**Проект: "CatGallery"**

**Описание:**  
Создайте мобильное приложение на React Native, посвящённое котикам. Приложение должно позволять пользователям просматривать галерею изображений котов, регистрироваться, сохранять избранное, загружать собственные фотографии, получать push-уведомления и делиться контентом в социальных сетях. Реализуйте офлайн-режим с кэшированием данных, динамическую тему (светлая/тёмная) и продвинутые анимации.

**Описание:**  
Создайте мобильное приложение на React Native, посвящённое котикам. Приложение должно позволять пользователям просматривать галерею изображений котов, регистрироваться, сохранять избранное, загружать собственные фотографии, получать push-уведомления и делиться контентом в социальных сетях. Реализуйте офлайн-режим с кэшированием данных, динамическую тему (светлая/тёмная) и продвинутые анимации.

**Основные требования**

1. **Пользовательская аутентификация:**
   * **Регистрация и вход:** Реализуйте регистрацию и авторизацию пользователей (например, с помощью Firebase Auth или собственного решения).
   * **Социальные сети:** Добавьте возможность входа через Google, Facebook или другие популярные сервисы.
2. **Работа с API и локальное кэширование:**
   * **Получение данных:** Используйте публичный API, например [TheCatAPI](https://thecatapi.com/), для получения изображений и дополнительной информации о котах.
   * **Обработка ошибок и загрузка:** Реализуйте индикацию загрузки данных и обработку ошибок.
   * **Офлайн-режим:** Сохраняйте данные локально (через AsyncStorage, SQLite или Realm), чтобы обеспечить доступ к информации даже без подключения к интернету.
3. **Просмотр и управление галереей:**
   * **Список изображений:** Отображайте изображения в виде прокручиваемого списка (FlatList) с поддержкой "pull-to-refresh".
   * **Детальный просмотр:** При выборе изображения открывайте экран с детальной информацией, реализуя плавные анимационные переходы (например, с использованием react-native-reanimated).
   * **Фильтрация и поиск:** Добавьте возможность сортировки или фильтрации изображений по категориям или другим параметрам.
4. **Избранное и локальное хранение:**
   * **Механизм избранного:** Позвольте пользователям отмечать понравившиеся изображения, сохраняйте эти данные в локальном хранилище для доступа между сессиями.
5. **Загрузка и отправка изображений:**
   * **Загрузка фото:** Реализуйте функциональность для загрузки собственных фотографий котов, используя камеру или выбор из галереи.
   * **Предварительный просмотр:** Добавьте возможность предпросмотра изображения перед отправкой и базовую обработку (например, обрезку или фильтрацию).
6. **Push-уведомления и "Кот дня":**
   * **Уведомления:** Настройте push-уведомления (например, с помощью Firebase Cloud Messaging), чтобы информировать пользователей о новых изображениях или "Коте дня".
   * **Персонализация:** Разработайте логику, позволяющую пользователям получать уведомления по выбранным тематикам или категориям.
7. **Расширенный стейт-менеджмент:**
   * **Управление состоянием:** Используйте Redux, MobX для организации глобального состояния приложения.
   * **Асинхронные действия:** Реализуйте обработку асинхронных операций (например, с помощью Redux-Saga или Redux-Thunk).
8. **UI/UX и динамическая тема:**
   * **Адаптивный дизайн:** Обеспечьте корректное отображение приложения на различных устройствах и платформах (iOS и Android).
   * **Динамическая тема:** Реализуйте поддержку светлой и тёмной тем, позволяющую пользователю переключаться между режимами.
   * **Анимации:** Добавьте продвинутые анимационные эффекты для переходов между экранами и интерактивных элементов.