React Native Mobile App Developer Internship Assignment

1. Introduction

Thank you for your interest in joining our team as a **React Native Mobile App Developer Intern**. This assignment is designed to evaluate your understanding of React Native fundamentals, UI implementation, state management, API handling, and basic real-time communication using Zego Cloud SDK.

You will be required to build a mobile application based on the provided design and functionality requirements.

---

2. Design Reference

All UI screens and design specifications are available in the Figma file:

Figma Link: Design File

- The UI matches the design as closely as possible
- Proper spacing, fonts, colors, and alignment are maintained
- The app is responsive for both Android & iOS devices

---

3. Assignment Overview

You are required to develop a React Native mobile applications with the following core features:

Core Functionalities

1. Implement all screens as per the Figma design

2. Navigation flow must replicate the design
3. Integrate **audio/video calling** using **Zego Cloud SDK**
4. Implement basic state management (Context API or Redux)
5. Handle form validations where applicable
6. Dummy data is acceptable unless specified

---

4. Technical Requirements

Tech Stack

- **React Native (0.81 +)**
- **TypeScript** (preferred but optional)
- **React Navigation**
- **State Management:** Context API / Redux
- **Audio/Video Calls:** Zego Cloud SDK

Platform Support

- Android
- iOS (optional but preferred)

---

5. Zego Cloud SDK Integration Requirements

For audio/video calling functionality, you must use **Zego Cloud Real-Time Communication SDK**.

1. Create Zego Account

- Sign up at Zego Cloud (https://www.zegocloud.com)
- Generate **AppID** and **AppSign** from the console

2. Expected Features

- One-to-one video call
- Mute/unmute microphone
- Enable/disable camera
- Call end functionality

---

6. Deliverables

You must submit the following:

Required

- Complete source code (GitHub repo link)
- README with setup instructions
- Screens implemented as per design
- Working video demo (screen recording)
- Zego calling integration

Bonus Points For

- TypeScript usage
- Clean architecture (e.g., folder structure, reusable components)
- Pixel-perfect UI
- Error handling and loading states

- Using environment variables for credentials

---

7. Evaluation Criteria

Your submission will be evaluated based on:

- UI accuracy compared to Figma
- Code quality & readability
- State and navigation handling
- Successful Zego SDK integration
- Completion of all required features
- Problem-solving approach & documentation

---

8. Submission Guidelines

- Submit
    - GitHub repository link
    - App demo video link
    - Any additional notes or dependencies

---

9. Support & Rules

- You may use official documentation and online resources
- Do **not** copy existing repos or third-party templates

Good luck!

We look forward to reviewing your work and potentially welcoming you to our team