Rid3

Forever Akpabio, Emmanuel Obu, **Akintayo Salu**

Use Case

- **Problem** : Current navigation options for bicycle riders can be distracting and lack essential safety features.
- **Solution: Safely navigate places** on bikes through **audio instructions** and vibration feedback for **blind spots**

Requirements:

- Blind spot detection: **95% accuracy rate**
- **Vibrational cue** within **1 sec** of object detected in user’s blindspot (less than 5 feet)
- User should **receive audio instructions to make turns within 200-300 feet before a turn**.

Design Requirements

Wristband

- **Battery life** (~5 hours)
    - 500 mAh Battery
    - Micro arduino (50mA) + Bluetooth module (30mA) + vibration motor (10mA) = 90mA
    - 500/90 = 5.5 hours
- **Weight** (~230 grams)
    - Micro Arduino (13g) + Vibration motor (1g) + Bluetooth module (2g) + Mini breadboard(13g)
    - 3D printed plastic encasing (180-200g)
- **Ease of usability**
    - Different vibrations for blind spot detection vs veering off intended route

Design Requirements

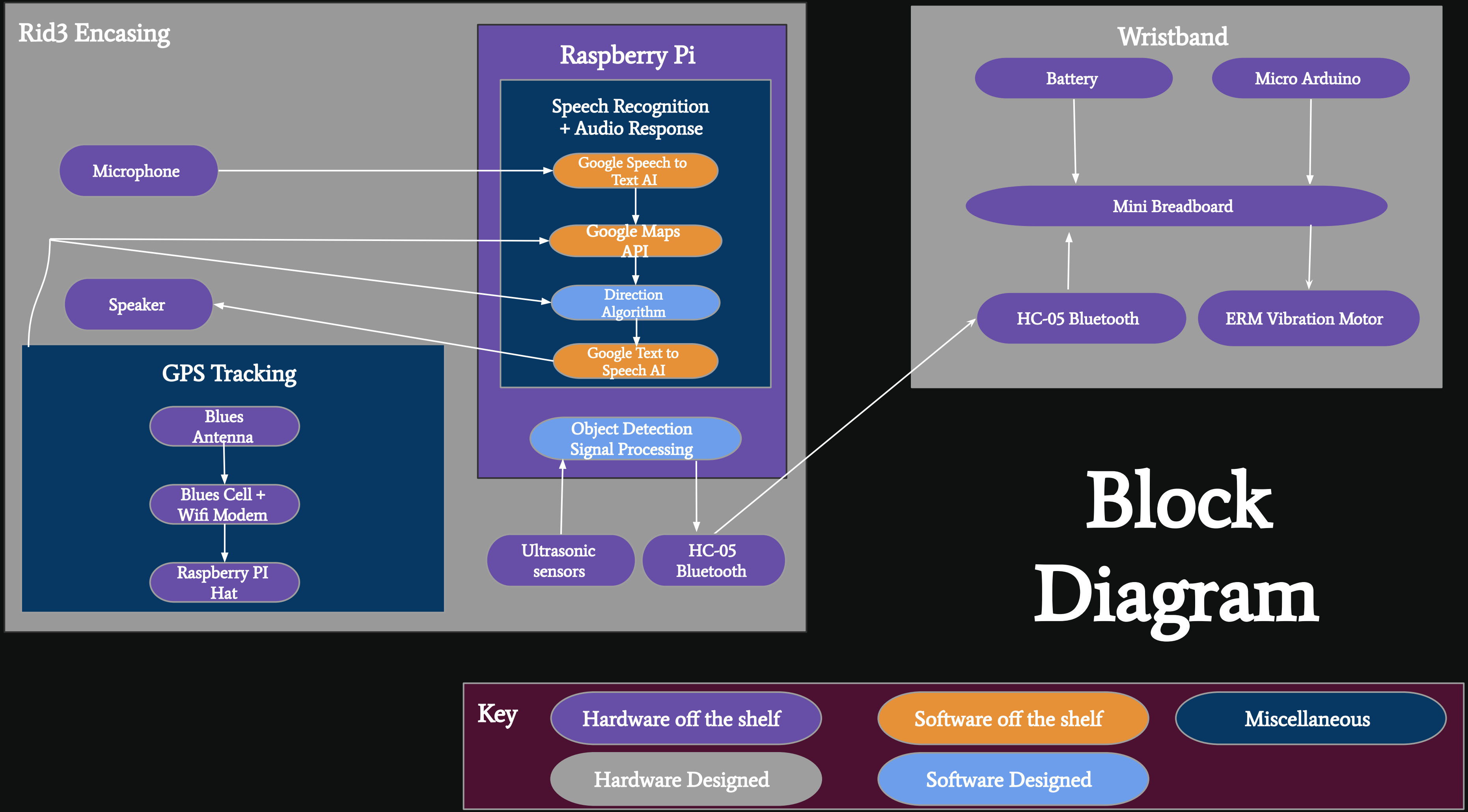
Navigation Device

- **Easily attachable**
  - 5x5x4 inch case with velcro strap, attach to pole underneath bike seat
- **Low Latency**
  - 1-2 seconds to give instructions based on new updated GPS location
  - 1 second from object detected in sensors to vibration in wristband
  - Bluetooth 38400/9600 baud rate and 100ms latency
- **Battery life** (~10+ hours)
  - 10 kA battery, bluetooth module (30mA), RPI 4 (600mA -1A), Sensors 8mA
- **Accurate detection**
  - Sensor range 3~450cm and FOV 30-45 degrees
- **Precise real time navigation (90% accuracy)**
  - Blues starter kit GPS tracking within 5-10 meters with proper connection.
  - Google Maps API estimated to have 90%+ accuracy
  - Using Google Speech To Text and Text To Speech for speech system, both are 90%+ accuracy.

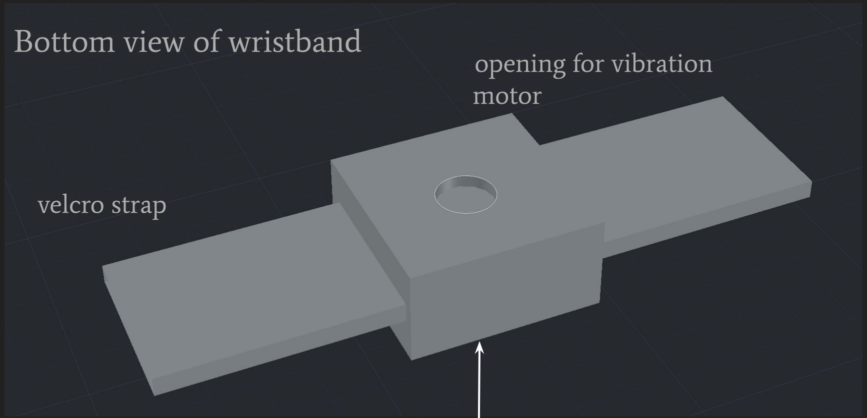
Solution Approach

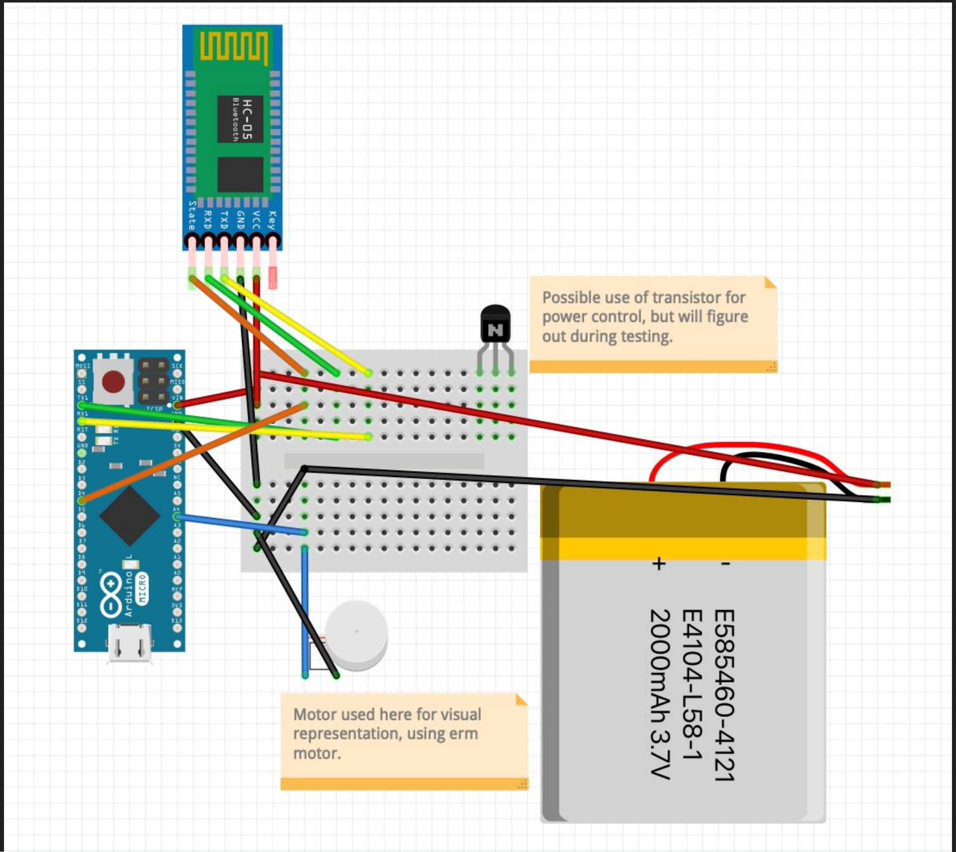
Solution Feature	Impact	Societal Concerns
Audio Input for designated location at start of journey.	- Simplifies process of starting journey and makes it hands-off.	- People have different vocal intonations. This concerned is addressed by google voice recognition.
Real time GPS tracking with audio output for direction correctness.	- Hands off - Allows user to focus on the direction that they are going.	- Audio cues can be muffled by outside noise.
Ultrasonic sensors for detecting objects.	- Extra safety on the roads for bicycle users, and wider range for detection.	- A lot of dependence on sensors catching correct angles of objects in blind spots, so continuous testing with different ranges of objects, speeds, and angles is needed.
If object detected, send ping to haptic feedback wristband that sends a light nudge to the user while riding.	- No need to check blind spots by turning head. - Subtle but easy to notice.	- Strength of Vibration could be distracting to users, so continuous testing of different vibration modes w/ user feedback

Block Diagram

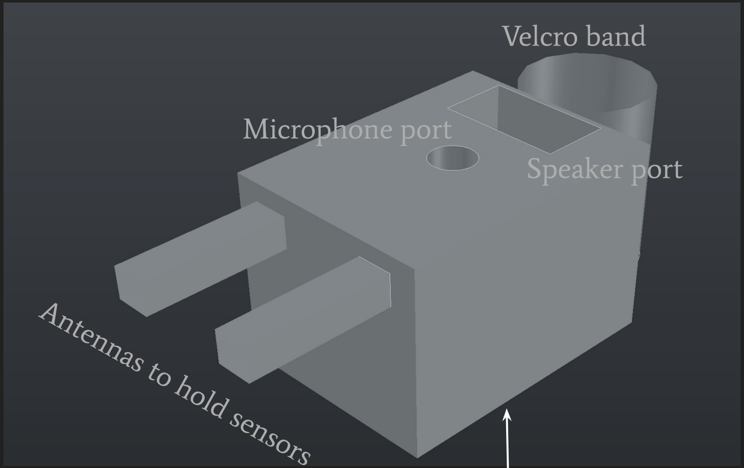


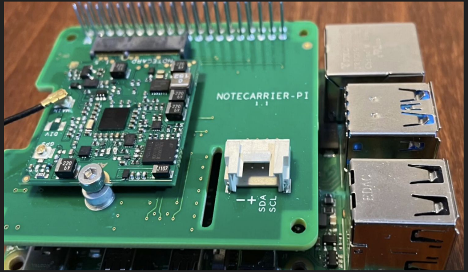
Wristband


Bottom view of wristband
opening for vibration motor
velcro strap

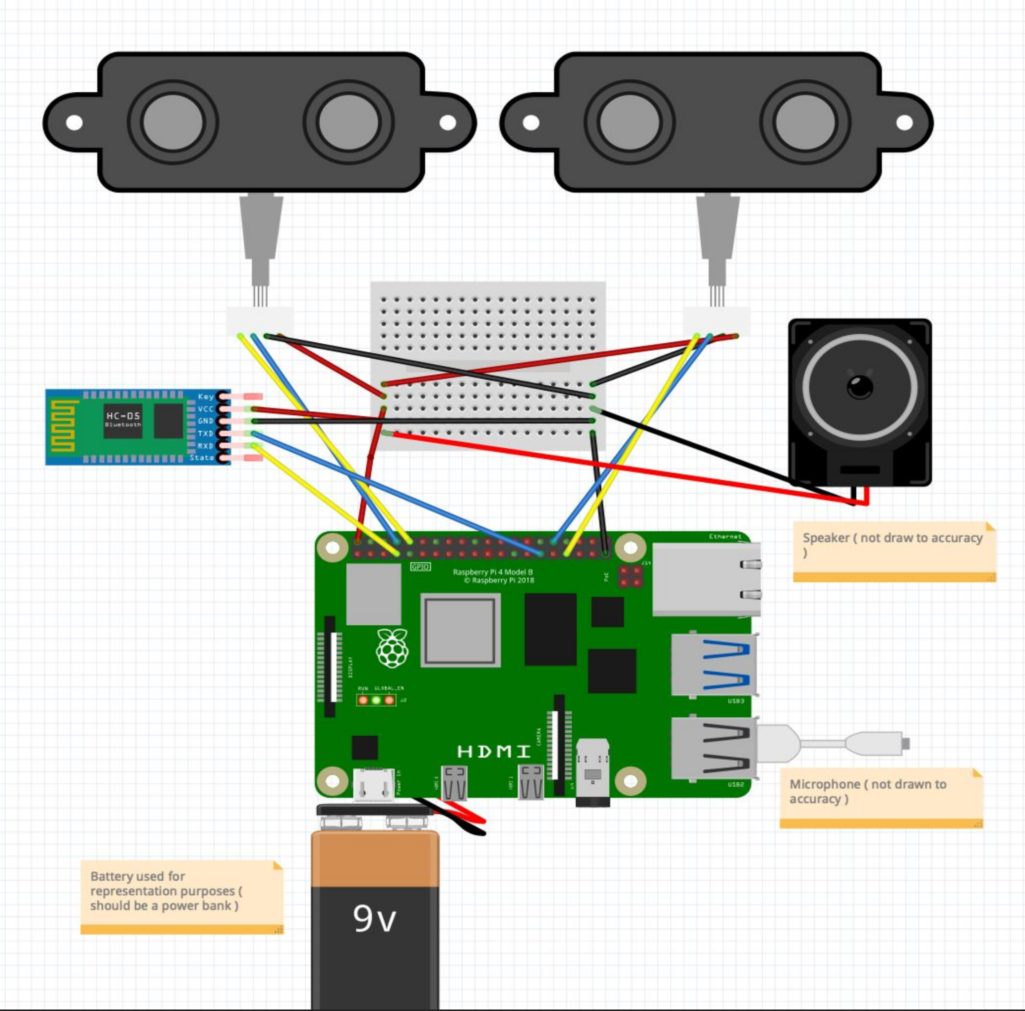

HC-05
Bluetooth
State
RXD
TXD
GND
VCC
Key
Possible use of transistor for power control, but will figure out during testing.
Arduino
MICRO
Motor used here for visual representation, using erm motor.
+
-
E585460-4121
E4104-L58-1
2000mAh 3.7V

Rid3 Encasing





Raspberry PI w/ modem and
Carrier hat connected



Pinout scheme for Raspberry Pi

Navigation + Speech System

1. Receive audio from user (containing desired location)
2. Convert audio to text using the Google Speech to Text AI system
3. Translate the destination to its longitude and latitude representation using Google Maps Geocoding API
4. Make Google Maps Routes API call to retrieve route for entire journey
5. Use the user's current GPS location to find the next turn in the journey
6. Convert instruction to audio
7. Send audio of instruction to external speaker

```python
import requests
api_key = open('ki.txt').read().strip()

url = '''https://maps.googleapis.com/maps/api/directions/
json?destination=DEST&mode=MODE&origin=ORIG&key=API_KEY'''

mode = 'bicyle'
orig = '40.447888036901304,-79.94648296363464'
destination = '40.44411653616266,-79.94209498494072'

url = url.replace('DEST', destination)
url = url.replace('MODE', mode)
url = url.replace('ORIG', orig)
url = url.replace('API_KEY', api_key)
```

Google Maps API Call

```
@asaluedu → /workspaces/rid3 (main) $ python3 maps.py
{'lat': 40.4479319, 'lng': -79.94546179999999}
Head <b>east</b> toward <b>Clyde St</b>

{'lat': 40.4489109, 'lng': -79.9466932}
Turn <b>left</b> onto <b>Clyde St</b>

{'lat': 40.4508856, 'lng': -79.9430531}
Turn <b>right</b> onto <b>Ellsworth Ave</b>

{'lat': 40.4446254, 'lng': -79.9430272}
Turn <b>right</b> onto <b>Morewood Ave</b>

{'lat': 40.4444945, 'lng': -79.9421004}
Turn <b>left</b> onto <b>Forbes Ave</b>
```

Google Maps API Return

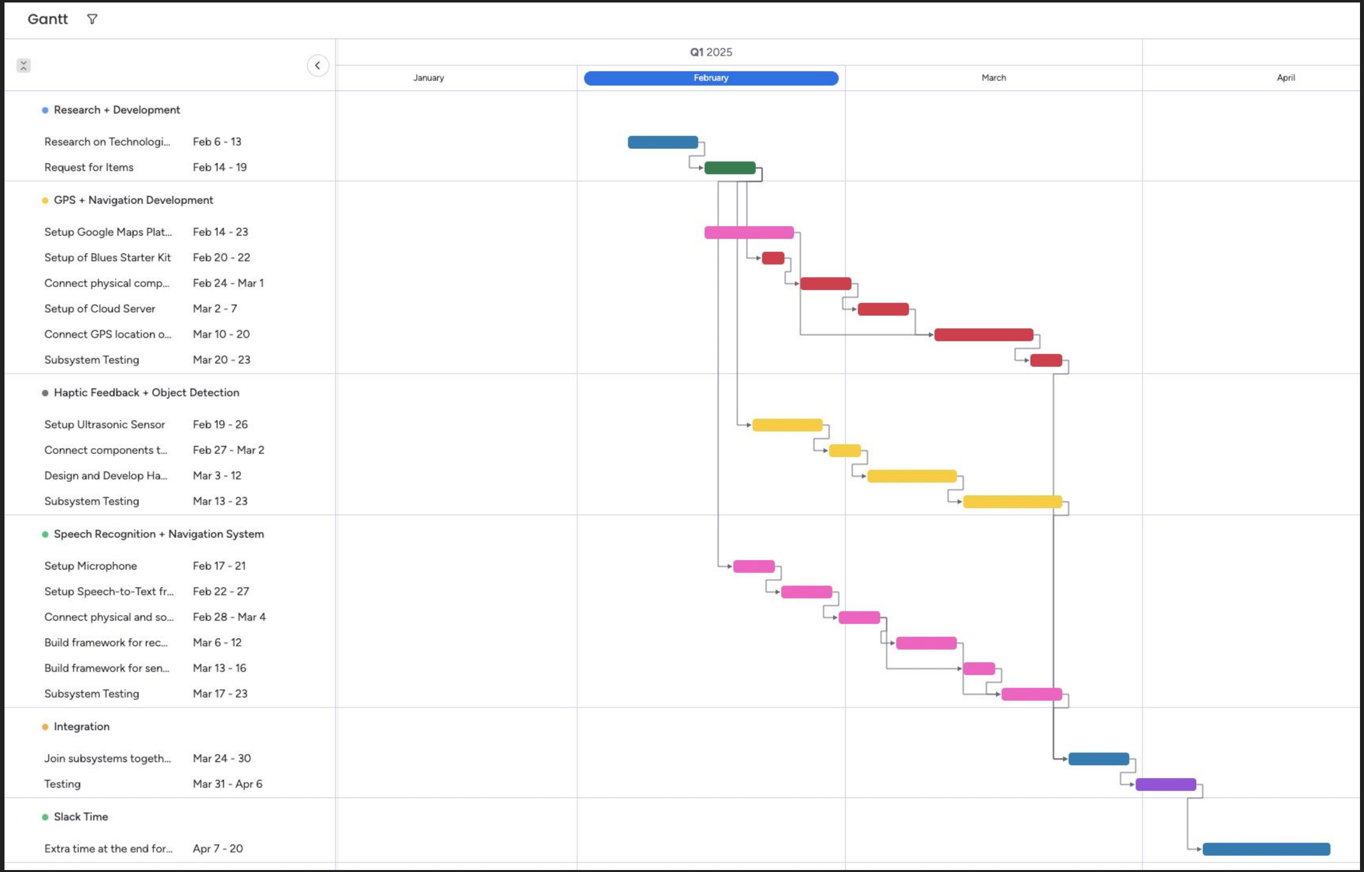
Test, Verification and Validation

Use Case Requirements	Test Specifics	Test Inputs	Test Outputs	Validation
95% accuracy rate for detecting object in blind spot	100+ tests with different obstacles (e.g. other bikers, drivers, etc.)	Object distance from ultrasonic pulse (from sensor)	Signal for object within 5 feet of the user	A valid test is when signal is correctly initiated when object distance is within 5 feet of the user.
Vibrational cue within **1 sec** of object detection	100+ tests where object detection occurs	Signal from object detection	Haptic feedback on wristband	A valid test is when latency between the signal and haptic feedback is less than 1 second.

Test, Verification and Validation

Use Case Requirements	Test Specifics	Test Inputs	Test Outputs	Validation
Receive audio instructions to make turns within **200-300 feet prior to turn**	20 different routes are tested	GPS location of user	Audio instruction for next turn	A valid test occurs when the correct audio instruction is received within 200-300 feet before the actual turn
90% translation accuracy for extracting destination from user's voice command	50+ tests with different destination commands said by different voice types	Audio of user's command	Speech-to-text output	A valid test is when the speech-to-text output from the user's command matches the actual desired destination for journey

Project Management



Emmanuel

- Object Detection system
- Haptic Feedback system

Forever

- GPS Tracking System
- RPi Integration

Akintayo

- Speech recognition System
- (Audio) Navigation Response System