Техническое задание на разработку мобильного приложения "Дзен Подписок"

# 1. Краткое описание проекта (суть)

Название мобильного приложения: Подписки

Описание: Простое и удобное приложение для учета подписок и регулярных платежей.

Цель: Создать эффективный инструмент для контроля расходов на подписки и напоминаний о платежах.

Аналоги: Subscrob, Bobby,Rocket Money.

# 2. Основные требования (Функциональные)

## 2.1. Пользовательские роли

Пользователь — добавление и управление подписками, получение уведомлений, просмотр статистики.

## 2.2. Базовый функционал (MVP)

- Управление подписками: добавление, редактирование, удаление.  
- Поля подписки: название сервиса, стоимость, валюта, дата следующего платежа, частота платежей.  
- Напоминания: push-уведомления за настраиваемое количество дней до платежа.  
- Статистика: общая сумма предстоящих и фактических платежей за месяц.  
- Оффлайн-режим: хранение данных локально на устройстве.

## 2.3. Нефункциональные требования

- Платформа: Android 8+.  
- Производительность: время отклика основных экранов <200 мс.  
- Безопасность: локальное шифрование базы данных.  
- Конфиденциальность: приложение не требует регистрации и не передает данные третьим лицам.  
- Локализация: русский язык (английский опционально).

# 3. Предложенный стек технологий

- Frontend: Flutter.  
- Язык программирования: Dart.  
- База данных: локальная (SQLite).  
- IDE: Android Studio.  
- Figma: прототипирование.

# 4. Архитектура и данные

Приложение работает полностью оффлайн. Данные сохраняются в локальной базе SQLite.

## 4.1. Структура БД

# Servis: {id\_ser, servis,logo\_ser}

# Sub: {id\_sub,ser\_id, cost, chastota, next\_payment\_date, is\_active, created\_at}

# 5. UI/UX: основные экраны

- Экран подписок: список всех активных подписок.  
- Экран добавления/редактирования подписки: форма с полями.  
- Экран статистики: сумма расходов за месяц.  
- Экран уведомлений: список напоминаний о платежах.

# 6. Тестирование

- Unit-тесты базовой логики.  
- Интеграционные тесты добавления и редактирования подписок.  
- Проверка push-уведомлений.  
- Тестирование на разных версиях Android.

# 7. План разработки

Этап 1 — MVP (1 неделя):  
 - Реализация интерфейса добавления/просмотра подписок.  
 - Настройка локальной базы данных.  
 - Настройка push-уведомлений.  
  
Этап 2 — Тестирование и оптимизация (1 неделя):  
 - Проверка работы приложения на устройствах.  
 - Оптимизация скорости работы.  
 - Финальная отладка.