roots

mai h | naomi l | angela w | aimen e

Introduction

roots is an app designed to strengthen connections between immigrant parents and their children. Parents progress through multi-modal lessons about effective communication, practice these skills, then engage in playful activities face to face with their kid designed to break the ice and allow them to get to know each other better. The app culminates in a digital "scrapbook" where parents can record memories of playing with their children.

Design Tools

We built our high fidelity prototype using React Native, Expo, and Supabase. We used Apple's Xcode Simulator and the ExpoGo mobile application to test.

Operating Instructions

1. Download the "Expo Go" application from the App Store (Apple iOS devices)or the Google Play Store (Android devices) onto your device.

2. Scan this QR code to download the app and try it out!

3. After opening the app on your device, please create an account by providing an email address and password. Once the account is created, all work completed in the app will be saved to this account.

Limitations

The prototype has some limitations, namely it only currently supports one end-to-end flow: Lesson 1 → Game 1 → Scrapbook. In a full production version of Roots, we would:

- Add multiple lessons covering a broader range of empathy and communication skills.
- Define progression rules (e.g., “complete Lesson 2 to unlock Game 2”) so that games are systematically unlocked as families progress.
- Allow users to revisit and reorder lessons or games based on their interests and needs.

Additionally, the app does not support personalization or adaptive feedback. At the moment, the app:

- Does not adapt to a family’s specific needs, background, or responses over time.
- Does not provide targeted feedback on the user’s written reflection prompts. (For example, the app doesn’t highlight strengths in a parent’s response or suggest more effective phrasing.)

Wizard of Oz Techniques

To simulate features that would require a more complex backend or personalization engine in a production app, we used several Wizard-of-Oz techniques. We simulated robust multilingual support with the React-i18next library, which provides a language toggle to demonstrate how the interface could switch between languages. The i18next library loads static translation strings for core UI elements, including the lessons/lesson content and games/game content. Additionally, the onboarding intake survey signals how the app could become tailored to each family’s linguistic, cultural, and emotional needs, but there is no logic actually defined to change the app according to the user’s answers.

Hard Coded Items

Lesson 1 is hardcoded and the rest of the “locked” lessons are statically rendered. The onboarding state also does not persist. The app does not store whether a user has already completed onboarding, so the user must “complete” onboarding every time they log into the app. Finally, the set of possible cards for Game 1 (question/action cards) is currently hard-coded. On each run, the app uses a simple randomization logic to select a random subset of cards from Supabase so the user sees different combinations of cards on different playthroughs.