Speed Force

Oscar & Akanksha

Purpose

Speed Force can detect a crash using your smartphone sensors and inform your designated emergency contact person that a possible crash has occurred. While similar tools exist, to our knowledge the nearby contacts feature is not included.

Speed Force will send a prompt to nearby users informing them of a possible accident nearby and ask them if they are willing to assist, in addition to the emergency contact.

This tool is meant to be used in a community setting to support each other and ensure immediate emergency response. Maybe Speed Force can save a life at a busy intersection with local businesses, or while driving to get groceries for loved ones.

We are not the first to have this idea:

Papers

WreckWatch: Automatic Traffic Accident Detection & Notification with Smartphones

Jules White, Chris Thompson, Hamilton Turner, Brian Dougherty, and Douglas C. Schmidt

Car Accident Detection and Notification System Using Smartphone

Hamid M. Ali, Zainab S. Alwan

Apps

http://www.sosmartapp.com/
https://www.noonlight.com/
https://www.carlock.co/

Why are we different?

We use the nearby connection features both in Android and iOS as a key social aspect. We want the community to be engaged. When an accident occurs not only are the user and user’s emergency contact person (ECP) contacted but nearby users will be alerted of a possible accident or unsafe conditions.

Sometimes as a bystander you feel helpless, but with this app you can contribute by allowing the ECP to contact your phone if the hurt user’s phone happens to be damaged. Noticed a car or someone fleeing the scene? take a picture and possibly solve a hit and run!

Detection Phase

Detection start at 24Kph =14.9 Mph

- This assumes you/phone are in motion and not Usain Bolt who's world record is 23.35 mph

Acceleration values greater than or equal to 4G

- minimizes false positives from dropped phones

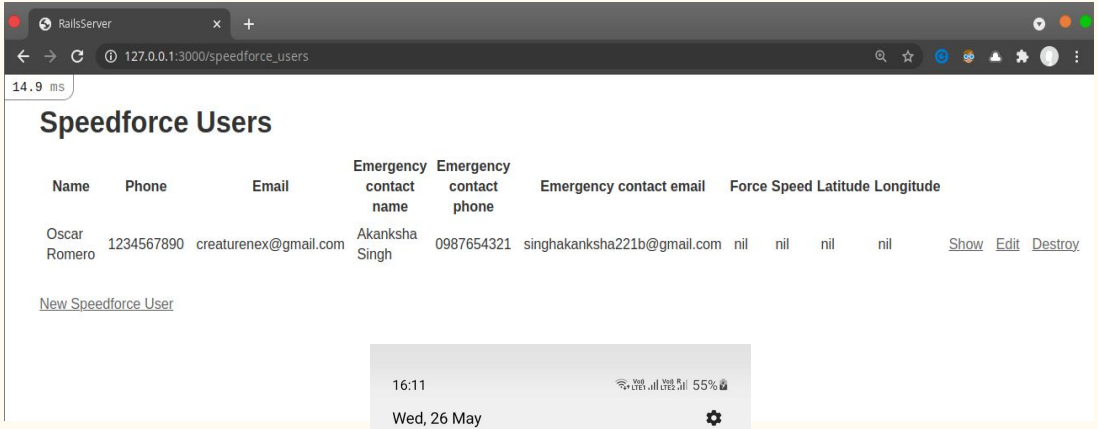
Both values must be met. **other sensors used but limited for project**

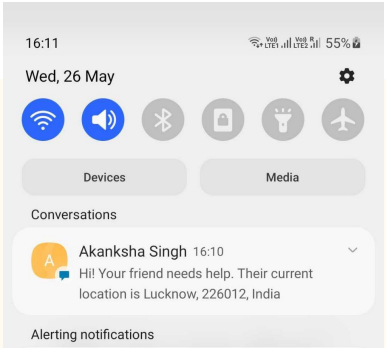
Values derived from research papers and are for a person in a car at constant velocity, but research discusses potential application for people outside of car ie pedestrians or cars traveling at slower speeds ie city intersections with constant starting/stopping.

Collection and Transmission Phase

Send relevant data to server to be incorporated into database

- Name
- Phone
- Email
- ECP Name
- ECP phone
- ECP email
- Speed
- Force
- Latitude
- Longitude





Access relevant from database to alert

- User
- Emergency Contact
- Nearby Users

Termination Phase

Everything OK

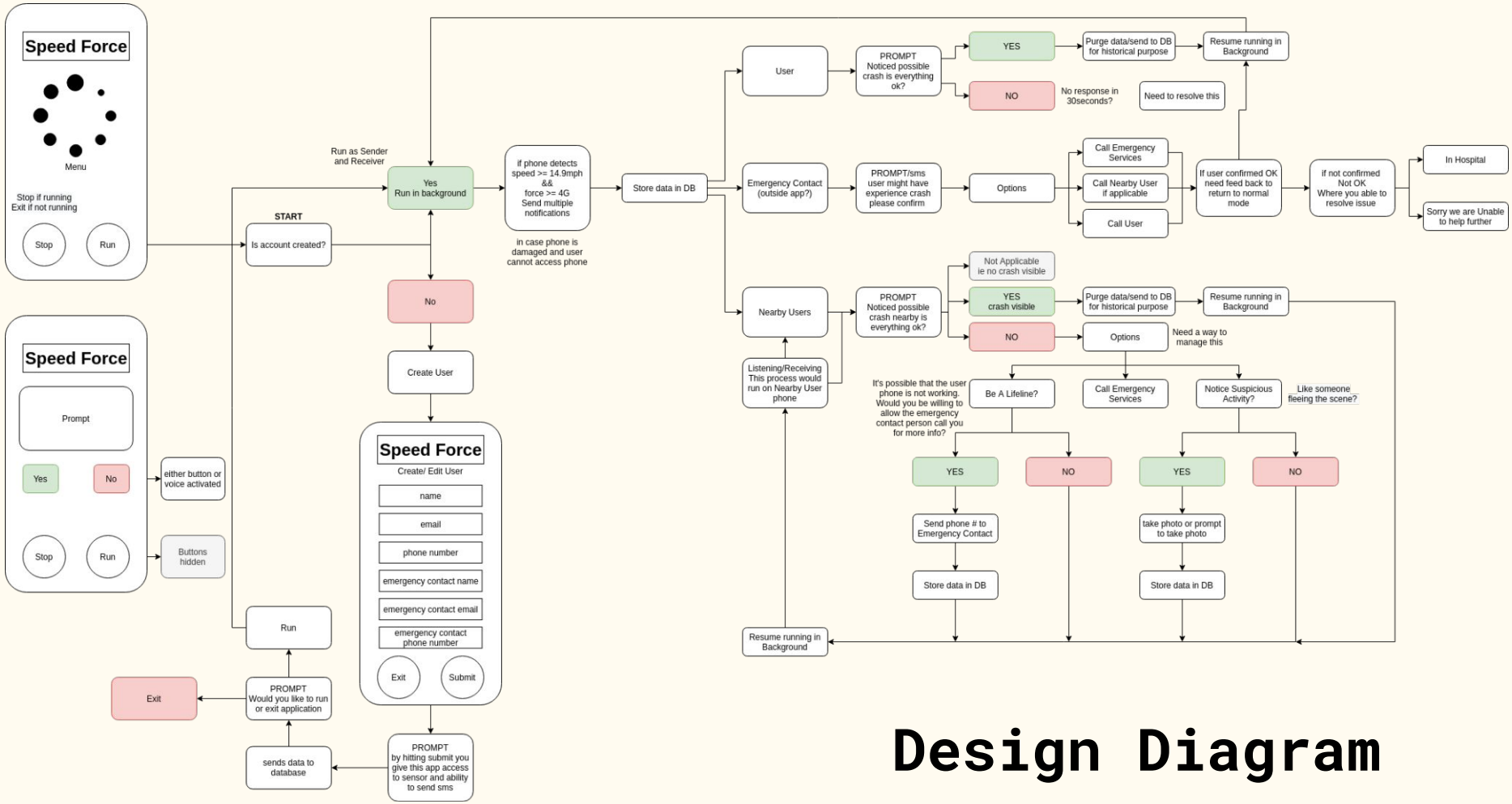
- User or Emergency Contact confirm everything ok and app returns to listening mode.

Everything NOT OK

- User or Emergency Contact reach Emergency Personnel for Help.
- If no solution is made after X time. We are sorry no further data to assist.

For Nearby User

- If no accident noticed, returns to listening mode
- If accident noticed
  - Willing to be point of contact for Emergency Contact
  - Notice suspicious behavior? Take photo!
  - Return to listening mode



Design Diagram

What was accomplished?

Design Diagram to focus design choices

Designed a user interface demo using Flutter and Android

- Incorporated multiple plugin for the relevant sensors
  - geolocator 7.0.3 - GPS
  - sensors 2.0.1 - Accelerometer & Gyroscope
  - flutter nearby connections 1.0.11
  - flutter sms 2.3.1

Constructed a Ruby on Rails Server with a database to capture user data

- Had to learn Rails :)

Presentation Material

What was not accomplished and why?

Not able to interface the mobile UI/UX with the backend server.
- Neither of us has had this experience before.

Not able to incorporate New Relic dashboard.
- First we did not have a completed project to test against
- What metrics are important? Ie connection speed, transmission speed or dropped connections etc

Not sure of the best way to model nearby user for database purpose
- new NearbyUser model or use User model and add more parameters

The idea and scope of this project is large. But we choose Hack for Good because we believe strongly in using technology to help instead of hinder. We are a two person team created last minute who both have limited experience with this type of project. We learn a lot quickly but time still ran out :)

Future Improvements?

Build a robust algorithm so this can also be used for pedestrians.

Build better social tools, along the lines of Waze

Build a Web Application to visualize data. Did someone say New Relic Agent?

Incorporate Twitter somehow ie post accidents to Department of Transportation showing dangerous intersection hopefully build traction for municipal improvement.

Biggest issue is how to entice users and pay for servers.

For Danny and Lovisa

And The Relicans