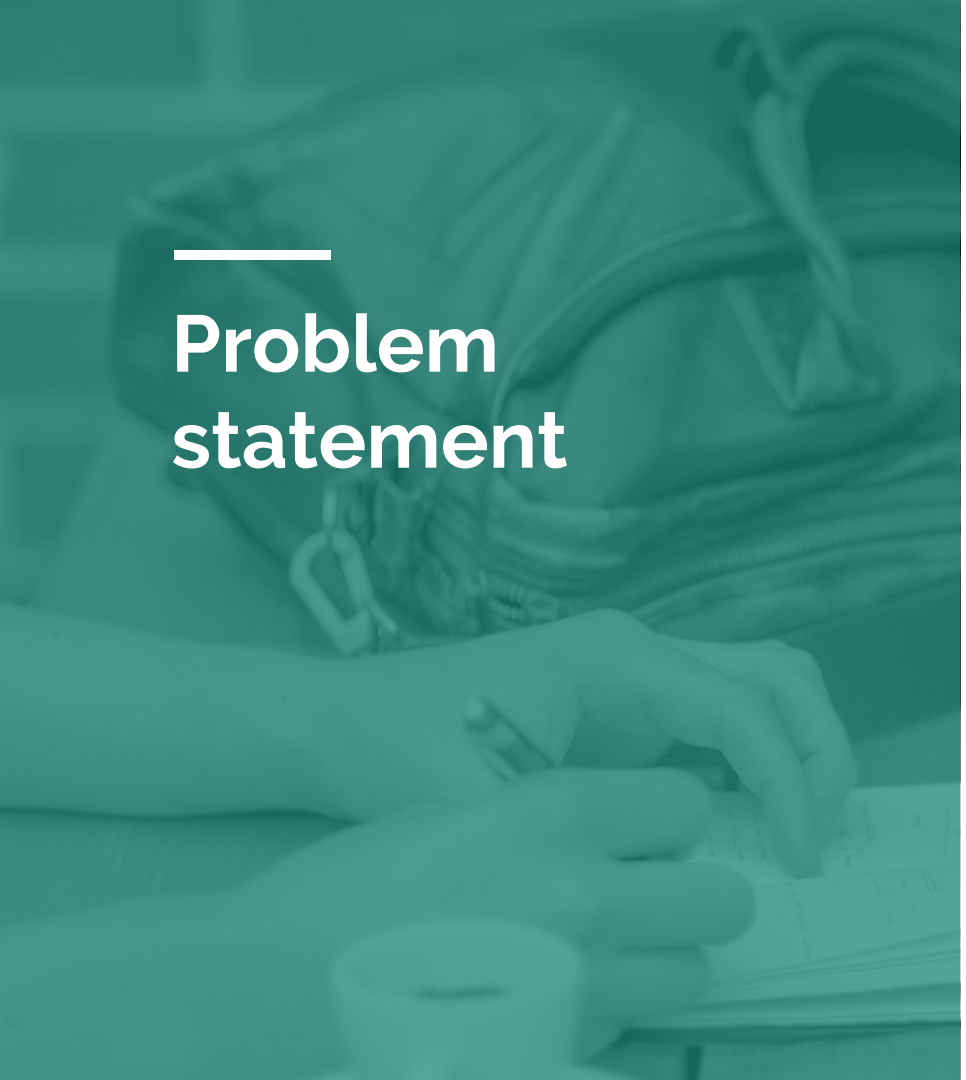
Emergency Messenger App

Mesh communication for
emergency and disaster scenarios

Outline

The Problem



Problem statement

In an emergency or disaster scenario, power outages and damaged infrastructure makes communication through mobile phones difficult, and in some cases impossible.

What people use today

- Walkie-talkies/radios
- Go low tech - yell, wave, shout, write signs.
- Use “offline” chat apps - Firechat, Zello, Bridgefy

Positioning Strategy

The emergency messenger app will be used to promote Right Mesh, alongside the other items in our portfolio of apps in a new Play Store account for RightMesh (not Left). It will be used as a consumer facing demo of the RightMesh platform.

This is a real way to help people - if we get it right, we could save lives.

Use cases / user stories

- As an office worker in a downtown highrise, I carry a phone all the time and generally always have a data or internet connection. However, I've been told the "big one" [earthquake] is due, and I just want to be able to keep safe when it happens. It would be chaos downtown, especially without internet. Can you imagine?
- As a person with some first aid training, I always hope I never have to use it, but there is a reason I took it in the first place - I want to be prepared. I would help anyone else I could if something were to happen though.
- As a mother with a teenager in a school nearby, I want to keep tabs on my daughter. This means knowing where she is and how she is. She's just 13 and I want to respect her freedom, but with all those shootings and things nowadays you can never be too careful.

Assumptions

- RightMesh technology and the app are reliable and stable and can support higher traffic volumes.
- RightMesh gets autonomous switching and broadcast messaging in the short terms.
- People will keep an app installed on their phone that they do not use frequently for safety.
- People would use this app over other traditional communication methods (like radios) in an emergency.

Solution Proposal

Solution description

Our Emergency Messenger app (name yet to be decided) allows people to communicate offline using the RightMesh SDK. Individuals can see other members on the network and their location, set a status to show if they need help, and have individual or group conversations. In a conversation text, voice, photo and location messages can be sent offline, allowing people to communicate in an emergency without the need internet or data.

Why it’s better than existing solutions

Bridgefy - no features or marketing that make it emergency specific, no audio or photo messaging.

Zello - though people think it’s offline, it’s actually not. You need internet or data.

Firechat - limited messaging only text and images. No location or audio messaging, no emergency specific features.

In summary, the RightMesh platform creates a more powerful messaging distribution service than our competitors. Furthermore, we have a unique positioning of being designed for emergencies, which will make us stand out amongst other messaging services.

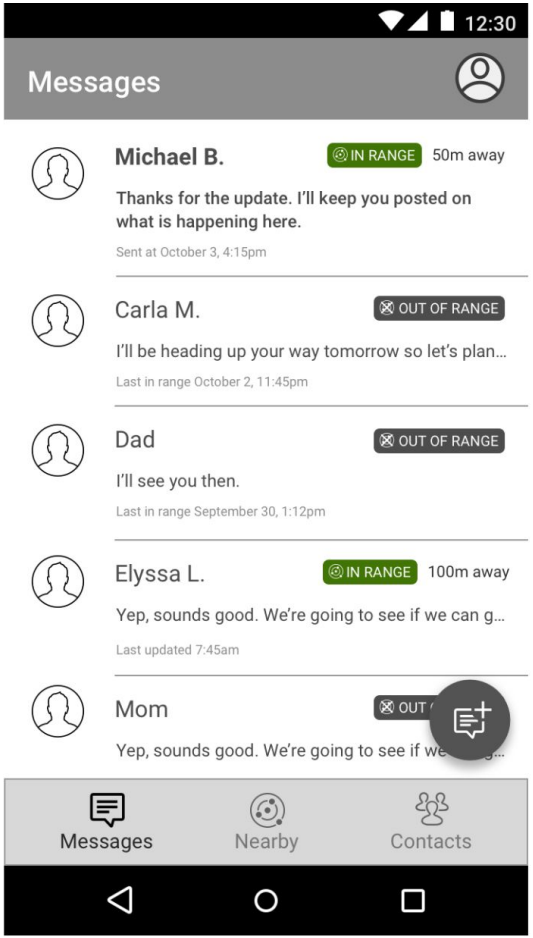
Wireframes

The whole project can be viewed at:
https://invis.io/KQM12139738WJB

Important! Wireframes in progress

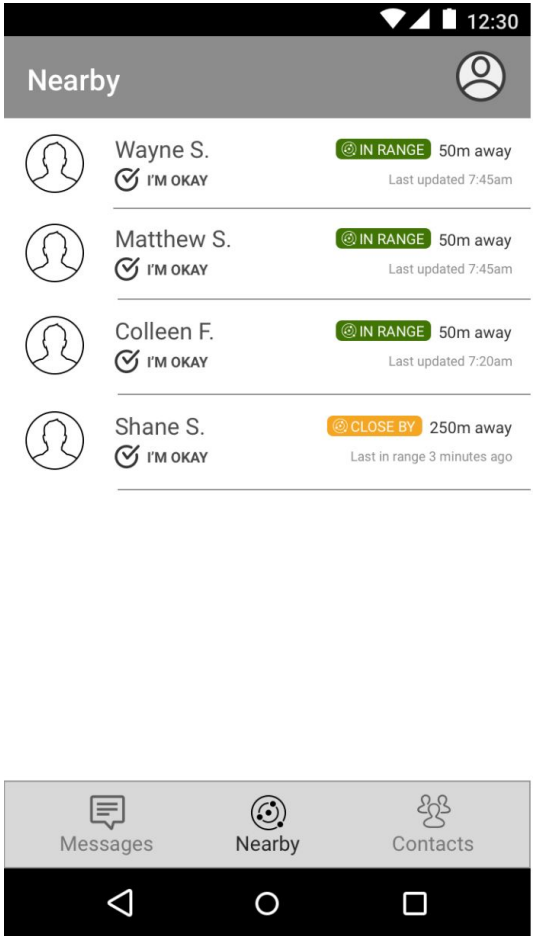
The following wireframe is an in progress version. Currently, this prototype needs to undergo at least one more round of Usability testing, so expect changes to follow. On top of this, it is completely unbranded - we need a name and a brand. So, if you are looking at this, take it with a grain of salt, so to say. Expect some of the layouts and language to change. The core functionality should not change significantly… everything here is the base for our MVP. Not all screens are included in this document, just high

Use the **invision link** to see the full latest version.



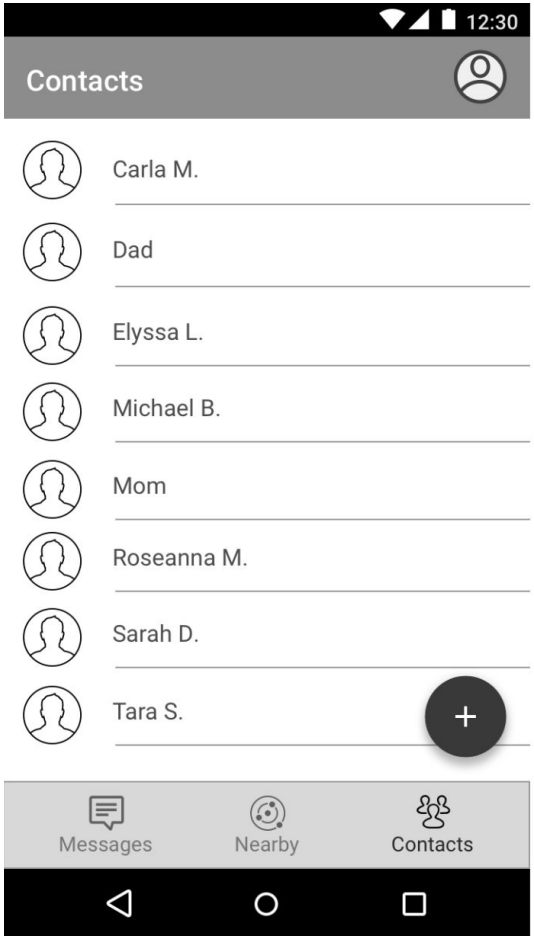
- Main home screen - messages page
- Unread and read messages distinguished
- Each conversation is marked with user status of “in range” or “out of range” showing whether they are available on the mesh.
- If in range, the approximate distance (if they are sharing their location) to you is shown.
- User profile and user name shown.
- Tap the FAB to create a new message
- App navigation done through tabs on the bottom.

Home - Messages Tab



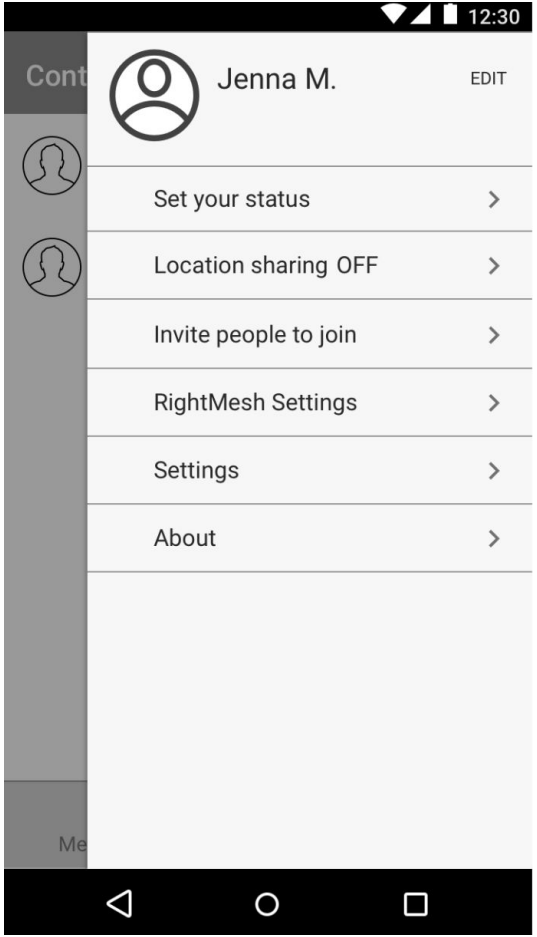
- Nearby shows all users who are within range of the mesh.
- Each user is marked with their profile image, their username, their status, and the distance from you. This also shows the last time this was updated.
- To manage spotty connections, a person whose connection drops is not instantly hidden from the nearby screen. They remain with a different status “close by” for a set period of time before they are removed entirely if they haven’t been seen.

Nearby Tab



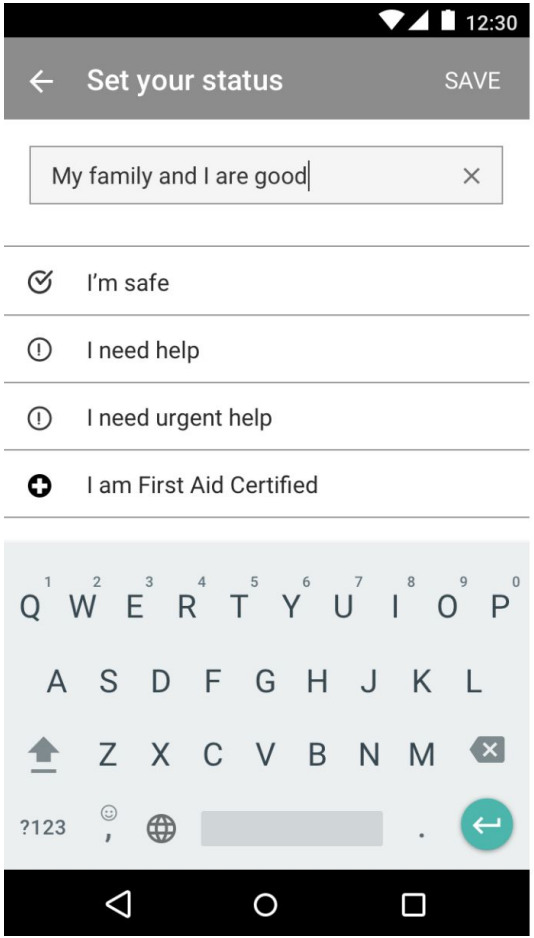
Contacts is an alphabetical list of people that you have saved over the mesh. Add contacts through the FAB.

Contacts Tab



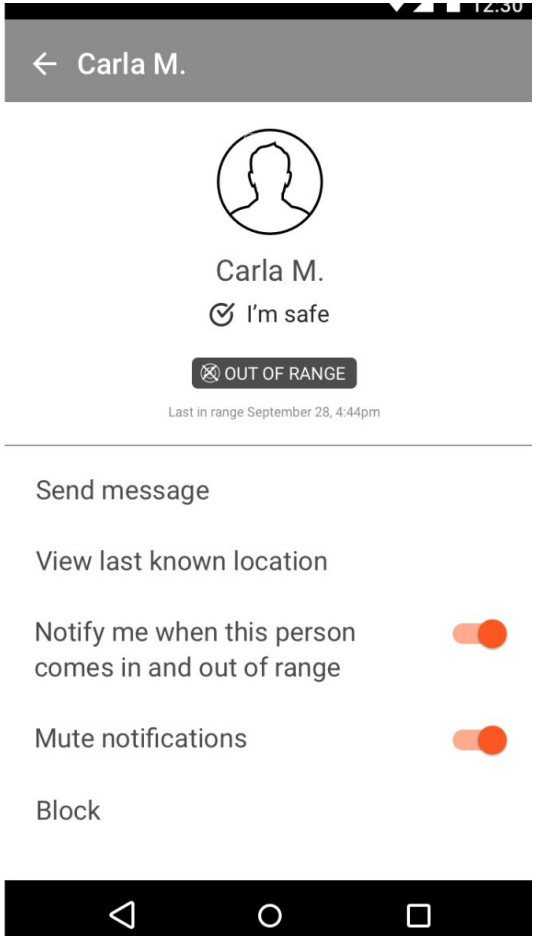
Accessible from all main tabs is your own settings menu. Manage your status, location sharing and RightMesh role from this menu. Swipe out from the right side, or tap your profile image to open.

Profile Menu



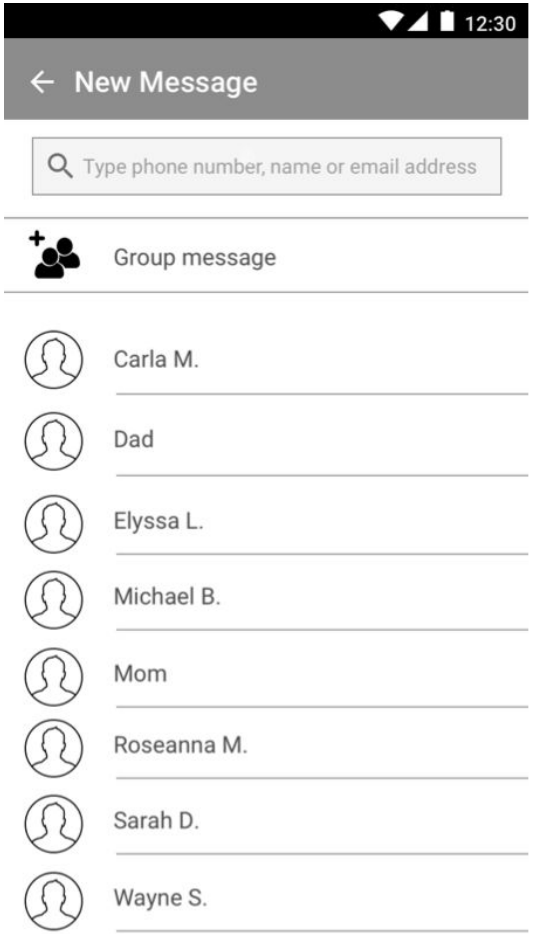
Tap an item in the Profile menu to be taken to an activity to edit it.
The Status includes a menu of predefined statuses, and the ability to set a custom one.

Edit Status



Other users on the mesh have profiles which can be accessed from a conversation or from the contacts list.

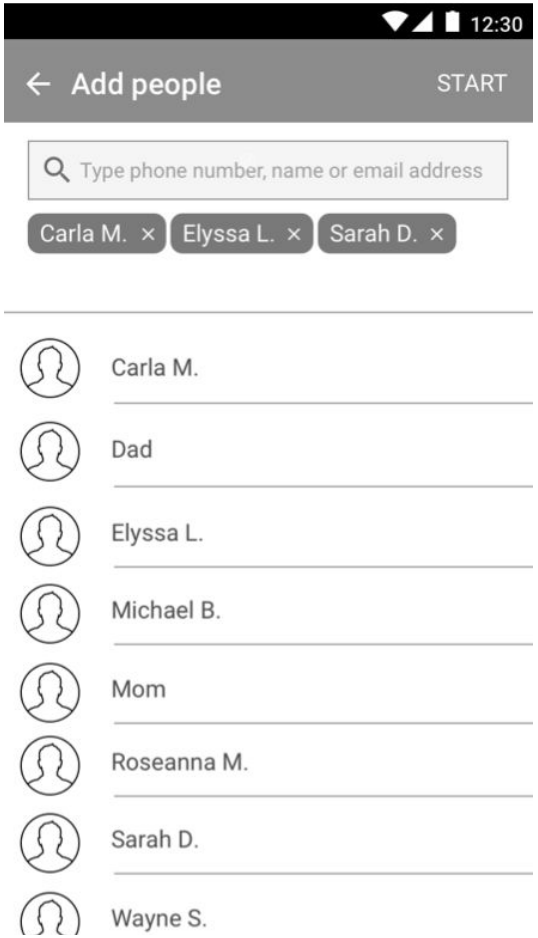
Other user profiles



When starting a new message (from FAB on Messages screen), the option is to create a private group conversation, or a 1:1 message.

Select from your contact list.

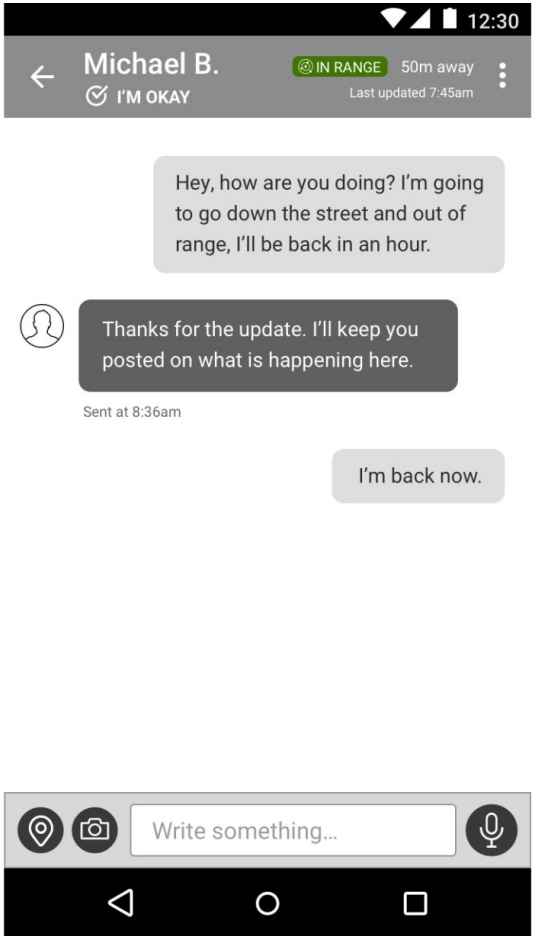
Create a New Message



Tap to select people to add to a group conversation. Tap their selected name again to remove.

To research:
Need to set a limit of the number of people that can be included in a group message.

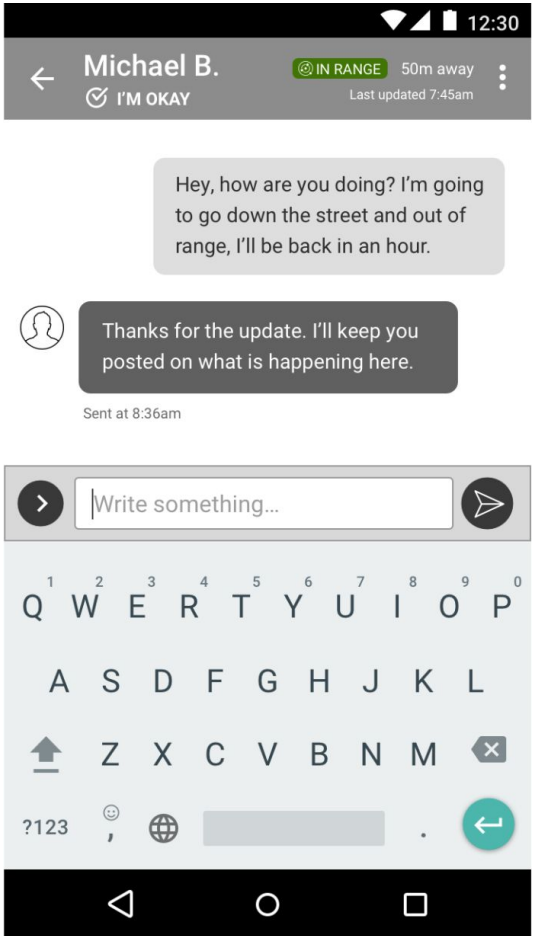
Create a new Group Message

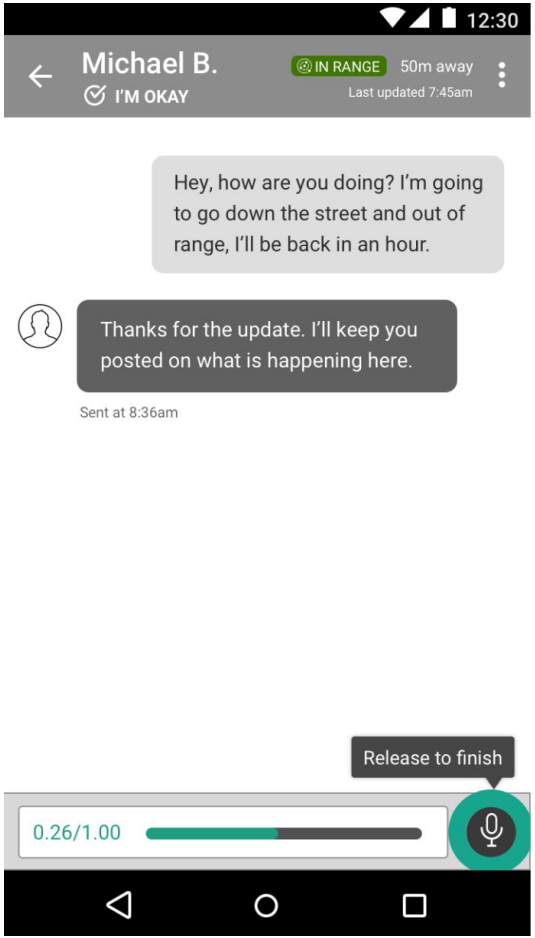


Conversations let you send text, voice messages (1:00), photos, and location.

To research:
Sending location will require a lightweight offline maps api integration.
Photos should not be sent at full size. Will need to compress and send (benchmarking to have files under 1mb before sending).
Audio is currently limited to 1:00, but will need to test file size and adjust.

Conversation window





Access to other types of messaging are hidden based what the user is doing at the current time.

Conversation window

Possible future features

- ➔ Broadcast or channel messaging - send messages to everyone on the mesh.
- ➔ Quick access widget/panic button.
- ➔ Save and share other personal medical information (allergies, blood types etc.).
- ➔ Map location of nearby resources (water, emergency shelter, food, medical etc..).
- ➔ Share the app without the internet

Other things to consider

- ➔ Keep as lightweight as possible to make sharing of the app easy
- ➔ We'll have to monitor battery usage, and try to keep this at a minimum (critical for emergencies).
- ➔ Mesh autonomous role switching has not been built yet - we will need clear explanations in the app of these roles in the meantime.
- ➔ File sizes for everything sent in a conversation should be kept under 1mb.
- ➔ Limit the exchange of user data - not a constant check of user location and status - set an interval
    - ◆ Initial discovery of nearby should send package of user data - profile picture and username. This is cached and the same image and info is used everywhere in the app.

Next Steps

Review, plan and collaborate

- ➔ Review wireframe through invision. Comment on the prototype there any questions or feedback.
- ➔ We will have a call on Tuesday night 9:30pm CAD, Wednesday morning 10:30am to go through the wireframe and the project plan in detail as a team.
  - ◆ Discussion of questions and overview
  - ◆ Developer review and workflow.
  - ◆ Design timeline and project timeline.

What's next in the design process?

- ➔ Perform usability tests with people in and outside of the office.
- ➔ Revise this wireframe based on testing results.
- ➔ Get developer feedback and make adjustments to wireframe
- ➔ Decide on name, and create brand and logo.
- ➔ Create high fidelity wireframe with applied branding.
- ➔ Cover all edge cases and onboarding.

Questions?