Proposal

Overview

A device that can communicate with and control IR devices. This provides the ability to automate and remotely control numerous home devices, including TVs, set top boxes, fans, and much more. Rather than having to purchase new devices that can be controlled remotely, this device will provide IoT capabilities to any device that uses IR as a control point.

Software Components

There are three distinct components that will comprise the software for this project. Each individual unit will serve a particular function, and have varying degrees of complexity.

On-Device API

The hardware module designed and developed by the the team will have an API that enables interaction between it and the *Cloud API*. There are multiple components of this subsystem that should be illustrated. Security is critical for this system, and it will begin with a secure authentication establishment process between the device and the server-side API. Besides security, the device API will allow for specific infrared (IR) codes to be transmitted, which will be communicated by the server. The intention of this interface is to provide ample functionality, while maintaining a minimal surface of attack.

Cloud API

A server-side cloud API will serve as the main control point for the device, thus acting as a conduit for remote interaction. Amazon Web Services will be used to host the service at a low-cost, while also maintaining maximum availability at all times. Load balancing will be applied sparingly for the prototyping stage, as it will not be necessary. Ideally, AWS Lambda will be used as an on-demand provisional compute service to reduce costs further. It is likely that the server-side API will be written using JavaScript (i.e., NodeJS). In addition to compute services, it is reasonable that a database service like DynamoDB and a storage service (i.e., S3) will be used to support additional functionality.

Mobile Application

An end-user application will be developed to provide a control mechanism for the device, as mediated by the cloud API. A framework will be developed to interact with the server-side API, which will allow for the abstraction of complex logic within the application itself. Furthermore, using a framework-based approach will allow for modularity with respect to developing applications for multiple platforms. Currently, the application will be targeted for iOS and watchOS, but there may be the possibility of an Android application as well, if time permits such. The application will allow a user to control devices such as a TV, AV receiver, set top box, etc. Any devices that can be controlled with infrared will be applicable for manipulation through the application. Learning IR codes for a particular device will be built-in to the functionality of the application.

Prototype Plan

Throughout the course of creating the device, there may be multiple prototyping methods used; however, the most dominant prototyping will be evolutionary and horizontal with the development of the various components to a broad level to ensure the system integrates together into a functioning device. *Add additional details (Exp + Vert for certain comps?)*

Hardware

The hardware that will be required to be implemented include the Arduino Uno R3 board as well as various sensors and transmitters to be used in order to communicate with devices. Also, as a potential alternative to an Arduino, a Raspberry Pi might be used in the event where it provides more functionality. The sensors and transmitters required include IR (infrared) in order to communicate and manipulate devices of interest, as well as bluetooth and/or wifi transmitters in order to communicate with the user. In addition, the inclusion of RF technology would also extend the versatility of the device into being able to communicate with a wider range of devices that are becoming more prevalent.

Anticipated Challenges

Security is critical to the success of this project, and thus will be a major challenge throughout its design and development. Encryption and authentication must be built-in to each of the systems, and they must be tested against potential attacks.

The development of this project under the provided timeline will undoubtedly be the most difficult challenge, but the team already has certain components ready for integration, which will lighten the workload to some degree. The three software components are the most complex, and must be designed before any development takes place.