



# eMotion

Linda D. | Ethan F. | Devorah S. | Hawi A.

## High Fidelity Prototype README

### Tools Used

The tools that we used to implement our Hi-Fi prototype include ReactNative as a software framework, Expo to create and run the application, Visual Studio code as our IDE, or integrated development environment, and GitHub, to collaborate.

### Operating Instructions

Our prototype is accessible on any iOS device, android device, or online, although it is optimized for iOS. To access the app using an iOS device:

- Download "Expo Client" from the App Store
- Login to Expo Client with username: TeamMovement and password: eMotion123!
- Visit <https://expo.dev/@teammovement/eMotion?serviceType=classic&distribution=expo-go> on your phone browser, or scan the QR code
- Click on the button labeled "Open project using Expo", and press "Open" when prompted



### Wizard of Oz Techniques

Our wizard of oz technique is employed in the Community tab of eMotion. There is no real database of others' data, so when a user accesses this tab, data about other users' emotions and movements are made up. Additionally, when the user presses "copy invite link", they receive an alert that an invite link was copied to their clipboard, when no link is really created or copied. This is done to simulate the user experience rather than focus on the implementation of an invite system.

## Hard-Coded Techniques

To simulate a realistic user experience, we had to hard-code several elements of our app. These include:

- Movement data to suggest exercises based on a user's emotions
- List of contacts that a user is able to add as friends
- Previously logged user workouts

## Limitations

While we were able to implement all of our necessary task-flows, there are a few features that we weren't able to implement, which include:

- User ability to choose an exercise outside of our hard-coded list of options
- Option to remove a movement from a past or current workout
- Real communication between users of the app