федерации и мире, полная поддержка через нативные компоненты;

- iOS — покрытие пользовательской базы устройств Apple, реализация через UIKit и Foundation;
- Web — доступ через браузер без установки приложения, работа в современных браузерах через wasmJs;
- Аврора ОС — поддержка отечественной операционной системы в рамках требований по импортозамещению.

Единая кодовая база для всех платформ обеспечивает унификацию пользовательского опыта и бизнес-логики. Платформенно-специфичные функции, такие как выбор файлов, открытие ссылок в браузере и локальное хранение данных, реализуются через механизм `expect/actual` с использованием нативных возможностей каждой ОС.

Е. Интеграция с внешними сервисами

Функциональная архитектура системы предполагает взаимодействие с несколькими внешними сервисами:

- ML-сервисы анализа изображений — серверные компоненты, реализующие нейросетевые модели сегментации и классификации узловых образований;
- Сервис авторизации — управление учётными записями пользователей, выдача и обновление JWT-токенов;
- Платёжные провайдеры — обработка подписок и платежей за доступ к сервису;
- Сервис логирования — сбор диагностической информации о работе приложения для последующего анализа инцидентов.

Для пользователя взаимодействие с этими сервисами абстрагировано: все запросы выполняются в фоновом режиме, а результаты отображаются в едином интерфейсе приложения.

III. ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ

А. Механизм кроссплатформенной разработки

Поддержка четырёх платформ (Android, iOS, Web, Аврора ОС) реализована с использованием механизма `expect/actual Kotlin Multiplatform`. Платформенно-специфичные функции, такие как выбор файлов, открытие ссылок в браузере и получение информации об устройстве для логирования, объявляются в общем коде и реализуются отдельно для каждой платформы. Такой подход позволяет сохранить единую бизнес-логику при использовании нативных API операционных систем.

В. Адаптивный слой для работы с бэкенд-сервисами

Одной из ключевых архитектурных задач стала необходимость интеграции с разнородными бэкенд-сервисами, реализованными на разных технологических стеках (Golang и Django). Основные отличия между API проявлялись в трёх аспектах:

- Различия в путях эндпоинтов — одинаковые операции доступны по разным URL (например, `/login` в Golang против `/auth/login/` в Django);
- Различия в именах полей — идентичные данные передаются под разными ключами (например, `access_token` против `access`, `fullname` против `fullName`);
- Различия в структуре ответов — некоторые сервисы возвращают `refresh`-токен при обновлении, другие нет.

Для решения этой проблемы был разработан адаптивный слой абстракции на основе интерфейса `AuthDtoUtils`, который определяет единый контракт для преобразования данных между доменной моделью приложения и DTO бэкенда. Реализации `AuthDtoUtilsGolangImpl` и `AuthDtoUtilsDjangoImpl` инкапсулируют специфику каждого бэкенда, что позволяет клиентскому коду работать с единым интерфейсом независимо от выбранного провайдера.

Механизм переключения между провайдерами реализован через внедрение зависимостей (Koin DI), что обеспечивает возможность смены бэкенда без модификации клиентской логики. Такая архитектура предоставляет следующие преимущества:

- Резервирование — возможность переключения на альтернативный бэкенд при недоступности основного;
- A/B-тестирование — параллельное тестирование разных ML-моделей на одних и тех же клиентских устройствах;
- Миграция — постепенный переход между версиями API без выпуска обновлений приложения.

IV. РЕЗУЛЬТАТЫ И ПЕРСПЕКТИВЫ

А. Итоги разработки

В рамках данной работы разработан и реализован рабочий прототип кроссплатформенного мобильного приложения для анализа ультразвуковых снимков щитовидной железы. Приложение функционирует на четырёх платформах: Android, iOS, Web и отечественной операционной системе Аврора ОС. Текущий статус проекта — бета-тестирование, в ходе которого проверяется стабильность работы основных функциональных модулей и корректность взаимодействия с серверными компонентами.

Реализована полная цепочка пользовательского сценария: от авторизации и загрузки медицинского снимка до визуализации результатов сегментации и классификации по шкале EU-TIRADS. Все ключевые модули приложения (`composeApp`, `uzi`, `auth`, `core`, `logging`, `settings`, `room`, `subscription`) интегрированы и проходят проверку в реальных условиях эксплуатации.

В. Потенциальный социальный эффект

Архитектура разработанной системы потенциально способна повысить доступность предварительной ультразвуковой диагностики для жителей удалённых и сельских территорий Российской Федерации. Использование мобильных устройств как точки входа в систему позволяет преодолеть географические барьеры и обеспечить доступ к современным методам анализа медицинских изображений без необходимости очного визита в специализированный центр.

Предполагается, что автоматизированная предварительная оценка снимков может способствовать снижению нагрузки на врачей-эндокринологов за счёт фильтрации случаев, не требующих срочного очного вмешательства, и приоритизации пациентов с высоким риском злокачественности. Однако подтверждение данного эффекта требует проведения дополнительных исследований в условиях промышленной эксплуатации системы.

С. Импортзамещение и соответствие требованиям

Поддержка отечественной операционной системы Аврора ОС реализована на уровне архитектуры Kotlin Multiplatform, что обеспечивает выполнение требований по импортзамещению программного обеспечения в сфере здравоохранения. Единая кодовая база для всех платформ позволяет оперативно адаптировать приложение к изменениям в нормативной базе и техническим требованиям российских регуляторов.

Хранение и обработка персональных данных пользователей осуществляется с соблюдением принципов информационной безопасности: токены авторизации хранятся в защищённом локальном хранилище, передача данных между клиентом и сервером защищена протоколом HTTPS, а архитектура системы предусматривает возможность развёртывания серверных компонентов на инфраструктуре, расположенной на территории Российской Федерации.

Д. Планы развития

Дальнейшее развитие проекта предполагает следующие направления:

- Клиническая валидация — проведение исследований с участием медицинских учреждений для оценки диагностической точности системы в реальных условиях и получения экспертной оценки со стороны врачей-эндокринологов;
- Расширение функциональности — адаптация архитектуры для анализа ультразвуковых изображений других органов (молочная железа, печень, почки) с минимальными изменениями клиентской части;
- Интеграция с медицинскими информационными системами — разработка механизмов обмена данными с региональными и федеральными системами для обеспечения преемственности наблюдения пациентов;
- Сертификация — подготовка документации для регистрации программного комплекса в качестве медицинского изделия в соответствии с требованиями Росздравнадзора.

Реализация данных направлений позволит трансформировать прототип в полноценный инструмент телемедицинской поддержки, готовый к внедрению в практику здравоохранения.

V. ЗАКЛЮЧЕНИЕ

В рамках данной работы разработан и реализован кроссплатформенный мобильный интерфейс интеллектуальной системы анализа ультразвуковых снимков щитовидной железы. Приложение функционирует на четырёх платформах (Android, iOS, Web, Аврора ОС) благодаря архитектуре Kotlin Multiplatform, что обеспечивает единый пользовательский опыт при использовании нативных возможностей каждой операционной системы.

Техническая реализация включает адаптивный слой для интеграции с разнородными бэкенд-сервисами искусственного интеллекта, что позволяет переключаться между различными ML-провайдерами без модификации клиентской логики. Визуализация результатов сегментации и классификации по шкале EU-TIRADS реализована через кастомные Canvas-компоненты с поддержкой масштабирования и навигации по сериям снимков.

Текущий статус проекта — бета-тестирование. Дальнейшее развитие системы предполагает проведение клинической валидации с медицинскими учреждениями, расширение функциональности на другие органы и интеграцию с медицинскими информационными системами для обеспечения преемственности наблюдения пациентов.

БЛАГОДАРНОСТЬ

Авторы выражают признательность сотруднику НМИЦ Эндокринологии им. акад. И.И. Дедова Минздрава России Трухину А.А. за консультации в процессе создания и помощь в тестировании программного модуля.

СПИСОК ЛИТЕРАТУРЫ

- [1] Russ G. et al. European Thyroid Association Guidelines for Ultrasound Malignancy Risk Stratification of Thyroid Nodules in Adults: the EU-TIRADS // European Thyroid Journal. 2017. Vol. 6. No. 5. P. 225–237. – DOI 10.1159/000478927.
- [2] Zhan J. et al. Diagnostic performance of ultrasound characteristics-based artificial intelligence models for thyroid nodules: a systematic review and meta-analysis // Frontiers in Oncology. 2025. Vol. 15. Art. 1614603. – DOI 10.3389/fonc.2025.1614603.
- [3] Мельниченко Г.А., Трошина Е.А., Платонова Н.М. и др. Йододефицитные заболевания щитовидной железы в Российской Федерации: современное состояние проблемы // Консилиум Медикум. 2019. Т. 21. № 4. С. 14–20.