

# VIPS: Vector-Generating Indoor Positioning System

Jesse Campbell, Roman Parise, Jason Wang, Tanner Emerson, Chandler Ditolla

## 1 Project Description

In the pre-IoT world, two steps are required to access a device: 1) the location must be determined, and 2) the user must physically interact with the device. The menu of the IoT device's mobile application eliminates 2). However, with a sufficient number of IoT devices in a smart home, 1) becomes more complicated in that users are beginning to have to navigate through an ever-increasing number of menus in mobile applications in order to interface with more and more devices. It would be easier if simply determining the location of a device could allow the user to use the device.

We propose a system for determining the location of various IoT devices and being able to remotely access them. The project has 3 main parts:

1. To create an indoor positioning system using the Return Time of Flight localization technique in order to create a 3D map of relevant devices via an anchored, central hub. The central hub defines a common origin to which various IoT devices can be referenced and communicates to the IoT devices whether they should turn on, off, or perform more complex functionality.
2. To create a module attached to a mobile phone. This module's position and orientation are tracked by sensors in (1). The central hub will use the module's position and orientation to generate a vector extending forward in simulated space to the IoT device at which the user is pointing.
3. To create a mobile application that is used to interface with the selected device via the central hub.

By taking the users orientation into account, we can quickly determine exactly which device a local user is attempting to use and generate the appropriate interface in the mobile application.

## 2 Implementation Plan

Below is a proposed plan for the project's implementation.

## 2.1 Stage 0

Early Deadline: End of this next week

Expected Deadline: End of the quarter

Late Deadline: First weekend of summer break

- Develop an algorithm to determine the position of IoT devices given distance from various reference nodes. This should include robust error handling
- Determine the physical mechanism that ascertains the device's distance from these reference nodes (i.e. communication protocol and hardware used)
- Determine the method by which the user can access the IoT devices, specifically how the user locates the device IoT device
- Determine the specification of the central hub hardware that contains the reference nodes if one needs to exist

## 2.2 Stage 1

Early Deadline: Week 0 of Fall 2018 quarter

Expected Deadline: Before fall design review

Late Deadline: End of winter break

- Write the program that determines IoT device position using algorithm determined in Stage 0. The input to the program at this stage is simply generated test data
- Using the hardware and protocol determined in Stage 0, write program (and configure hardware) to acquire the device's distance from a reference node. Data should be verified against hand measurements.
- Design and assemble the central hub hardware if one needs to exist
- Design a test UI to interact with a basic IoT device

## 2.3 Stage 2

Early Deadline: Before fall design review

Expected Deadline: End of winter break

Late Deadline: Before winter design review

- Determine a trivially simple IoT device to use for the test
- Integrate the components developed in Stage 1 into one system

- Test using the UI to turn on/off the IoT device selected. If central hub is used, the test should pass even in the presence of obstacles between the device and the hub.

## 2.4 Stage 3

Early Deadline: End of fall quarter

Expected Deadline: End of winter break

Late Deadline: Before winter design review

- Determine the interface a hardware vendor must provide to the system to utilize the device's functionality
- Determine how a UI is to be generated from information on IoT device
- Ensure that the existing protocol supports an arbitrary number of IoT devices. Tweak algorithms if necessary

## 2.5 Stage 4

Early Deadline: End of winter break

Expected Deadline: Before winter design review

Late Deadline: N/A

- Write the UI generator. The input is simply descriptions of fictitious IoT devices.
- Develop the current system so that IoT device information and capabilities are sent to the user
- Test the current on-off system on multiple trivially simple devices

## 2.6 Stage 5

Early Deadline: Before winter design review

Expected Deadline: N/A

Late Deadline: N/A

- Develop a few simple IoT setups with limited functionality. Write the hardware specification for these. (i.e. three LEDs and a choice to turn on one of them)
- Integrate the system with the UI generator.
- Test the setup