

СОЗДАНИЕ ANDROID-ПРИЛОЖЕНИЯ, ПОМОГАЮЩЕГО СПОРТСМЕНАМ ОТСЛЕЖИВАТЬ РЕЗУЛЬТАТЫ

Пышный Е. С.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Сурков К. А. – ст. преподаватель

Цель исследования – продемонстрировать инженерные решения, позволившие создать мобильное приложение для комплексного отслеживания спортивной деятельности, включая соревнования, тренировки, медицинские показатели, питание и спортивные сборы. В статье рассматриваются вопросы выбора технологического стека, проектирования архитектуры, реализации ключевых функций и обеспечения стабильной работы приложения. В результате показано, что выбранные подходы позволяют создать масштабное и надёжное решение, предоставляющее пользователям уникальный опыт управления своей спортивной активностью.

Современные требования к подготовке спортсменов предполагают системный подход к анализу физического состояния, тренировочных нагрузок и восстановления. Однако существующие решения часто фокусируются на отдельных аспектах (например, дневник тренировок или медицинские показатели), что усложняет формирование целостной картины. Разработка приложения, объединяющего инструменты для отслеживания соревнований, физических параметров, медицинских данных и повседневной активности, демонстрирует возможность создания универсальной платформы, автоматизирующей сбор и визуализацию ключевых метрик для спортсменов и тренеров.

Разработка мобильного приложения с широким функционалом традиционно требует значительных ресурсов, включая создание сложных механик, интеграцию аналитики и обеспечение стабильности в условиях высокой нагрузки. Однако использование современных технологий и модульной архитектуры позволило реализовать проект силами небольшой команды, минимизируя риски и ускоряя процесс разработки.

Приложение реализовано по клиент-серверной модели с использованием современных инструментов.

Клиентская часть написана на Kotlin с применением Jetpack Compose что обеспечивает адаптивный интерфейс, быструю разработку и нативную производительность на Android. Для сетевого взаимодействия выбраны Retrofit и OkHttp благодаря их стабильности, поддержке асинхронных запросов и удобной интеграции с REST API.

Серверная часть на Python с фреймворком FastAPI обеспечивает высокую скорость обработки запросов и автоматическую генерацию документации Swagger. Данные хранятся в PostgreSQL с использованием ORM SQLAlchemy, что гарантирует надёжность и гибкость работы с relational-моделями. Миграции управляются через Alembic для контроля изменений схемы БД и быстрого развертывания БД на сервере.

Аутентификация реализована на основе SID-сессий, хранящихся в Redis, что обеспечивает безопасность, низкую задержку при проверке токенов и масштабируемость, а Redis применяется для кэширования и снижения нагрузки на сервер.

Архитектура приложения построена на принципе модульности. Клиентская часть включает модули для работы с соревнованиями, учебно-тренировочными сборами, результатами тестирования ОФП/СФП, результатами медицинских и комплексных обследований, а также для мониторинга антропометрических параметров и ведения дневника тренировок для мониторинга сна, питания, тренировок и активности в течении дня. Также клиент включает репозитории для синхронизации с сервером.

Ключевые функциональные возможности приложения сосредоточены на комплексном решении задач спортивной подготовки. Основной акцент сделан на инструментах для организации и контроля тренировочного процесса. Важной составляющей является система мониторинга физического состояния, позволяющая фиксировать результаты тренировок, отслеживать динамику различных антропометрических показателей и результатов тестирования физической подготовки.

Наиболее часто пользователь будет взаимодействовать с модулем дневника тренировок. В дневнике тренировок пользователь может отслеживать ключевые параметры такие как сон, питание и активность в течении дня. Для отслеживания сна пользователь фиксирует время засыпания, продолжительность сна, субъективную оценку качества сна по шкале от 1 до 10, состояние после сна от 1 до 10, а также ЧСС лежа и стоя за 15 сек. Для отслеживания питания пользователю необходимо самому добавлять приемы пищи пометчая время приема, субъективную оценку аппетита перед едой, и вводить БЖУ еды с описанием, а также пользователь может добавлять количество выпитой за сутки воды и количество спортивного питания. Для отслеживания активности пользователь фиксирует субъективную оценку активности в течении дня (без учета тренировок) от 1 до 10, субъективную оценку самочувствия вечером от 1 до 10, субъективную оценку утомления за день от 1 до 10, а также время отхода ко сну. На рисунке 1 изображён дизайн экранов дневника тренировок.

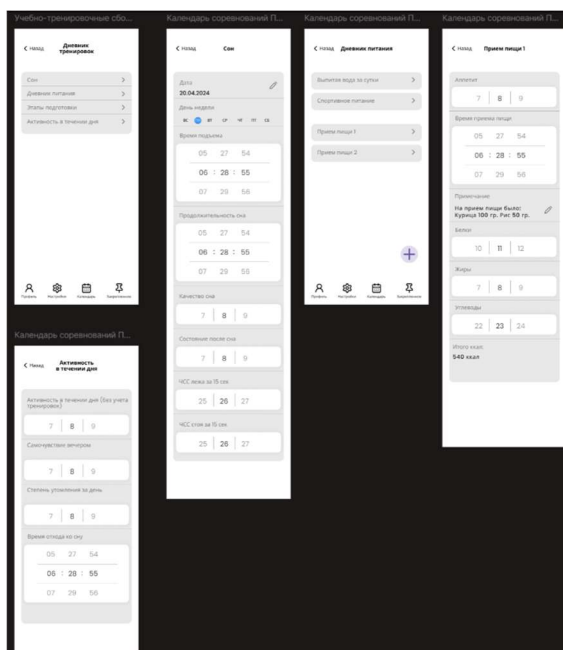


Рисунок 1 – Дизайн экранов дневника тренировок

Медицинский блок приложения обеспечивает систематизацию информации о посещениях специалистов, диагнозах и врачебных рекомендациях, создавая целостную картину состояния здоровья спортсмена.

Также приложение предоставляет возможность для сохранения результатов общего/специального физического тестирования и антропометрических параметров. По этим данным пользователь может построить график для отслеживания своего прогресса.

Еще одним из основных модулей является модуль соревнований, он предоставляет возможность для просмотра списка всех соревнований, отсортированных относительно текущей даты на прошедшие/текущие/будущие, просмотра и изменения подробной информации о соревновании, добавление нового соревнования, удаление, просмотра списка всех дней соревнования, а также просмотра и редактирования подробной информации о конкретном дне.

Особое внимание уделено обеспечению стабильности и производительности. Проведено нагрузочное тестирование сервера, оптимизированы SQL-запросы и индексация в PostgreSQL. Данные передаются по HTTPS с шифрованием алгоритмом AES-256, а резервное копирование базы выполняется ежедневно. Асинхронная обработка запросов через корутины в Kotlin и асинхронные методы FastAPI предотвращает блокировку интерфейса.

Результаты разработки подтверждают эффективность выбранных решений. Модульность архитектуры ускорила процесс разработки, Jetpack Compose обеспечил создание интуитивного интерфейса с плавными анимациями, а сочетание FastAPI и PostgreSQL гарантировало высокую производительность сервера даже при пиковых нагрузках. Приложение демонстрирует, что современные технологии позволяют небольшим командам создавать комплексные решения для профессиональных спортсменов, объединяя аналитику, безопасность и удобство использования.

Список использованных источников:

1. Jetpack Compose Docs [Электронный ресурс]. – Режим доступа: <https://developer.android.com/develop/ui/compose/documentation> – Дата доступа: 25.02.2025