**Разработка мобильного приложения для фитнес-тренировок**

**▎1. Введение**

Данное техническое задание (ТЗ) предназначено для разработки мобильного приложения, которое будет служить платформой для фитнес-тренировок. Приложение должно предоставлять пользователям доступ к тренировочным программам, видеоурокам и возможностям отслеживания прогресса.

**▎2. Цели и задачи**

Цели:

• Создание удобного и интуитивно понятного интерфейса для пользователей.

• Обеспечение возможности взаимодействия между пользователями и тренерами.

• Реализация системы отслеживания прогресса и результатов тренировок.

Задачи:

1. Разработка пользовательского интерфейса (UI) с учетом принципов UX.

2. Реализация функционала для создания и управления тренировочными программами.

3. Внедрение системы отслеживания прогресса пользователя.

4. Обеспечение возможности обратной связи между пользователями и тренерами.

5. Реализация системы уведомлений и напоминаний о тренировках.

**▎3. Функциональные требования**

1. Регистрация и авторизация пользователей:

• Возможность регистрации через электронную почту и социальные сети.

• Реализация многофакторной аутентификации.

2. Создание тренировочных программ:

• Тренеры должны иметь возможность создавать новые программы, добавлять описание, видеоуроки, упражнения и рекомендации по питанию.

• Возможность редактирования и удаления программ.

3. Отслеживание прогресса:

• Пользователи должны иметь возможность отслеживать свои достижения (вес, количество выполненных подходов, время тренировки).

• Графическое отображение прогресса в виде диаграмм и таблиц.

4. Обратная связь:

• Возможность оставлять отзывы о тренировочных программах и тренерах.

• Чат для общения пользователей с тренерами.

5. Уведомления:

• Реализация системы уведомлений о новых тренировках, предстоящих занятиях и сроках выполнения заданий.

**▎4. Нефункциональные требования**

1. Производительность:

• Приложение должно поддерживать одновременное подключение не менее 500 пользователей.

2. Безопасность:

• Все данные пользователей должны быть защищены с помощью шифрования.

• Регулярные резервные копии данных.

3. Кросс-платформенность:

• Поддержка iOS и Android.

4. Интерфейс:

• Адаптивный дизайн для различных экранов мобильных устройств.

**▎5. Технические требования**

1. Технологический стек:

• Frontend: React Native или Flutter.

• Backend: Node.js с использованием Express.js.

• База данных: MongoDB или Firebase.

2. Интеграции:

• API для интеграции с платежными системами (например, PayPal, Stripe).

• Интеграция с системами аналитики (Google Analytics).

**▎6. Сроки выполнения**

• Разработка прототипа: до 15 октября.

• Разработка функционала: до 30 ноября.

• Тестирование: до 15 декабря.

• Запуск приложения: до 31 декабря.

**▎7. Бюджет**

• Общий бюджет на разработку: 400 000 рублей.

▎Распределение бюджета:

• Дизайн и прототипирование: 100 000 рублей

• Разработка фронтенда: 150 000 рублей

• Разработка бэкенда: 100 000 рублей

• Тестирование и запуск: 50 000 рублей

Разработка мобильного приложения для фитнес-тренировок требует выбора технологий и языков программирования, которые обеспечат хорошую производительность, удобство использования и возможность интеграции с различными устройствами. Вот несколько технологий и языков, которые стоит рассмотреть:

**▎1. Языки программирования**

▎a. Swift

• Платформа: iOS

• Почему использовать: Swift — современный язык программирования от Apple, который обеспечивает высокую производительность и безопасность. Он идеально подходит для разработки приложений под iOS и имеет множество библиотек для работы с графикой и анимацией.

▎b. Kotlin

• Платформа: Android

• Почему использовать: Kotlin является официальным языком для разработки Android-приложений. Он более лаконичен и безопасен по сравнению с Java, что позволяет быстрее разрабатывать приложения с меньшим количеством ошибок.

▎c. JavaScript (React Native)

• Платформа: iOS и Android

• Почему использовать: React Native позволяет создавать кроссплатформенные приложения с использованием JavaScript. Это ускоряет процесс разработки, поскольку вы можете использовать один код для обеих платформ.

▎**2. Фреймворки и инструменты**

▎a. Flutter

• Почему использовать: Flutter — это UI-фреймворк от Google для создания кроссплатформенных приложений. Он использует язык Dart и позволяет создавать красивые интерфейсы с высокой производительностью.

▎b. Xamarin

• Почему использовать: Xamarin позволяет разрабатывать приложения на C# и предоставляет доступ к нативным API для iOS и Android. Это также хороший выбор для кроссплатформенной разработки.

**▎3. Базы данных**

▎a. Firebase

• Почему использовать: Firebase предлагает облачное решение для хранения данных, аутентификации пользователей и аналитики. Это упрощает разработку бэкенда и позволяет легко интегрировать функции, такие как чат или отслеживание прогресса тренировок.

▎b. SQLite

• Почему использовать: SQLite — это легковесная база данных, которая может быть встроена в приложение. Она подходит для хранения локальных данных, таких как пользовательские настройки или история тренировок.

▎**4. API и интеграции**

▎a. HealthKit (iOS) и Google Fit (Android)

• Почему использовать: Эти API позволяют интегрировать ваше приложение с системами отслеживания здоровья и физической активности, что может быть полезно для фитнес-приложений.

▎Заключение

Выбор технологий зависит от целевой аудитории, бюджета и функциональности приложения. Кроссплатформенные решения, такие как React Native или Flutter, могут значительно сократить время разработки, в то время как нативные языки, такие как Swift и Kotlin, обеспечивают максимальную производительность и доступ к специфическим функциям платформы.