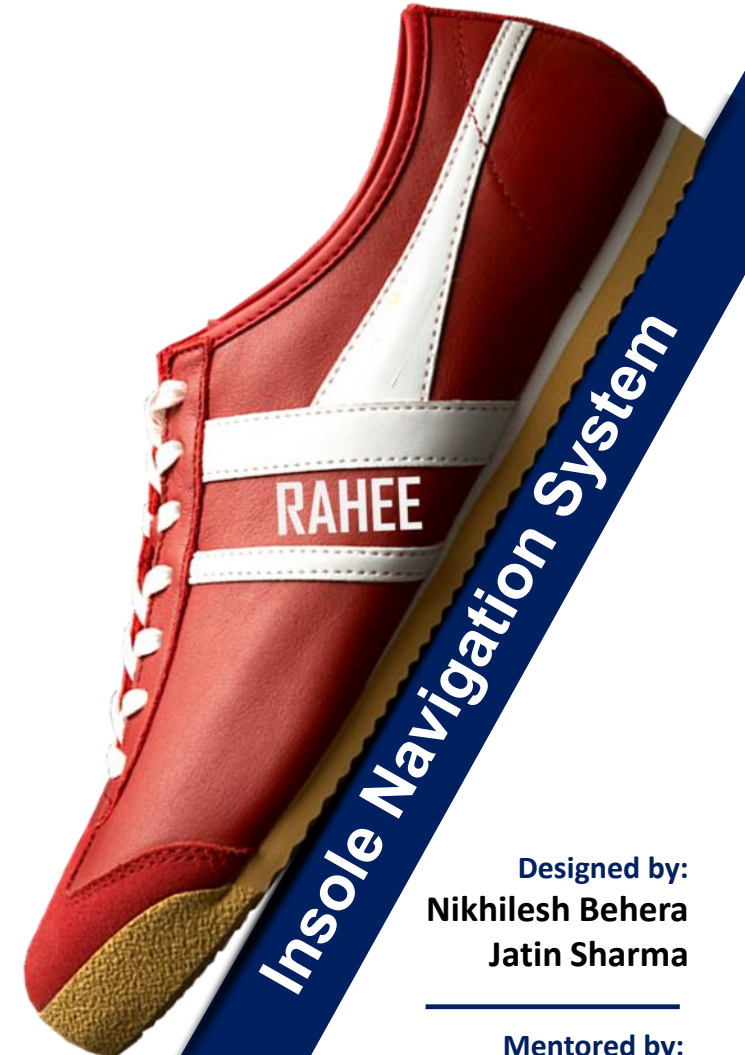




- 285 million Visually Impaired
- Nearly 90% live in developing countries
- Walking canes just provide basic cues
- Affordability vs Employability
- Rahee attempts aiding cane with Haptic cues

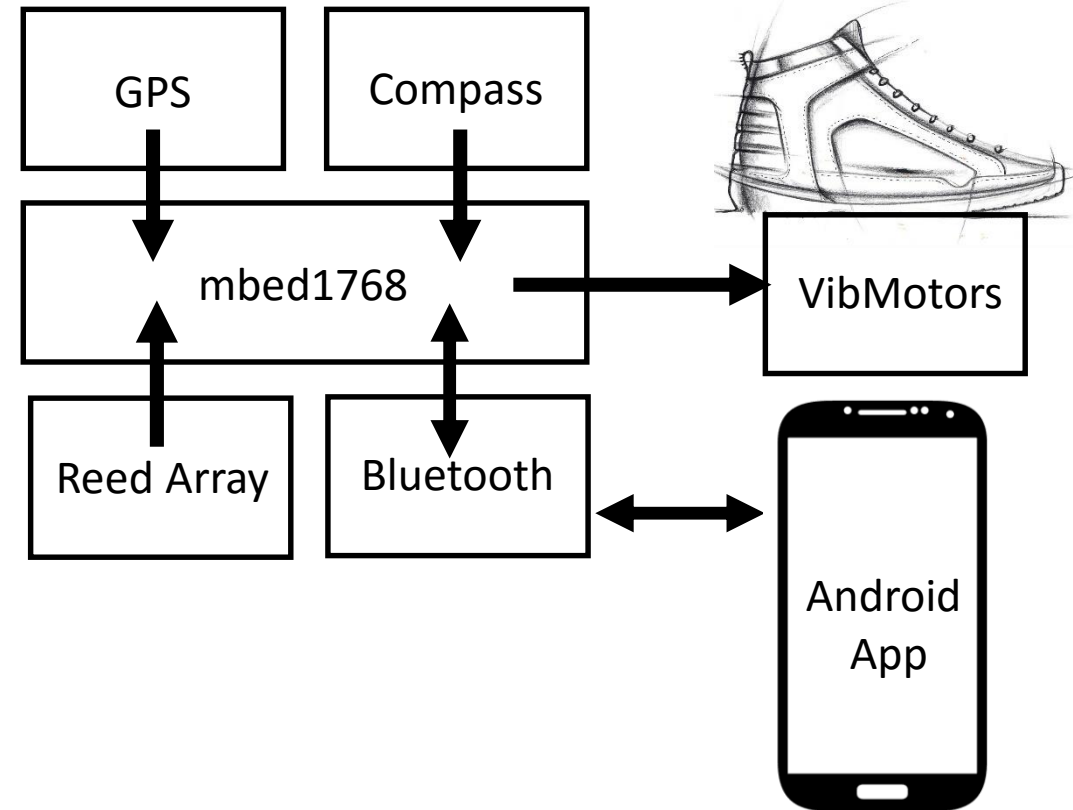


Designed by:
Nikhilesh Behera
Jatin Sharma

Mentored by:
Prof. Rahul Mangharam

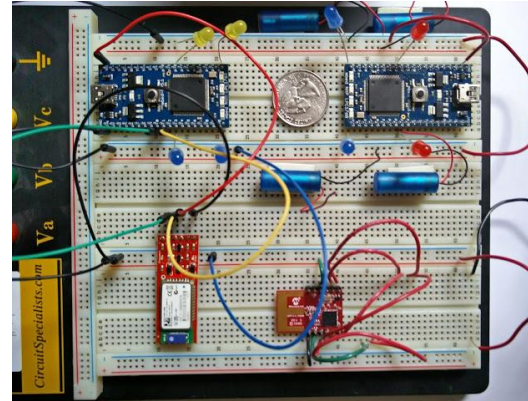
System Diagram

- **Mbed receives current coordinate from GPS and direction from Compass at all time.**
- **When user requests for navigation, Android app queries OpenStreetMaps to get a routeList.**
- **Mbed utilized Remote Procedure Call (RPC) over Bluetooth backhaul to invoke mbed's RTOS functions to execute PWM.**
- **Mbed listens to ReedArray interrupt for new hotspots and add them to list.**

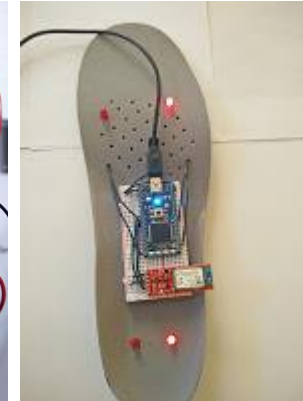


Hardware/Mechanical Effort

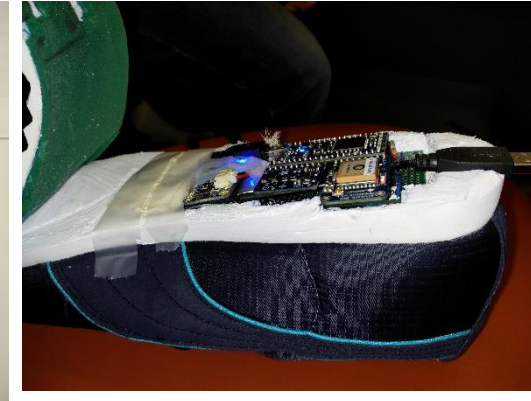
- Sensor Interfacing (GPS, 9-DOF, Flex, Reed, Bluetooth, Vibe, Boost)
- PCB Design (Compact, 2-level circuitry)
- Soldering
- Cutting shoe
- Haptics
- Sensor placement/Pinout restrictions
- Weight balancing
- Recharging and Programming



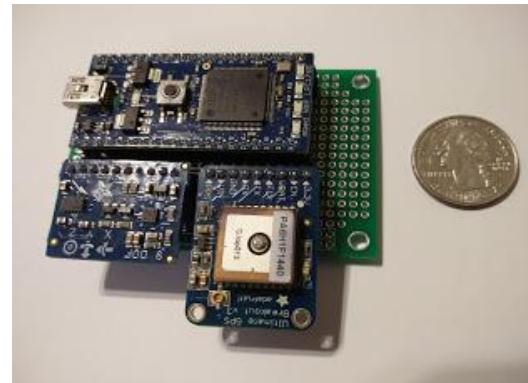
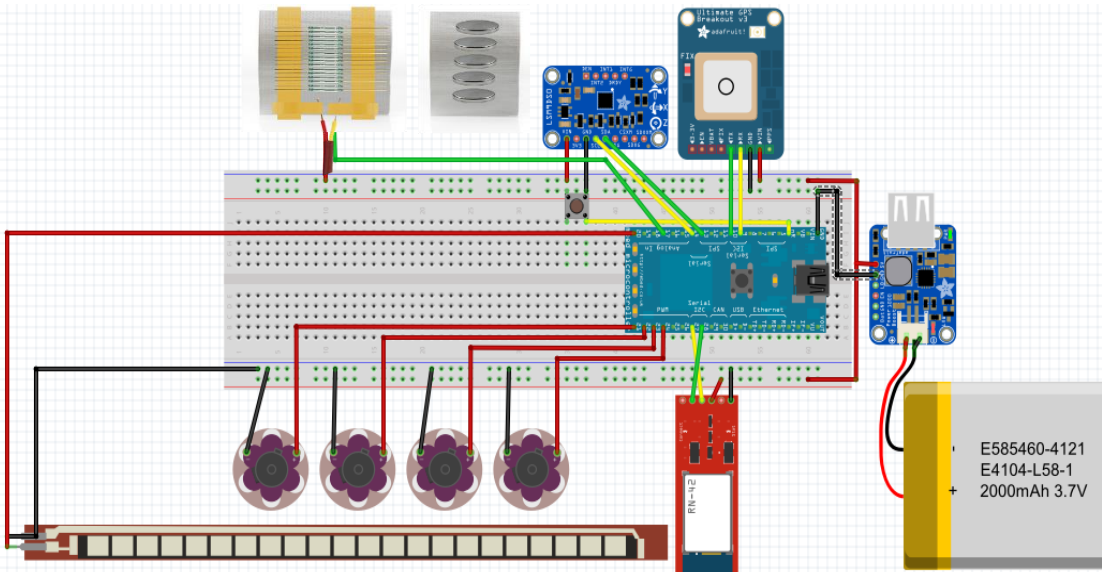
RaheeTestbed



RaheeSole



RaheeShoe



Main Circuit



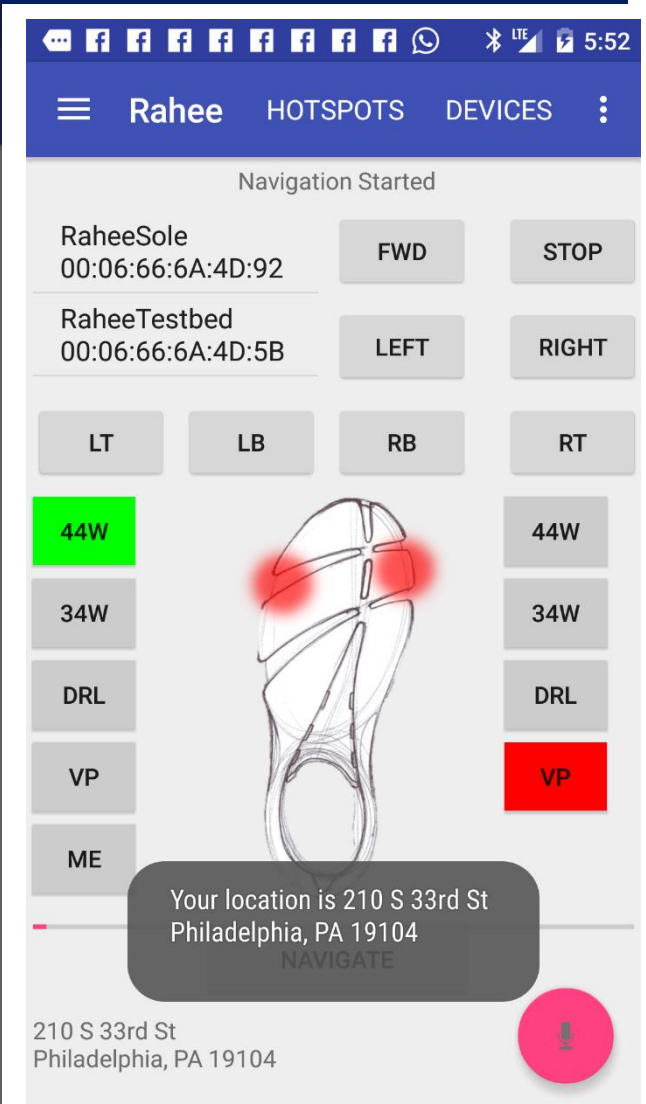
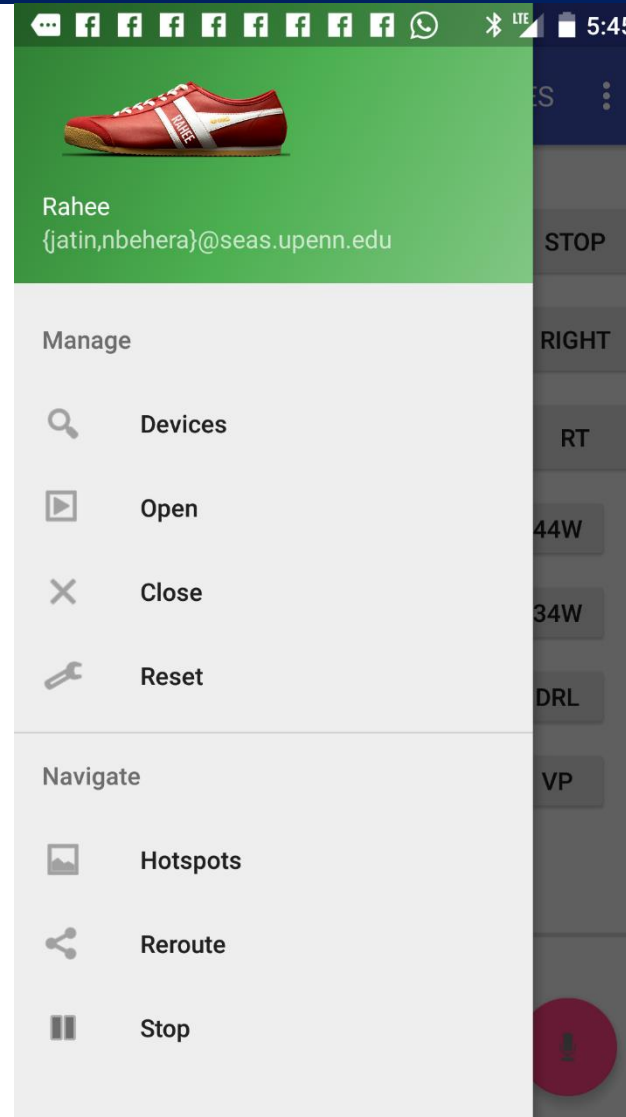
Packing



WalkReady

Software Effort

- Offline Maps storage, rendering and marking
 - XML Parsing
 - Navigation (JumpList, Segments, Haversine)
 - Direction (heading, bearing, margin)
 - Hotspot Marking and Detection
 - Bluetooth Communication over Serial port
 - Remote Procedure Call
 - Voice Recognition and Voice Feedback
 - Android Programming
 - ShoeSimulation
 - Debugging over Bluetooth
 - Source and Destination Rerouting
 - Interactive UI
-
- RTOS
 - C/C++, Java
 - I2C
 - Multithreading
 - ISR – Interrupt handling



Why is Rahee awesome?

Technically

- RPC on mbed
- Offline Map rendering
- Navigation
- Wall Charging
- Hotspot Marking
- Voice Recognition
- Debugging over Bluetooth
- Real-time Emulator

Performance

WalkTime : 9 hrs -10 hrs
Weight : 350g (shoe), 60g (main ckt)
X-Factor : Style symbol, Cool
Usability : Working prototype, comfortable

