INDOOR TRACKING: EQUIPMENT AT CHOA

Brian Weber
Alex Sabulski
Zach Davis

INTRODUCTION

- Indoor tracking of equipment within the hospital
  - 1 floor
  - 1 wing of the hospital
- Provide indicators for required maintenance
  - Include a way to reset maintenance alerts
- View tracked devices within an iOS application

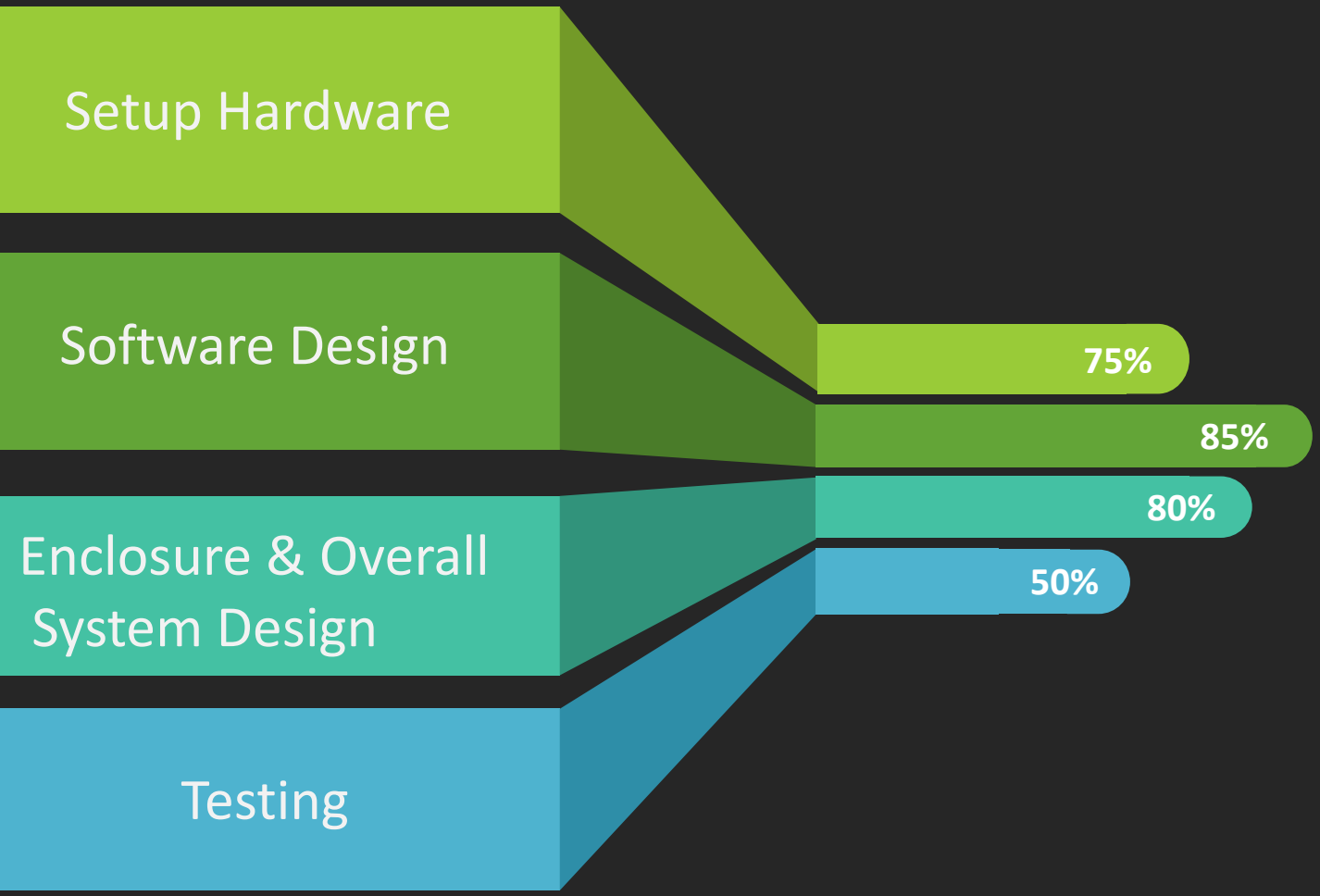
SPECIFICATIONS

- Localization with minimal error
  - 3 meters max; as small as possible
- Tracking over the entire hospital wing (100 m x 100 m area)
- The system must be portable for use with new equipment
- Maintenance requests should be automated
- Localization data must be provided visually to the user

DESIGN

- Implementing the Pozyx ultra-wideband communication system
- Software:
  - iOS application, Google Cloud Platform, MySQL DB
- Hardware:
  - Raspberry Pi Zero W
  - Pozyx shield tag
  - 4 Pozyx anchors
  - Button to reset the maintenance timer
  - LEDs
  - Rechargeable USB power supply
- Description:
  - Raspberry Pi receives 2-D coordinates in mm accuracy using Pozyx positioning shield. Shield ID is uploaded to MySQL database with coordinates and other useful information
  - Red LED indicates that maintenance is needed; Green LED indicates use
  - iOS application receives database information from Google cloud and shows location for equipment on hosptial map
    - With usage info and maintenance time

IMPLEMENTATION: OVERALL



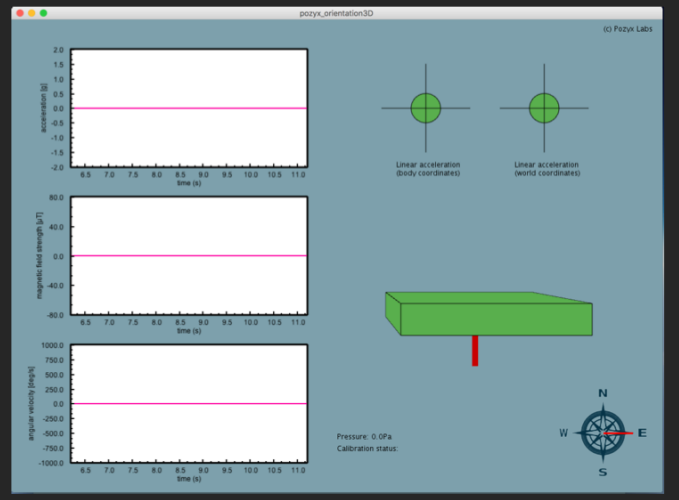
- Setup hardware in our test environment
  - Must test in the hospital
- Implemented the Pozyx Python library with our MySQL database
  - Need to finish visualization within the iOS app
- We know the general design of our tracking device
  - Need to figure out dimensions & design an enclosure
- We've tested our system in a room
  - Need to expand to a bigger area after finishing our implementation

IMPLEMENTATION: iOS APPLICATION

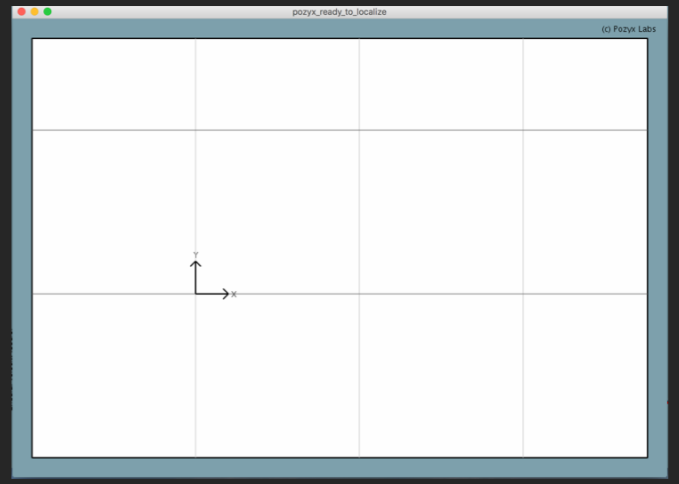
- Our Clients have specifically asked for an iOS application that Doctors, Nurses, and other hospital staff would all have access to.
  - The app is to have a map of the hospital wing that equipment we will be tracking is kept.
  - Pins marking the location of all tracked equipment relative to the map of the wing.
  - Detailed information about the specific equipment such as
    - Use State
    - Floor
    - Maintenance Record/Tracker
    - Rented/Owned
- In order to do these things we will need a MySQL DB that will receive data from the tracking IC chips as well as cloud storage for that DB so that this can be pulled from a device over a network to be used by the app.

IMPLEMENTATION: iOS APPLICATION

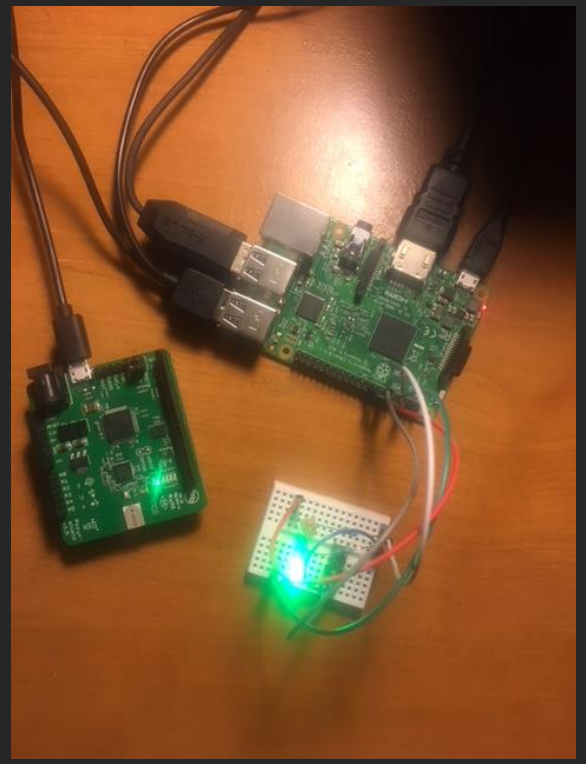
- To develop the GUI we have used an IDE called Processing to communicate with the Pozyx localization IC chip.
  - This will communicate with Pozyx system to collect positional and motion data that could be seen from the application if need be.

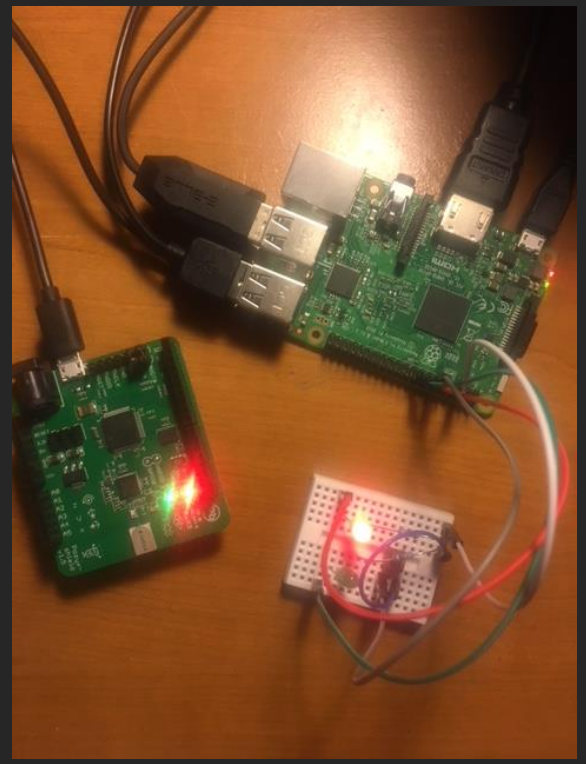


- This is the scheme for the main portion of the application.
  - Display 2D location of a Pozyx tracking chip in 2D space live
  - Uses data collected via the anchors and previous GUI to do this
  - Intend to under-lay an image of the hospital wing and create virtual boundaries.



IMPLEMENTATION: HARDWARE





IMPLEMENTATION: SCRIPT & DATABASE

Shell

```
Read   -- ID:  1234 | x:  286.0 | Y:  1884.0 | Use:  0 | Maintenance:  2018-03-06 17:56:48

Insert -- ID:  1234 | x:  278 | y:  1906 | Use:  0 | Maintenance:  2018-03-06 17:56:50
Read   -- ID:  1234 | x:  278.0 | Y:  1906.0 | Use:  0 | Maintenance:  2018-03-06 17:56:50

Insert -- ID:  1234 | x:  296 | y:  1879 | Use:  0 | Maintenance:  2018-03-06 17:56:52
Read   -- ID:  1234 | x:  296.0 | Y:  1879.0 | Use:  0 | Maintenance:  2018-03-06 17:56:52

Insert -- ID:  1234 | x:  287 | y:  1897 | Use:  0 | Maintenance:  2018-03-06 17:56:54
Read   -- ID:  1234 | x:  287.0 | Y:  1897.0 | Use:  0 | Maintenance:  2018-03-06 17:56:54

Insert -- ID:  1234 | x:  297 | y:  1867 | Use:  0 | Maintenance:  2018-03-06 17:56:56
Read   -- ID:  1234 | x:  297.0 | Y:  1867.0 | Use:  0 | Maintenance:  2018-03-06 17:56:56
Goodbye

>>>
```

Shell

```
Read   -- ID:  1234 | x:  394.0 | Y:  1821.0 | Use:  0 | Maintenance:  2018-03-06 17:53:16

Insert -- ID:  1234 | x:  579 | y:  1763 | Use:  0 | Maintenance:  2018-03-06 17:53:18
Read   -- ID:  1234 | x:  579.0 | Y:  1763.0 | Use:  0 | Maintenance:  2018-03-06 17:53:18

Insert -- ID:  1234 | x:  414 | y:  1828 | Use:  0 | Maintenance:  2018-03-06 17:53:20
Read   -- ID:  1234 | x:  414.0 | Y:  1828.0 | Use:  0 | Maintenance:  2018-03-06 17:53:20

Insert -- ID:  1234 | x:  460 | y:  1806 | Use:  1 | Maintenance:  2018-03-06 17:53:22
Read   -- ID:  1234 | x:  460.0 | Y:  1806.0 | Use:  1 | Maintenance:  2018-03-06 17:53:22

Insert -- ID:  1234 | x:  324 | y:  1334 | Use:  1 | Maintenance:  2018-03-06 17:53:24
Read   -- ID:  1234 | x:  324.0 | Y:  1334.0 | Use:  1 | Maintenance:  2018-03-06 17:53:24
Goodbye

>>>
```

	ID	X	Y	InUse	Maintenance
Edit Copy Delete	1234	297	1867	0	2018-03-06 17:56:56

	ID	X	Y	InUse	Maintenance
Edit Copy Delete	1234	324	1334	1	2018-03-06 17:53:24

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	*None*			Change Drop Primary Unique Index More
2	X	double			No	*None*			Change Drop Primary Unique Index More
3	Y	double			No	*None*			Change Drop Primary Unique Index More
4	InUse	tinyint(1)			No	*None*			Change Drop Primary Unique Index More
5	Maintenance	datetime			No	*None*			Change Drop Primary Unique Index More

CONCLUSION

- Problems: None (so far)

- Work Remaining:
  - Connect local Raspberry Pi DB to google cloud
  - Incorporate rechargeable battery life into script file and DB
  - Update tracked ventilator maintenance time
  - Create login for iOS Application
  - Pull data from google cloud into iOS Application
  - Transfer script file and software from Raspberry Pi 3 to Raspberry Pi Zero
  - Design Map of hospital wing for iOS application and update Pozyx anchor coordinates in script file
  - Solder button and LEDs to Raspberry Pi Zero
  - Design and 3D print a case for Tracker

Questions?