

# Project Proposal

Objective: To build a full-stack application for android and ios devices that helps users connect with language interpreters.

## Step-by-Step navigation of application

### 1. Landing Page

This will be the face of our application. Users will get to know what the app does in a short introductory way.

Includes: relevant images and links.

#### **Tech stack needed:**

react-native

### 2. Sign up and Sign in section

This section of the app will be solely responsible for the onboarding of users in two roles, as a user or as an interpreter.

First, there will be an option to sign up as a user or an interpreter.

The sign-up section for USER will contain the following:

Input fields: name, age, occupation, phone no., email address, password, and profile picture.

The sign-up section for INTERPRETER will contain the following:

name, age, interpreter for (roles), phone no., email address, password, and profile picture.

Both User and Interpreter can sign-in using their email address and password.

#### **Tech stack needed:**

frontend: react-native, formic/yup (for form validation), redux (for managing state)

Backend: Node.js with express (for building RESTful APIs), JWT and Passport.js (for authentication and authorization of user), express-validator and bcrypt (for security)

Database: MongoDB

### 3. Main section

The most important section of our app is the call logging section.

This section will have the following:

1. A call button where a user can click it and they will get connected to an interpreter
2. At the interpreter side, a message displaying, 'username is calling you' where username is the name of the user, will occur.

**Tech stack needed:** ZegoCloud' s API to handle voice calls (first 1000 minutes are free)

### 4. Feedback and history

The last feature of the app, will be the call history. Here users can see the interpreters that were once connected and drop feedbacks and ratings.

For the interpreter side as well, there will be history of users they managed to interpret for.