

# Prankster Mobile Application

## A React Native App with Appwrite Authentication

April 25, 2025

### Abstract

**Prankster** is a mobile application designed for entertainment, allowing users to securely log in and explore prank-related content. Built using React Native and Expo Router, the app features user authentication powered by Appwrite, a secure backend-as-a-service. This document outlines the project's architecture, functionality, and technology stack.

## 1. Introduction

**Prankster** is a cross-platform mobile application that introduces users to a fun, lighthearted platform to share and explore pranks. The main aim is to provide an intuitive, secure, and scalable environment for users using modern frontend and backend technologies.

## 2. Technology Stack

- **React Native (v0.76.9)** — Cross-platform mobile development.
- **Expo Router (v4.0.20)** — File-based navigation system.
- **Appwrite (v17.0.2)** — Open-source backend for authentication and user management.
- **TypeScript (v5.3.3)** — Strongly typed JavaScript for safer development.

## 3. Features

1. **User Authentication:** Secure sign-up and login using Appwrite.
2. **Navigation:** Smooth screen transitions using Expo Router.
3. **UI/UX:** Clean, responsive layout built with React Native's styling system.
4. **Modular Structure:** Screens and services separated into logical directories for scalability.

## 4. Appwrite Integration

To integrate Appwrite:

- Register at Appwrite Cloud.
- Create a new project and platform.
- Enable **Email & Password** authentication.
- Replace the endpoint and project ID in `appwrite.ts`.

## 5. Screens Overview

### 5.1. Login Screen

Allows users to log in with email and password. If authenticated, users are navigated to the Home screen.

## 5.2. Sign Up Screen

Registers new users using Appwrite's `create()` API and stores the user data securely.

## 5.3. Home Screen

Initial landing screen post-login.

## 6. Conclusion

**Prankster** is a scalable foundation for a fun-oriented mobile app. By leveraging modern frameworks and secure backend services, it ensures a smooth user experience with the potential for future expansion.