

Bluetooth® Based Presence Detection



Computer Science

KEYWORDS : Bluetooth®, Presence Detection, Raspberry Pi, Mobile Detection

Prithvi Sharan S

Student, Department of Computer Science, M S Ramaiah Institute of Technology, Bangalore, India

ABSTRACT

a Bluetooth enabled mobile devices in today's world are in disposable. Emerging Bluetooth technologies aim to develop communication protocols which are faster, secure and more robust in nature. Our aim is to leverage this technology to detect the presence of individuals in a finite space. This demo implementation is done using the computing power of the Raspberry Pi 2 (model B), a proprietary powerful system on chip, in order to continuously monitor the presence of registered Bluetooth enabled mobile devices in the space, store the information and report the same on a webpage hosted on the local network for users who would request access to the same. Using web technologies, the data is store and displayed in a suitable and efficient manner.

I. Introduction

About Bluetooth®

Bluetooth is an excellent option which can be used to improve the way to detect the way in which objects and even people can be detected. Any device that is Bluetooth enabled can be detected by another Bluetooth enabled device, and this property can be exploited to measure proximity of the two devices. The implementation aims to use Bluetooth enabled mobile device in order to detect the presence of individual based on the proximity from the central Bluetooth module interfaced on the Raspberry Pi system on chip.

Bluetooth, a wireless technology standard for exchanging data over short distances uses short wavelength Ultra High Frequency radio waves in the band from 2.4 to 2.485 GHz from fixed and mobile devices, in order to aid building personal area networks (PANs) was invented by a telecom vendor, namely Ericsson in 1994.

Bluetooth is managed by the Bluetooth Special Interest Group (SIG), that has more than at least 20,000 partner companies, mainly in the areas of telecommunication, computing, networking, and most importantly consumer electronics. IEEE standardized Bluetooth as IEEE 802.15.1, but no longer maintains this standard. The Bluetooth SIG monitors the development of the specification, manages the qualification program, and protects against trademark and its violations. Manufacturers must make any device meet Bluetooth SIG standards to market it as a Bluetooth enabled device. A network of patents apply to this technology, which are licensed to individual qualifying devices and their designers.[1]

About Raspberry Pi 2

Raspberry Pi is a series of card-sized single-board computers designed and developed in the United Kingdom by the Raspberry Pi Foundation with the aim of promoting teaching of basic computing science in high schools and developing countries. Originally Raspberry Pi and Raspberry Pi 2 are developed in several board configurations using licensed and authorised manufacturing agreements with Newark Element14 (Premier Farnell), RS Components and Egoman.

The primary essence and core used to implement the chip's hardware remains the same across all manufacturers. All Raspberry Pi's include the same VideoCore IV GPU, and

either a single-core ARMv6-compatible processor or a newer and a more advanced ARMv7-compatible quad-core one (in Pi 2); and 1 GB of RAM (in Pi 2), 512 MB, or 256 MB. They have a Secure Digital (SD) (models A and B) or MicroSD (models A+ and B+) socket for boot media

and persistent storage. Recently in 2014, the Raspberry Pi Foundation launched the Compute Module, for use as a part of embedded systems for the same computing power as the original Pi. Early in February 2015, the next-generation Raspberry Pi, Raspberry Pi 2, was released. The new computer board was initially available only in one configuration (model B) that had a quad-core ARM Cortex-A7 CPU and a GB of RAM with remaining specifications being similar to those of the previous generation model B+. The Raspberry Pi 2 retains the same price point of the model B, with the cheaper model A+ remaining on sale. In November 2015, the Foundation launched the Raspberry Pi Zero, a smaller product priced at an even lower price.

The Foundation provides Debian and Arch Linux ARM distributions for download. Tools are available for Python as the main programming language, with support for BBC BASIC via the RISC OS image or the Brandy Basic clone for Linux C, C++, Java, Perl, Ruby, and Squeak Smalltalk.

II. DESIGN

Experimental Setup

The overall design centres around the system on chip, the Raspberry Pi 2 B. The board is interfaced with an external USB Bluetooth module manufactured by Adafruit, a leading brand of all major system on chip compliant electronic components. Before the interfacing, Raspbian, the operating system compatible with board, was loaded and into it and dependencies such as Python were installed. Post successful installation the Python based Bluetooth module named Bluez were downloaded and flashed using commands. The Bluez module provides the basic skeletal framework to access and work with a Bluetooth dongle.

The Raspberry Pi module was then connected to a router using a standard grade RJ-45 ethernet cable. This put the Raspberry Pi module on the LAN making it accessible by other systems on the LAN.

To emulate a server running on the network a local server running on a system also connected to the same LAN. This server was used to store and also provide the information to users who request for the same after being on the same network.

A user interface was provided in the form of a php page which is compatible with mobile and well as desktop browsers.

Architecture

The external Bluetooth module, was then interfaced with the board. Python scripts used primarily to sense the pres-

ence of other Bluetooth enabled devices, was also used to obtain the unique Bluetooth address associated with every device. Using another Python script, all the detected device's addresses were forwarded to a remote server, located on the same local area network for security and to ensure privacy of the data. In the server, based on the pre defined mappings between the Bluetooth address and a name associated to the name, all unmapped entries were dropped. Along with the addresses, the timestamps of when the data was obtained were also passed in order to indicate the freshness of the data so obtained. The mapped addresses were replaced by their associated names and this along with the timestamp was posted on a webpage, which is accessible only if the device trying to access this data is part of the same network as the server on which the data is hosted. This ensures the security of the data and prevents misuse of the same. As and when data is posted on the webpage, it is also added to a database, which can be later used for analysis purposes. Figure 2.1 details this proposed architecture.

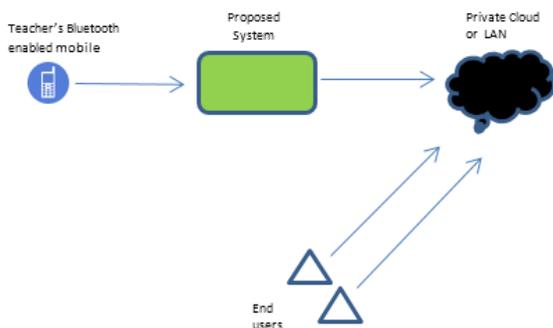


Figure 2.1: Architecture

III. IMPLEMENTATION

Sensing and collecting data

The framework of Bluez provides a manual tool to scan for the devices. However, for convenience a python code for the same which enabled storing and output information in a better format and one which is desired was written. This is shown in fig 3.1.

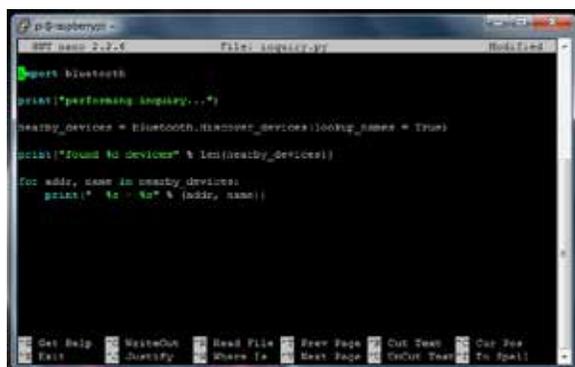


Figure 3.1: Sensing

This was the code used to perform the initial enquiry of the nearby Bluetooth devices. This is performed by a call to the Bluetooth discover function specified in the Bluez framework.

Another code snippet was used to transfer the information obtained from sensing in a suitable format to be displayed onto a web page. This was done using the PHP code in Figure 3.2.

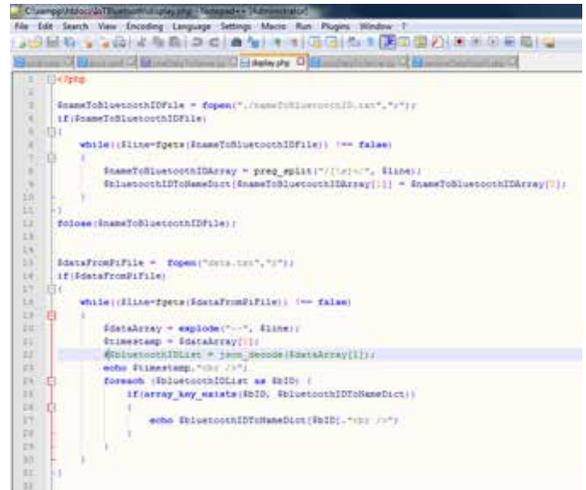


Figure 3.2: PHP code

Storage of data

The data obtained needs to be stored before further processing.

Another data set which was to be stored is the mapping of the Bluetooth addresses of the people and their respective devices. This is necessary to identify the users/persons of interest to our system. Therefore, all data received from the scanning process is to be compared with this database and only the statuses of the devices of interest are updated and displayed. This removes the possibility of other Bluetooth devices in the vicinity from being unnecessarily considered. The storage of the mapping information is shown in Figure 3.3.

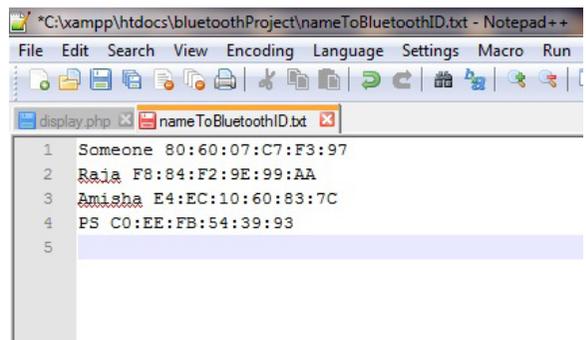


Figure 3.3

Presentation of information

The data collected is processed in the backend.

On the front end after login authentication, the user is displayed with a status of the requested people, if allowed. This is using a php/html code. This page is optimized for use on mobile browsers as well and can hence be conveniently be accessed from mobile devices.

IV. TESTING AND COMPARISON

The entire set up was tested out with a series of Bluetooth enabled devices and codes in order to compare, not only the ease of execution in terms of the load on the processor, but also accuracy of sensed data, reliability of data and ease of implementation. Several scripts were considered and implemented to check the accuracy with which they sensed the presence of active Bluetooth enabled mobile devices as this is the main point of focus in the implementation. Failure to sense the devices on time, effectively lowers the robustness of the set up whereas, failure to sense the

VII. CONCLUSION

Now that it has been established that the Internet of Things (IoT) is the most hyped emerging technology today, and that the term and the associated technologies is far from new. Given the steady growth in this field, and the simplistic yet arguably efficient design of the Bluetooth based presence detector, it will not be hard to market, and with little work on parameters user interface and some logical extensions, it could even be a great success story. These mind-blowing estimates from companies developing and selling IoT-related products and services, investing time to research and perfect even small implementations that serve some purpose would yield a proportional amount of revenue and recognition, even in the global scale.

ACKNOWLEDGMENT

I would like to take this chance to sincerely thank our educational institution, M S Ramaiah Institute of Technology, for giving us this wonderful opportunity to work on the research paper and implementation, in lieu of course credits. I would also like to thank our mentors, Prof. K G Srinivas and Asst. Prof. Sowmya B J for their constant support and guidance in every phase of the implementation, without which it wouldn't have been possible.

References

- [1] <https://en.wikipedia.org/wiki/Bluetooth>
- [2] https://en.wikipedia.org/wiki/Raspberry_Pi
- [3] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, Marimuthu Palaniswami, Internet of Things (IoT) : A Vision, Architectural Elements and Future Directions
- [4] Friedemann Mattern and Christian Floerkemeier M. Wegmuller, From Internet of Computers to Internet of Things
- [5] Wireless Communication Standards for the IoT : Greengeek White Paper
- [6] Wireless Connectivity for the Internet of Things - TI White Paper