Home automation using Raspberry Pi

Matija Novak, Dragutin Kermek
Faculty of Organization and Informatics
University of Zagreb
Pavlinska 2, 42000 Varaždin, Croatia
{matnovak, dkermek}@foi.hr

Abstract. *Home automation is becoming a very popular activity with the development of credit card-sized and cheap single-board microcontrollers like Arduino and especially single-board computers like Raspberry Pi. Every person which has small knowledge in programming and electronics can start some project to automatize his/her home. With dedicated device (Arduino or Raspberry Pi) and a few sensors for temperature, light (optosensor), humidity, infra-red light, etc. one can automate his home environment.*

There are various electronic devices in one's home like radios, television, etc. that are not in use due to obsolescence or malfunction but the repair of such devices is not profitable. Some parts or even modules of such devices can still be of good use when doing home automations and to enforce so called the green approach.

The development of home automation project should be based on small modules (hardware and software) that are testable and then gradually integrated into final project. Few decisions must be made about features one would like to have and usually few compromises are set on the path to the solution.

This article shows a proof-of-concept that combines home automation using the Raspberry Pi, so called green approach and modular development.

Keywords. Home automation, Raspberry Pi, Internet of Things, green approach, open source, Java, sensors.

1 Introduction

It the area of modern computing very popular term during last years is Internet of Things (IoT) which "represents the concept of computers and machines with sensors, which connect to the Internet to report status and accept control commands"[1]. IoT was first used in 1999 by Kevin Ashton. Another wider definition describes IoT as "the pervasive presence of a variety of devices—such as sensors, actuators, and mobile phones—which, through unique addressing schemes, are able to interact and cooperate with each other to reach common goals." [2]

Currently, a lot of research is done in the topic IoT e.g. [3], [4], [5], [6]. There were few big global founded projects like [7] dealing with different areas and applications of the IoT with participants from academia and industry that were coming from many European Union (EU) countries. Communities like Open Web Application Security Project (OWASP) are working on specific projects like security issues associated with the IoT[8]. Few well-known examples for IoT applications today are [9]:

- Wearable devices
- Home automation
- Industrial asset monitoring
- Smart energy meters.

In the focus of this paper is home automation as one of aspects of IoT. Early definitions of home automation cover "dealing with the control of home appliances from a central location." [10] Later generation is wireless network based so it defines home automation as "wireless embedded sensors and actuators that enable monitoring and control applications for home user comfort and efficient home management." [11] Recent definition see home automation as "systems represent the front-end of smart grids, where the energy monitoring and control operations are enabled through smart devices installed in households and residential buildings." [12]

There are various technologies that can be used for home automations. In the article "A comparison of the popular home automation technologies" by Withanage et al. they compare different technologies [12]. But all those technologies are not likely to be used by common person.

In this article we are focusing on how to do home automation that could be done by a common person. By common person we mean someone who has a basic knowledge in programming and electronics, but he/she should have average level of using of computers. This could be a student at high school or university, an engineer, or any kind of profession with do it yourself enthusiasm etc. Using Raspberry Pi is one way to do first steps in home automation that can be done by common person.

According to recent publication "The 2015 DZone Guide to the Internet of Things" [13] 66% of the participants are interested in using IoT for hobby development. Developers are asked about what type of IoT products have worked on. Home automations in on the top with 23% followed by transportation (10%) and environmental (9%). Similar question was about what IoT products are interested in working on. Home automations in again on the top with 68% followed by drones (63%) and environmental (59%).

Using Raspberry Pi for home automation is not new idea because it was already presented in several articles like “Raspberry Pi based interactive home automation system through E-mail” by Jain [14].

This article shows a proof-of-concept that combines home automation using the Raspberry Pi, so called green approach and modular development. By green approach it is meant dismantling existing broken (damaged, non-usable) electronics, like radios, television, computers, etc. (usually called e-waste) and use their functional parts to reuse them to build new things.

In Section 2 home automation approaches are described and explained why Raspberry Pi was selected. In Section 3 the green approach is explained. Section 4 gives an example how one can combine Raspberry Pi, few sensors and other elements, and parts from old broken television (TV) to make a small home automated environment using “green approach”. A project has gradually expanded by adding new hardware and software modules in order to simplify its development and to fulfill some of requirements of IoT. We presented some development issues we met and tradeoffs we set in order to achieve affordable, modular and easy to build solution. The Section 5 gives the conclusion.

2 Raspberry Pi and home automation

First approach of doing home automation is to leave it to the professionals. In this approach of home automation the user hire a professional firm that will prepare a project plan according to his/her desire and finally deliver and install proper solution. This solution is usually a high level solution of home automation and it is expensive. On the other side its benefit is that it takes less time than the second approach.

Second approach is the so called “do it yourself”. It is usually cheaper but requires more time due to various reasons. Everything from design to realization needs to be done by yourself. For common person that approach usually demands a lot time in searching for ideas and solutions, ordering parts and modules, learning to acquire knowledge and skills in several electronics and programming areas etc. Good starting point are popular books for Raspberry Pi beginners like [15], [1],[16].

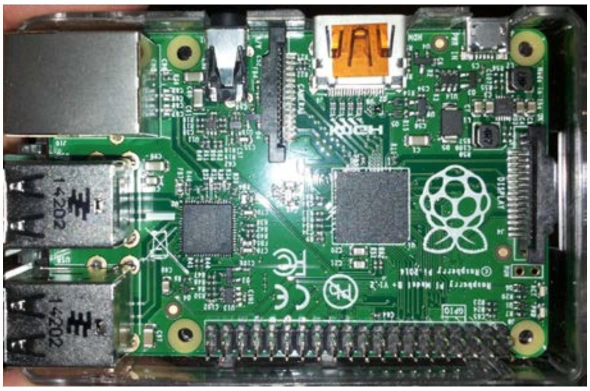

Figure 1 Raspberry Pi

There are various way of doing the second approach. Devices, small computers, like Arduino, Raspberry Pi, Banana Pi, etc. are available for affordable price. In this article Raspberry Pi is used because of its characteristics, good community and support for various programming languages.

Raspberry Pi (Fig. 1) is a simple one-board computer (device) on sale since February 29, 2012 [17] for 30$. So it is a good candidate for cheap option to build home automation around it. Raspberry Pi is small device so it does not take much space and it does not use up much electric energy. Raspberry Pi is open to be used with many operating systems (OS), but mostly it is used with some Linux OS distribution (e.g. Raspbian). Linux versions prepared for Raspberry Pi can be downloaded from the Raspberry Pi home page [18].

Raspberry Pi is constantly upgraded and currently there are five versions of Raspberry Pi: Model A, Model A+, Model B, Model B+, and Model B – generation 2. There is also a compute module that can be used for more professional work. Raspberry Pi’s recent sales data are as follows[19]:

- 3.8 million October 23, 2014
- 5 million February 17, 2015
- 6 million June, 8, 2015.

Raspberry Pi software is mostly open source and various programing languages can be used like C, C++, Python, and Java. Github search for term Raspberry gave around 11.000 projects. Almost half of them have been written in Python. This is not oddly because from the beginning Raspberry Pi has been promoted as “little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python.” [20] Support for other programming languages, and for instance Java as the most popular language of all, has come later in 2013. [21]

Some of the programing languages have special editions of development kit (like Java ME Embedded or Java SE Embedded) that are prepared to be used on embedded devices such as Raspberry Pi.

Besides CPU, main memory, storage (SD card) and network connectivity (LAN or wireless), Raspberry Pi has the general purpose input output (GPIO) interface over which one can easy work with sensors, engage relays to control different devices like (stepper or servo motors, fans, doorbells, light, etc.). Model B+, that was used in example has 40 GPIO pins which can be used to get input/output signals.

To do own small home automation project, except Raspberry Pi, electronic parts like sensors, resistors, LEDs are needed. At the beginning that could be at least one of the following sensors: humidity, temperature, moisture, etc. They can be bought online for 1$ or 2$. Downside of these cheap sensors could be their durability and sometimes precision, too. For example, longer usage of sensor for measuring ground humidity will result in probe melt down and then it is no longer useful.

To lower the price for building home automation it was decided to use so called “green approach” which is described in the next section.

3 Green approach

The “green approach” means that several existing electronics are reused which usually would end on the junk yard. Most people have in their home some radios, televisions, or other electronic devices that brake down over time and which repair is too expensive.

Electronics TakeBack Coalition publish report [22] on E-Waste for year 2010 for USA and it shows disposal of 51,900,000 computers and 28,500,000 televisions with recycling rates of 40% and 17% respectively. So significant percentage of e-waste might be candidate for pre-processing to extract useful parts and then put in normal recycling.

The fact is that when the whole device is not usable the most of its electronic parts like sensors, conductors, resistors, etc. could still working properly and can be used for home automation building. For example usually when television (TV) breaks down then the infra-red receiver and the remote controller are still intact and perfect for home automation.

Fig. 2 shows high level “green approach” and how it is combined with Raspberry Pi and other bought sensors. Using the “green approach” for home automation projects has two main benefits. First it is the reduction of cost for the user that has intended to do home automation by itself. Second if the working parts are used for home automation projects the amount of e-waste can be reduced.

When using the “green approach” a modular development should be used. Modular development helps to develop parts of the system, test them individually and finally integrate them in specific solution. When using the parts from old devices their manual is not always present. So it is necessary to figure out how it works to successfully build a module around individual part. In the next section an example project is presented where the “green approach” was used and the section describes how the whole system was built using the modular development.

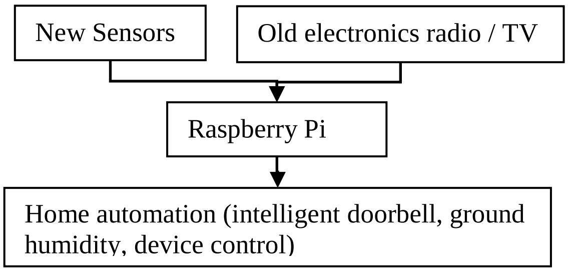
New Sensors
Old electronics radio / TV
Raspberry Pi
Home automation (intelligent doorbell, ground humidity, device control)

Figure 2 High level “green approach”

4 Example of usage

The proof-of-concept was based on few small modules that could function as separate projects but here they were assembled as final project. Starting point was easy home automation that reuses old but correct electronics parts hooked up to Raspberry Pi. The authors have previously described some development details in following technical publications [23], [24].

4.1 Using the “green approach”

We had an old broken Sony TV whose repair was more expensive than buying a new one. Decision was to take out its parts that are working and reuse them in our project. One such part was the remote IR sensor which is used to control the TV with a remote controller. We took the whole front board (Sony – H1 - board) and connected it to the prototyping breadcrumb board which was connected to Raspberry Pi GPIO.

To get the remote controller to work first it is needed to install drivers and libraries on Raspberry Pi that can enable getting the pressed button. We used Linux infrared remote control (LIRC) library [25] which is a generic library that can be configured to work with any kind of remote controller. All we need to do is to install LIRC and start the setup process so the frequency of the remote controller can be set up. Once this is done we need to configure the buttons and that basically means we define what output is generated when button 1 is pressed.

Next we needed a programing language to read an output from the sensor that was generated upon one press a specific button. We decided to use Java SE embedded edition. We could use Python, Java ME or any other programming language, too.

Java was chosen because LIRC has a jLIRC library that can be easy used with Java. Other arguments for choosing Java were compiled executive file (.jar), security model, execution speed, etc. There is also a LIRC library to use with Python if one prefers it. Although we prefer and use Java for final version of the project, we use Python to test specific small modules because many examples are written in Python and we could reuse them.

Java SE embedded application was built using jLIRC library whose source code must be downloaded and compiled before usage. At that point we have the application that consists of basic module that reads the output from IR sensor via LIRC, recognizes which button is pressed but it does not have concreate usage per se so it must be integrated in some context. In this case we agree it should start the low cost ($30) Raspberry Pi camera. The remote controller should control the camera to turn it on and off, and to shut down the whole Raspberry Pi. So we add new module for camera management

4.2 Extending the system

Inside startup procedure exists dedicated part that deals with our devices. First step is starting the camera (we used simple Linux Raspbian command raspivid over the Java Runtime class). Once the camera is operational next step is activating a live web streaming from it (we used Linux cvlc[1] command) or time-lapse picture capturing (we used motion-mmal[2] application). That means one can see the real-time video or picture with a browser over his/her computer. This simple module can then be used as surveillance camera that can be turned on and off with the remote control.

For humans very important is visual control of state of devices. LED is added to indicate when a button is pressed on the remote controller. To enable the feature with the indicating LED we just made a simple code that set the signal for GPIO pin to turn on and off the LED. The simplest way to implement this functionality was using device input output (dio) library which has the classes and interfaces to work with GPIO. Although it has very simple functionality we could see it as new module for LED management.

The project's hardware and software parts are organized in layers as shown in Fig. 3. The control data flow is based on event model so upon one press a button on the remote controller the system must do proper action. This task is entrusted to a ButtonListener who at first must discover which button is pressed and then trigger its dedicated action. The class diagram in Fig. 4 shows the usage of jLIRC and dio library. A prototype of the project is shown in Fig. 5.

The project has a practical functionality but one must admit that there are few limitations in its design and implementation. First one is in the area of Raspberry Pi camera because it gives solid quality picture as long as is bright enough. In low light or during the night Raspberry Pi camera produces not usable picture. To upgrade the system to a night (full day) surveillance camera several new parts should be added.

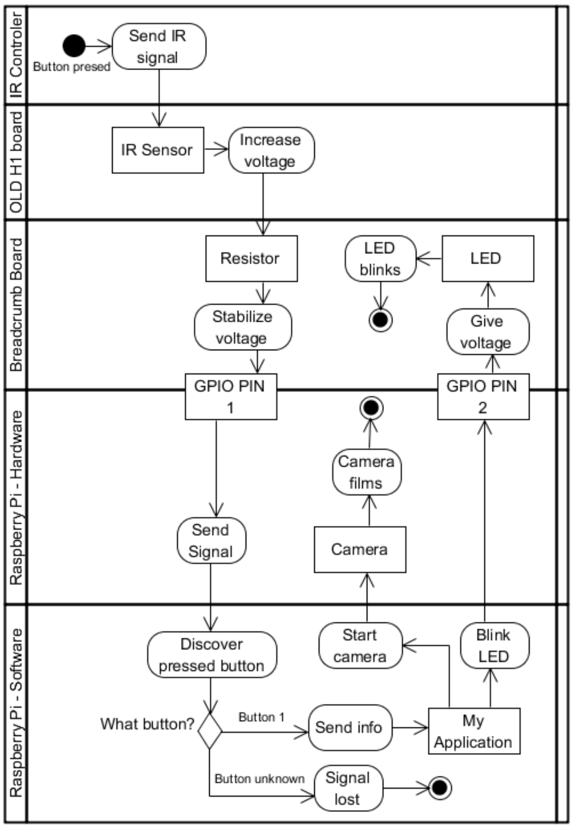

Figure 3 Activity diagram of connected parts

The infra-red LED boards are usually sold with Raspberry Pi camera as a kit or one can buy them separately. The Raspberry Pi camera kit gives simple solution because infra-red LED boards could be fixed to the camera's module by screws. At the same time their wholes serve as pins for a current and a ground so they do not need extra wires to hook them up to the system.

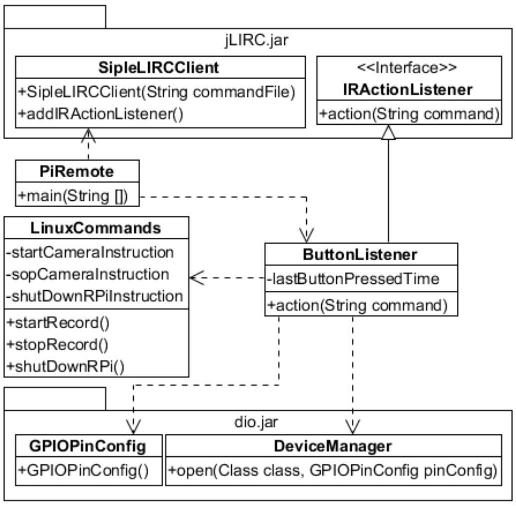

Figure 4 Class diagram of remote control

Should we think we have final solution? The test work has proved a few new issues. Infra-red LEDs emit high temperature after relative short period of work and their boards are very hot so one could get

[1] https://www.raspberrypi.org/forums/viewtopic.php?t=43969

[2] https://www.raspberrypi.org/forums/viewtopic.php?t=44966

burned on them. The problem is that there is no way to turn off infra-red LEDs via normal service provided by the kit so they are operating as long as the computer is on.

4.3 Using modular development

The solution could go in few directions and first one is to add a relay to which infra-red LED boards will be connected instead of the camera's module. Here we must use wires of proper length and could adjust each infra-red LED position as we like because they are not screwed to the camera module. The relay is connected to GPIO pins so we can programmatically turn it on or off which means to turn infra-red LEDs on or off. It is possible to use different power supply for infra-red LED boards. In this case time schedule is the main player in the system that will decide when right time to infra-red LEDs turn on or off is.

Probably the better solution is one which has the same hardware solution as previous one but is based on optosensor and event model around it. When sensor signals that there is lack of light that triggers LightListener which turns the relay to on state and infra-red LEDs begin to emit infra-red beam. So the picture will have acceptable quality and recognizable greenish/grayish tone that provide infra-red LED. On the opposite side when sensors reads enough light it signals different value to LightListener and it turns the relay off and consequently the infra-red LEDs stop they work. And they can cool off.

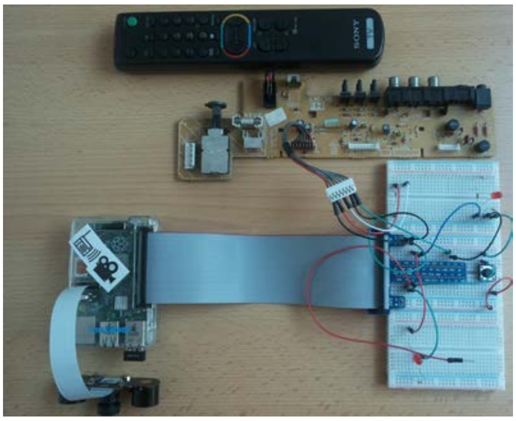

Figure 5 Example of a project

Sometimes one would like to see when infra-red LEDs are turn on. This is a perfect case to reuse previous module for LED management and adjust it to new functionality. Mainly this means usage of new GPIO port and different logic to control it.

According to the underling idea of "Internet od Things" people, devices, processes, and data should be networked. Although our project at current state has one way connectivity it is not sufficient for serious application. Second limitation lays in a way we present information to the users. Live web streaming or picture capturing suffers from processor hunger and in case of more users over wireless connection congestion channel could happen.

The simplest solution goes in direction of using front server to which our system should upload video clips or pictures. So the users will see and use that server instead of our back system based on Raspberry Pi. The only question is how our system will upload data to the server? There are many ways to do that task and among them we prefer Web upload, Web Distributed Authoring and Versioning (WebDAV[3]) or FTP because they have open source server software and it is easy to find Java libraries for client side purposes.

Another way is to implement own simple protocol that will consist of set of commands that will cover all needed activities. If we choose that way then we must build both sides wherein a client is our system and a server is our front server. The client will have new module for uploading data. For the server we must develop new application that will use one port and handle uploaded data in a way that web server software (e.g. Apache Http, Apache Tomcat) can fetch them upon user's demand.

Third limitation lays is a way we control the camera and it is only by physical presence via remote controller. We would like to control the whole system remotely and by this we mean Internet based solution. First solution is consisted of two parts, client and server, as in previous module. We prefer to build new module that will act as a server on one port, wait for user's request and allowed command transfer to sensor module. For client side we can choose one or more types of applications. They are graded from text mode application, over desktop application to mobile application. For the functionality of the system it does not matter what type of application client uses.

If someone prefers web applications then the second solution could be based on them. In that case application server must be installed (e.g. Apache Tomcat) and we must develop our Java web application that will accept request that client sends to it. This application must communicate to our system and the question is how because it is running in its own virtual machine? One answer is very close to the previous solution but with internal server so there is no need to explain it again. The system diagram in Fig. 6 shows the operational view of the modules.

The software part of the project has not been explained in detail. For the protocol and command execution we used some of GOF [26] patterns like Command and Interpreter. One could use more simple solution based on Regular Expressions. For Listeners we use Observer pattern. For integration of modules we use Builder, and for instancing Web parts Factories. LED modules are based on State pattern.

We could further extend the functionality of the system with various features that use other sensors (e.g. temperature measurement to control infra-red

[3] https://tools.ietf.org/html/rfc4918

LED boards heating), buzzer to send audio signal in case of overheating, LCD to present information (e.g. temperature, number of uploaded and downloaded pictures, number of users, status of camera etc.), touch screen to use as input device and to present information etc.

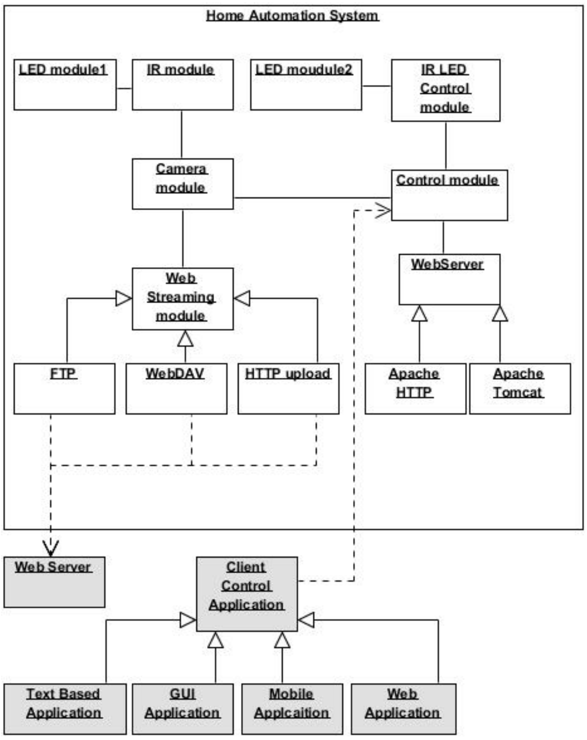

Figure 6 Operational view of the modules

Security was not in the focus of this project but it is very important according to DZone's research [13]. The developers think that security is the most important concern (79%) followed by privacy (69%) and connectivity/device management (44%). Those three concerns are on the top two years in a row with small increase in security.

5 Conclusion and future work

To enter "do it yourself" home automation, one just needs one small device like Raspberry Pi, few sensors, relays, modules, and a bit of knowledge in programming and electronics. For a beginner in this subject probably the most important factors for success are curiosity and courage to deal with new area.

On the other hand the most important factors for failure are setting to complicated project, breaking security measures, not reading manuals or specifications of electronic parts etc. This could result in destruction of some electronic parts like sensor, LED, screen, or even the Raspberry Pi. But the worst situation could happen if someone get hurt.

In this article it is shown that affordable home automation is possible on small scale and with application of "green approach" it can result in additional savings.

Home automation project must be planned in such manner that developer could build and test each module (with hardware and software) as solo project. After one module satisfies planed functionality it is ready for next step in development process. Each module is integrated in final project one by one performing unit tests and after all modules are integrated follows integration tests.

Home automation as sort of IoT requires connectivity and we mainly mean that user can get information about the functioning of the system (or its selected parts), send control commands etc.

In our current focus of work it is not necessary that a human being should be the only user that can communicate with the system. We establish simple communication protocol and for instance another Raspberry Pi system can communicate with the home automation system in order to collect data, supervise its work, and adjust parameters of selected modules depending on some algorithms by sending control commands.

6 Acknowledgments

This work has been supported in part by the Croatian Science Foundation under the project IP-2014-09-3877.

References

[1] D. Norris, *The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black*. Tab Electronics, 2015.

[2] D. Giusto, A. Iera, G. Morabito, and L. Atzori, *The Internet of Things*. New York, NY: Springer New York, 2010.

[3] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Futur. Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[4] M. Darianian and M. P. Michael, "Smart Home Mobile RFID-Based Internet-of-Things Systems and Services," in *Advanced Computer Theory and Engineering, 2008. ICACTE '08. International Conference on*, 2008, pp. 116–120.

[5] M. Kovatsch, M. Weiss, and D. Guinard, "Embedding internet technology for home automation," in *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*, 2010, pp. 1–8.

[6] M. Selinger, A. Sepulveda, and J. Buchan, “Education and the Internet of Everything,” 2013. [Online]. Available: http://www.cisco.com/web/strategy/docs/education/education_internet.pdf. [Accessed: 01-Jan-2015].

[7] “Internet of Things Architecture,” 2013. [Online]. Available: http://cordis.europa.eu/project/rcn/95713_en.html. [Accessed: 17-Apr-2015].

[8] Open Web Application Security Project, “OWASP Internet of Things Project,” 2015. [Online]. Available: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Top_Ten_Project. [Accessed: 17-Aug-2015].

[9] K. Lueth, “Why the Internet of Things is called Internet of Things: Definition, history, disambiguation,” 2014. [Online]. Available: http://iot-analytics.com/internet-of-things-definition/.

[10] A. Z. Alkar and U. Buhur, “An internet based wireless home automation system for multifunctional devices,” *IEEE Trans. Consum. Electron.*, vol. 51, no. 4, pp. 1169–1174, Nov. 2005.

[11] C. Gomez and J. Paradells, “Wireless home automation networks: A survey of architectures and technologies,” *IEEE Commun. Mag.*, vol. 48, no. 6, pp. 92–101, Jun. 2010.

[12] C. Withanage, R. Ashok, C. Yuen, and K. Otto, “A comparison of the popular home automation technologies,” in *2014 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*, 2014, pp. 600–605.

[13] DZone, “THE DZONE GUIDE TO THE INTERNET OF THINGS 2015 Edition,” 2015. [Online]. Available: https://dzone.com/guides/internet-of-things-1. [Accessed: 17-Aug-2015].

[14] S. Jain, A. Vaibhav, and L. Goyal, “Raspberry Pi based interactive home automation system through E-mail,” in *Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on*, 2014, pp. 277–280.

[15] D. Norris, *Raspberry Pi Projects for the Evil Genius*. Tab Electronics, 2013.

[16] M. McManus and S. Cook, *Raspberry Pi For Dummies*. For Dummies, 2013.

[17] BBC, “The Raspberry Pi computer goes on general sale,” 2012. [Online]. Available: http://www.bbc.com/news/technology-17190918. [Accessed: 01-Jan-2015].

[18] Raspberry Pi Foundation, “Raspberry Pi,” 2014. [Online]. Available: https://www.raspberrypi.org/. [Accessed: 17-Apr-2015].

[19] Adafruit, “About 6 million Raspberry Pis have been sold,” 2015. [Online]. Available: https://blog.adafruit.com/2015/06/08/about-6-million-raspberry-pis-have-been-sold-raspberry_pi-raspberrypi-mattrichardson-twit-newscreensavers/. [Accessed: 01-Jan-2015].

[20] Raspberry Pi Foundation, “What is a Raspberry Pi?,” 2014. [Online]. Available: https://www.raspberrypi.org/help/what-is-a-raspberry-pi. [Accessed: 01-Jan-2015].

[21] Raspberry Pi Foundation, “Oracle Java on Raspberry Pi,” 2014. [Online]. Available: https://www.raspberrypi.org/blog/oracle-java-on-raspberry-pi/. [Accessed: 01-Jan-2015].

[22] Coalition Electronics TakeBack, “Facts and Figures on E-Waste and Recycling,” 2014. [Online]. Available: http://www.electronicstakeback.com/wp-content/uploads/Facts_and_Figures_on_EWaste_and_Recycling.pdf. [Accessed: 01-Jan-2015].

[23] M. Novak and D. Kermek, “Home automation using Raspberry Pi,” in *CASE 27 - Razvoj poslovnih i informatičkih sustava*, 2015, pp. 41–47.

[24] M. Novak and D. Kermek, “Internet of things with RPi and Java,” in *JavaCro’15 - 4. international Java conference in Croatia,* 2015.

[25] A. Leamas, Christoph Bartelmus, Heinrich Langos, Karsten Scheibler, Jim Paris, and Milan Pikula, “LIRC - Linux Infrared Remote Control,” 2015. [Online]. Available: http://www.lirc.org/. [Accessed: 11-Mar-2015].

[26] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Pearson Education, Limited, 2005.