Avion - The Intelligent Medicine Box

Mohini Reddy, Videet Parekh, Chris Pinto, Vinay Pisharody, Devesh Ratho

Abstract—Avion, the Intelligent Medicine Box is intended to make a difference in the everyday lives of people. Medicines, which serve as a medium of saving human lives, can often be fatal. People forget to take their medicines on time or at times, patients take the wrong medicines. Avion is a device created with the intention of preventing these fatalities. Avion is a medicine dispenser with a difference as it can identify and remind its users when it is time to take their medicines. It is extremely useful in cases where correct and timely consumption of prescribed medicines is of utmost importance. This device will find extensive use in cases of patients suffering from long term diseases and elderly patients or patients undergoing rehabilitation.

Index Terms—Cheap Computing, DIY Boards, Health and Fitness, Intelligent Devices, Intelligent Medicine Box, Internet of Things, Open Source Servers, Raspberry Pi Applications

1 Introduction

Internet of Things(IoT) is one of the most emerging technologies and is the subject for a plethora of Research and Development missions in science and technology. In Internet of Things, there are several everyday objects that are connected to each other via a communication medium. Experts hypothesize that the IoT will consist of nearly 50 billion devices by 2020. The main reason for the increase in the development of IoT has been the improvements in wireless technologies, micro-electromagnetics systems and the development of Internet. IoT enables objects to sense and remotely control existing network infrastructure which manifests opportunities to connect the physical world and computer-based systems. [1]

An IoT device is completely different from a computing system. The often neglected main feature of an IoT device is that the functionality of the device being connected to the internet should not change. For example, the function of a fridge is to store food at a low temperature. This functionality of the fridge should not change when it is connected to the internet. It should not be used as a device for communication between two people. Another feature of an IoT device is that it keeps the computations to the minimum. All the computations and processing should be done by a remote device and exchange of messages should enable the IoT device to take whatever actions are required.

IoT has varied applications in the fields of Transport, Embedded Systems, Health and Fitness, Home Automation, Industrial Automation, Remote Surveillance etc. This paper concentrates on an application in the Health and Fitness domain [2]. Specifically, development of an intelligent medicine box that enables patients suffering from various illnesses to take their medicines on time, every time. Often, people forget to take their medicines on time or forget them altogether. Avion will enable them to be free from this hassle by automating the process of taking medicines. The outcome of such a device coming into existence is improving the quality of human life, enabling to increase the lifespan of a person. [3]

2 Architecture

2.1 Architecture of the System as a Whole:-

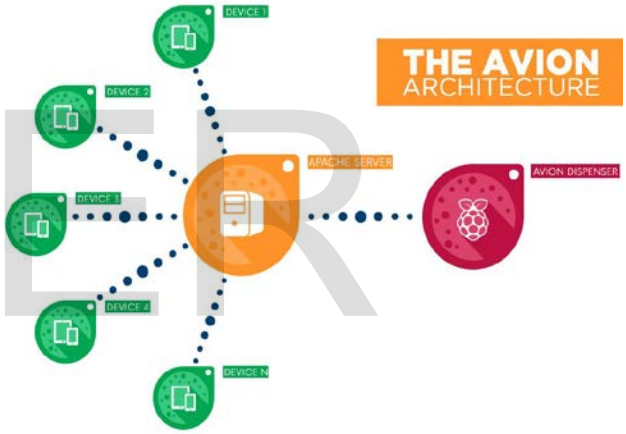

Fig. 1. The overall module view of the system.

As evident from Figure 1 which shows the overall module view of the system, we have the following components

1. User. The user makes use of the whole system. The role of the user is to receive notifications and respond to it by going near the box and collecting the medicines.

2. Avion: Avion is an intelligent medicine box that is used to dispense the pills. It is accountable for ensuring that it notifies the user when it is time to take pills. Once it notifies the user and the user is in close proximity, it will dispense the pills according to the dosage entered initially.

3. Server: The server acts as the module where information exchange takes place. Server is accountable for ensuring that the messages from the Avion are properly sent to the users' devices. It is also used to store the database as to which pill is to be dispensed to which user at what time. All the data is stored on the Server.

4. Pharmacy: The local pharmacy is notified by the Avion through the internet when the pills are going to run out of

stock so that the user does not have to keep track of the stock and the system is automated to a higher level.

5. Wi-Fi Interface: The Users' Devices, the server and the Avion are connected to the internet via the Wi-Fi Router or any other interface that already exists in the household.

Figure 2 shows the Communication Architecture that helps to understand the architecture better with the detailed communication flow.

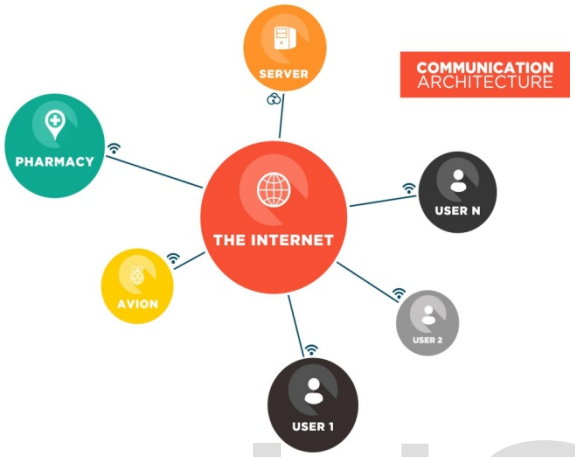

Fig. 2. The Communication Architecture of the System

2.2 Architecture of Avion

Figure 3 shows the physical architecture of Avion Box. Avion will be a circular box consisting of three levels. These levels will consist of two levels of pill storage (the top and the middle) and the bottom level will contain the Command Generating unit of the device. The top layer of the pill storage and dispensing will be used for large pills. The middle layer of the pill storage and dispensing will be used for smaller pills. Each of the two layers will contain up to six tubes from where the medicines will be dispensed. The segregation of the large and the small pill dispensing layers help in providing more space overall for the electronic functioning modules that are responsible for physically dispensing the pill.

The pill storing and dispensing compartments in each layer will be tilted. This will enable the use of gravity in the function; thus reducing the external work required to dispense the pill.

The bottom layer will contain the internal controlling module of the device as well as provide as a medium of the device communicating with the system with respect to notifying the user when it is time to take his/her pills, notifying the pharmacy when a particular compartment of the device is running out of pills and needs to be refilled.

There will be an LCD screen on top of the box that will show the information about the person who is taking the medicine that is being dispensed. This will assist in further reducing the human error that can occur.

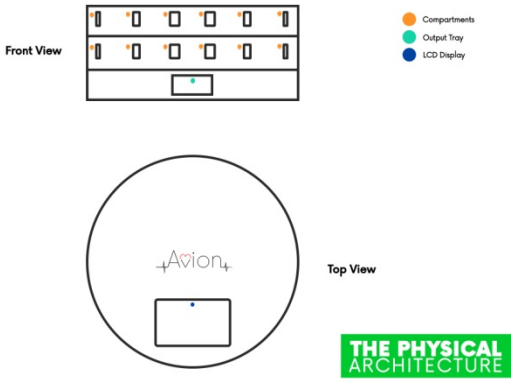

Fig.3. The Physical Architecture of the Box

3 Choosing the Communication Module

There are various boards that enable a user in the prototyping phase of a product. The cheapest and most user friendly hardware are called the Do-it-Yourself(DIY) Boards. Some examples of such DIY boards are: Arduino Development Boards, Raspberry Pi Development Boards, BeagleBone Black Development Board, Intel Galileo Development Board, pcDuino Development Board, Uruk Development Board, Goldilocks Development Board, Extracore Development Board, SparkCore Development Board, DigiSpark Development Board etc. [4]

Choosing the correct Development Board for the project depends on the requirements of the System. The vastly used, cheap yet powerful and user-friendly development boards of the ones mentioned, are the Arduino and Raspberry Pi. [5]

3.1 Arduino

Arduino DIY Board is an open-source microcontroller system that has enabled people from all walks of life to be innovative and experiment with electronics. It was first developed by Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis. It is a very extensible DIY board that is very user friendly and easy to use and program on using the open source program called Arduino. [6]

Figure 4 shows a picture of the Arduino Chipset.

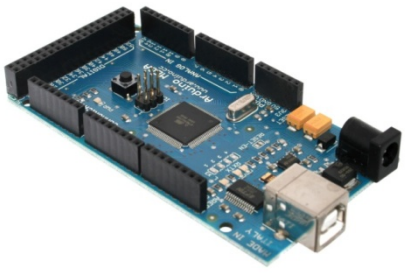

Fig.4. The Arduino Chipset

3.2 Raspberry Pi

Raspberry Pi DIY Board is from the family of Credit-Card sized computers. It is a cheap yet effective computer that provides a processing framework, capable of a great many uses. It was invented by Pete Lomas, Alan Mycroft, Jack Lang, Rob Mullins, Eben Upton and David Braben. It can run many Linux operating systems that are light weight on the processor. Some of the operating systems that are generally used for Raspberry Pi are: Raspbian, ARCH Linux ARM, OpenELEC, Pidora etc. [7],[8] Figure 5 gives a picture of the Raspberry Pi Board.

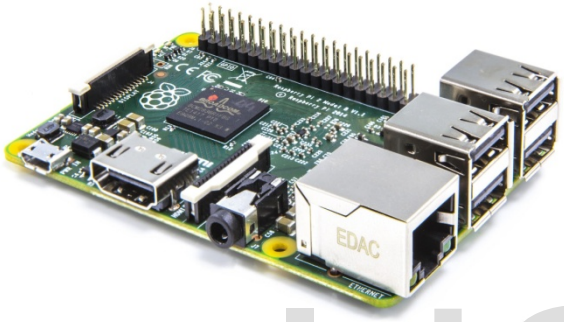

Fig.5. The Raspberry Pi Board

3.3 Comparison of Arduino and Raspberry Pi

TABLE 1
COMPARISON CHART BETWEEN ARDUINO AND RASPBERRY PI

Metric	Arduino	Raspberry Pi
Hardware	Microcontroller Board	General Purpose CPU
Processing Capabilities	Low	High
Wi-Fi module	External	Inbuilt
Programming Language	Arduino	Python
Operating System	No	Yes (Linux)
Internet	External	Inbuilt
Cost (Approx.)	$25	$35

From the comparison shown in Table 1, it is evident that the Raspberry Pi is more extensive in its functionalities and hence, is a better option for a DIY Board for development of projects.

4 CHOOSING THE SERVER MODULE

There are many open source servers that have been made for enabling developers and programmers to make their own servers for free. Some of these are Zenoss, Mono, Apache, SugarCRM, Drupal, OpenSolaris, Nginx etc. Of these, the most extensively used are Apache and Nginx.

4.1 Apache

Apache HTTP Server is the most extensively used Server in the world. It was developed by Robert McCool. It is an open Source Server that can be implemented and configured in any Linux System easily. [9]

Fig. 6. Apache Logo

4.2 Nginx

Nginx Server is one of the most used servers. It was invented by Igor Sysoev. Nginx is generally used for serving static web content. [10]

Fig. 7. Nginx Logo

4.3 Comparison between Apache and Nginx

TABLE 2
COMPARISON CHART BETWEEN APACHE AND NGINX

Metric	Apache	Nginx
Supported Operating Systems	Ubuntu, Fedora, Mac OSX, OpenSUSE	Ubuntu, Debian, Red Hat, FreeBSD
Content	Dynamic	Static
No of concurrent requests	Nearly Equal	Nearly Equal
Configuration	More Flexible	Less Flexible
User Support	Higher	Lower
Memory Usage	High	Low

From the comparison shown in Table 2 [11],[12], it is evident that for the chosen application, Apache is more reliable as it can handle dynamic content well. Apache will be the better choice as there are a lot of concurrent requests and multiple users dynamically adding their data that should be reflected on the database efficiently.

5 CONCLUSION

The paper proposes an intelligent medical automation system using a Raspberry Pi and an Apache Server implementation, targeted towards the improvement of medical health-care facilities, with special emphasis on hospitals, old age homes, and rehabilitation centers. The functionalities of the system are elaborated upon, and a systems and communication architecture is proposed.

ACKNOWLEDGEMENT

We would want to take the opportunity to acknowledge Prof. Supriya Agrawal who has been a support so as to give a structure to the course of the research.

REFERENCES

[1] A.B. Nkoro and Y.A. Vershinin, "Current and Future Trends in Intelligent Transport Systems on Cars and Infrastructure" in IEEE 17th International Conference on Intelligent Transport Systems (ITSC), OCT 2014

[2] M. Surya Deekshith Gupta, Vamsikrishna Patchava, Virginia Menezes, "Healthcare based on IoT using Raspberry Pi"

[3] Geng Yang, Li Xie, Matti Mäntysalo, Xiaolin Zhou, Zhibo Pang, Li Da Xu, Sharon Kao-Walter, Qiang Chen, and Li-Rong Zheng, "A Health-IoT Platform Based on the Integration of Intelligent Packaging, Unobtrusive Bio-Sensor, and Intelligent Medicine Box," IEEE Transactions on Industrial Informatics, Vol, 10, No 3, November 2014

[4] Mepits, "Best Development Boards for DIY Projects" https://www.mepits.com/project/236/DIY-Projects/12-Best-Development-Boards-for-DIY-Projects 2016

[5] Peter Jamieson, Jeff Herdtner, "More Missing the Boat - Arduino, Raspberry Pi, and Small Prototyping Boards and Engineering Education Needs Them"

[6] Arduino, https://www.arduino.cc/en/Main/ArduinoBoardUno 2016

[7] Raspberry Pi, https://www.raspberrypi.org/help/what-is-a-raspberry-pi/ 2016

[8] NetworkWorld, "Ten Operating Systems for Raspberry Pi"http://www.networkworld.com/article/2842678/computers/ten-operating-systems-for-the-raspberry-pi.html 2016

[9] Apache, www.apache.org 2016

[10] Nginx, www.nginx.com 2016

[11] Digital Ocean "Apache vs Nginx Practical Considerations"https://www.digitalocean.com/community/tutorials/apache-vs-nginx-practical-considerations 2016

[12] Dreamhost, "Web Server Performance Comparison" http://wiki.dreamhost.com/Web_Server_Performance_Comparison 2016