NodeMCU12e + Nextion Tft an Experimental Survey with Virtual Keyboard in IoT Projects

Antonio Carlos Bento[1], Anderson Rocha de Barros[2], Érica Oliveira Lopes da Silva[3], Leonardo Augusto Taniguti Mantovani[4]

Computer Science, Universidade Nove de Julho, São Paulo, Brazil
[1]acb01@hotmail.com; [2]andy.rb@hotmail.comr; [3]ericaolsilva@gmail.com; [4]leonardo.mantovani@hotmail.com.br

Abstract—This manuscript presents the results of an experimental research on the construction of virtual keyboards for the Internet of Things (IoT), using the NodeMCU12e controller in conjunction with the Nextion touchscreen display. The creation of projects for the Internet of Things is in constant development, applying technical and scientific models that meet the different types of solutions, in this way, the results of this experience are presented during the construction of a solution that serves parts of academic projects, contributing in this way as a research base for more complex projects, which require a solution with virtual keyboard for data entry, or even for the configuration of different types of devices, such as Arduino or Raspberry Pi controller. With the results obtained with this study, it was possible to make feasible the construction of a proposal that meets the needs of a small project, allowing to expand its possibilities and resources according to the need of the project to be developed.
Keywords—IoT; NodeMCU12e; Nextion; Virtual Keyboard; Internet.

I. INTRODUCTION

The main objective of this work is to present the results of an experimental research, using the Internet of Things or IoT, in which different devices connected to objects are used, so that it is possible to control at a distance, as well as the monitoring of environments, normally used in residential and business automation, nowadays it is very common to build and develop various solutions to meet the most complex types of projects [2].

As a specific objective, a solution was developed which demonstrates the main devices and tools necessary for the construction of a virtual keyboard, usually used for data entry, such as for accessing Internet access point names, passwords, locations and user information, these data are normally used to control and send information in different media, allowing, for example, access to a WiFi network, or even setting a given limit for device control, such as maximum or minimum temperature.

The motivation for the development of this project was based on an academic need that involved the creation of solutions for monitoring and controlling patient activities, such as people with disabilities, special needs, the elderly or even children, for the construction of solutions which could serve the project, it was necessary to develop projects that could control the data, as well as user input to access a residential WiFi network, or somewhere remote, allowing to enter the network name, user name and password.

It is observed that there are contents on the development of solutions in a technical way, but for scientific studies there is not a relevant quantity which can serve as bibliographical reference on the subject addressed, thus intending to contribute to the scientific knowledge, providing quality and theoretical and practical reference for more complex projects,

II. BIBLIOGRAPHY REVIEW

As a bibliographical basis technical and scientific materials were used, which made possible the creation of the project, much of the content on the subject has originated in technical materials developed and made available by the device manufacturers, these materials are usually developed by technical teams or specialists, the which make their work available on blogs, websites and discussion forums.

The bibliographies with academic contents were basically searched in the bases of IEEE Xplore, IEEE Latin America and Scopus, which did not provide a significant content for the study, few of the contents found, served as a basis for the theories and specification of the devices used, also demonstrating some concepts, such as Internet of Things, use of controllers, and applications in different projects.

The technical bases that were used include the controllers for the Internet of Things, types of touchscreen display, and have been the results of other comparative studies between different brands and types of devices. The algorithms used were adapted from the examples, models and templates provided by the manufacturers themselves, thus following a working model according to the best practices of the manufacturers.

With references on the topics covered on IoT, bibliographies were used: Frustaci et al. [19]; Muñoz et al [22]; Voas et al. [15]; Oteafy [24]; Want et al. [23]; Taivalsaari

& Mikkonen [5]; Bedi et al. [10]; Chernyshev et al. [18]; CERP [7]; ITU-T [14]; Arduino [6].

With references on the NodeMCU12e controller and the Nextion Tft, the following were used: Naranjo et al. [8]; Murdan & Emambocus [4]; Al-Kuwari et al. [17]; Kodali & Gorantla [21]; Ngamsuriyaroj et al. [25]; Rahman et al. [16]; Villamil et al. [27]; Arya et al. [20]; Zhang et al. [11]; Durrani et al. [26]; Mouna et al. [3]; Bento [1]; Bento [2]; Itead [12]; IteadNextion [13].

III. METHODS AND MATERIALS

It was used the experimental research method, which consists in the accomplishment of experiments using tools and techniques, thus allowing to perform an analysis on the results obtained during the development of the project, initially were conducted studies on manuals and technical documentation, enabling the understanding of possible solutions for the project, after the bibliographic studies, were carried out tests with prototypes on the solution, with the results it was possible to discuss and evaluate the proposal, allowing to make decisions about the best approach model on the problem.

The prototypes served as a basis for a practical validation of the studies, making it possible to simulate situations, as well as sending and receiving data, in the case, for the configuration of the controlling devices, for the purpose of transmitting or receiving data, over a WiFi network, or even to store information about the date and time when events may occur, during a monitoring process, or any other alert about problems that may occur during a follow-up, thus enabling more effective control over the event.

The NodeMCU12e devices were used as materials, being this one Arduino type controller, which is commonly used in design for projects that involve Internet of Things, being a device with low cost, also because it is easy to locate in the national and international market , this device was selected, by these main characteristics besides others as they are discussed by Bento [1].

Another device that has been selected is the touchscreen Nextion Tft display, this device allows to communicate with the NodeMCU12e controller, with advanced features such as touch, in which it is possible to use buttons and a simple interface, in which is easy to use for an user without much knowledge about the Nextion display was selected after the studies carried out by Bento [2].

A notebook with the Windows 10 64-bit operating system was also used for the construction of the algorithms, as well as for the development of the screen for use with the Nextion display, also serving for the transmission and reception of the data between the devices, also making possible the validation during data transfer.

As a development tool, the Arduino IDE was used [2], being a platform commonly used for the creation and configuration of the controllers used for projects on the Internet of Things, was used the C++ programming language

for the construction of the algorithm that will allow communication between devices. Another tool used was the ITEAD Nextion display editor, which is necessary for the construction of Nextion display screens, this tool is available by the manufacturer itself, which facilitates the construction of the screens in a simpler and more intuitive way.

Other materials used were: mini USB cable for communication with the NodeMCU12e controller and Notebook, wires for connection between the Nextion and NodeMCU12e devices, enabling a communication between them, a 16GB microSD memory card was also used to load the screen project between the Notebook and the Nextion display, enabling the update of features.

IV. RESULTS AND ANALYSIS

As a first step, studies were carried out on the materials and techniques used for the construction of solutions using the controllers normally used for projects involving the Internet of Things, based on other projects and experiences of professionals and technicians in the area of solutions for residential and commercial, equipment was then selected according to its capacity, cost and ease, and these premises are important and necessary for the development of the project.

During the studies, the different types of controllers, such as the Arduino Uno [1], were considered. This presented little capacity for storage, processing and memory, besides needing other devices to complete the project, such as an external WiFi device, another important factor that was considered the size of the device, compared to the NodeMCU12e controller, which has greater capacity, processing and memory as presented in the studies developed by Bento [2], besides the ability to have an integrated WiFi device, also having a good size less compared to the Arduino controller.

For the display device, different models such as ILI9341 and Nextion tft were evaluated, and the ILI9341 model, despite being one of the most used and available in the market, presented a lower cost, was observed in the results of other studies presented by Bento [2] , that its structure and direct programming in lines of code, makes difficult its use, since it is necessary the development of many lines to solve a simple problem, as for example, to present a value on screen, besides not having the resource of touchscreen, these problems increase the use of programming code lines and processing, in addition to needing more devices, as buttons to control user actions, as well as using many outputs or ports of controller connections.

After the comparative studies developed, the Nextion display was selected because of its high capacity, design, image quality, touchscreen features, and the ease of programming and creation of screens for control, with its own development tool called ITEAD [12], which enables the creation of professional screens with quality and ease, using diverse libraries, with drag and drop capabilities.

The studies provided the choice of devices that could contribute to the development of the project, such as the NodeMCU12e controller and the Nextion Tft touchscreen display, for its ease of use, capacity and speed, and the tools provided by the manufacturers, which effectively collaborated to the development of the solution, presented below.
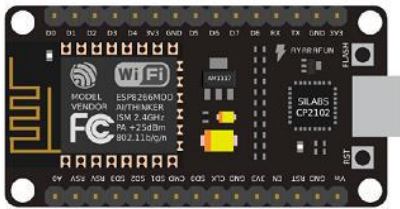
A. *The devices*

Fig. 1.NodeMCU 12e device, used in the project.

For the use of the NodeMCU12e controller, it was necessary to configure with the libraries and technical information available on the manufacturer's website, as well as the follow-up and studies developed by technical professionals, such as those presented in [2].

After the proper settings in the Arduino IDE, it was possible to perform the first tests of the component use, making some data transfers and using some examples available in the billiotheques of the device.
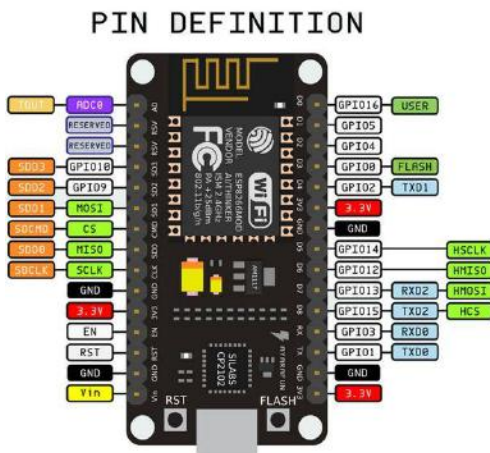

Fig. 2.The NodeMCU12e pin definition, used for connection with another devices.

The Nextion display has some parts, such as cables and a micro USB connector, making it possible to connect to other devices, as well as having a large number of libraries that can handle different types of projects. For the design the Nextion display mini-USB connector should not be used, this is normally used for transferring data directly between the NExtion display and a computer, or even to perform the transfer of power.

Fig. 3.The display Nextion Tft NX3224T024 2.4".

The Nextion connection settings allow the device to be used with different controllers, such as Arduino Uno, Raspberry Pi, NodeMCU12e, as the device uses only two cables, one for TX data transmission, one for RX data reception, this provides better facility for the availability of ports to other devices in the controller.
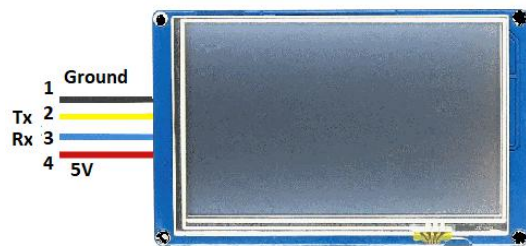

Fig. 4.The connection pins used with the display Nextion.

As shown in Figures 2 and 4, there are a large number of connection pins available on the NodeMCU12e controller, for the ease of connection of the Nextion device, which uses only 2 pins, this provides greater flexibility for the use of other devices that can be connected, such as a temperature sensor, humidity sensor, or even for heart rate monitoring.

For the presentation of this project only a virtual keyboard must be used, so that it is possible to configure a connection to a WiFi network, where the user must enter the name of the access point and the password, as the NodeMCU controller already has a device, it allows the connection in another local network, such as in a residence, then the algorithm developed and made available in the controller will allow communication with the local WiFi network, informing if the access was successful or not.

The feature of a virtual keyboard is very useful, in situations where it is necessary for the user to perform some type of configuration manually, to define, for example, some alert threshold, date and time information, as well as the configuration of a network access, such as WiFi.
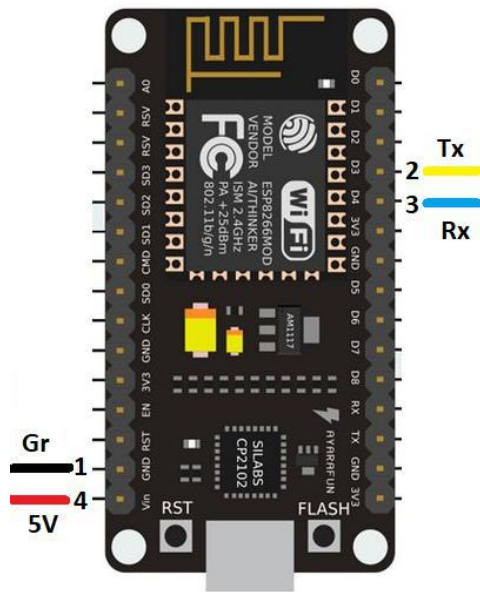
Fig. 5. The connection pins used with the display Nextion and NodeMCu12e.

As a basic configuration, the documents and manuals presented by the manufacturers of the devices, as well as other sources of information such as technical documents and blogs of different authors, were used to create solutions for the Internet of Things using the NodeMCU12e controller as they present [2]. After the necessary configurations and tests, it was possible to begin the construction of the algorithms of the project, to effectively develop the screens for the Nextion display.
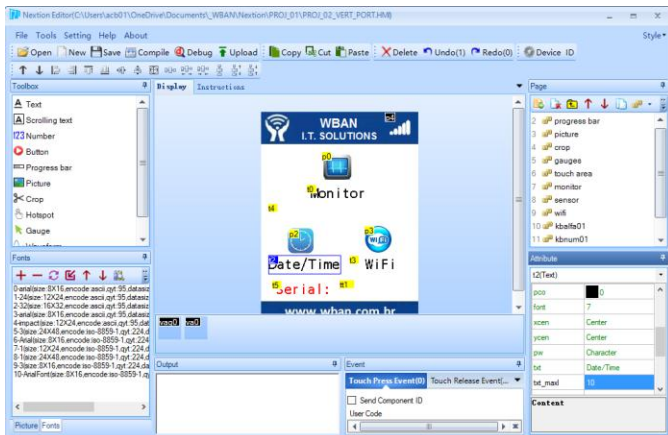
B. The Nextion screen editor Itead

Fig. 6. The project main screen created with the Nextion editor [12].

The main screen was developed to meet the project, initially presenting the options for the monitoring screen, another icon for setting Date and Time, another option for the configuration of the WiFi access point, the latter having the features necessary for discussion in this study. The other options should not be discussed during this project, as they contain a large amount of information, forming part of other sub-projects, which complete the main project.

For the WiFi access point configuration screen, one object button was used to save the data, two text boxes to enter the access point name and the password, by pressing the name field of the access point system directs the user to the virtual keyboard's main screen.
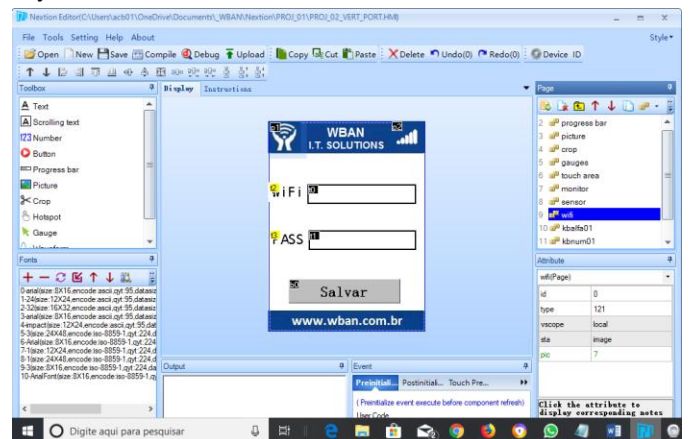

Fig. 7. The setup WiFi screen created with the Nextion editor [13].
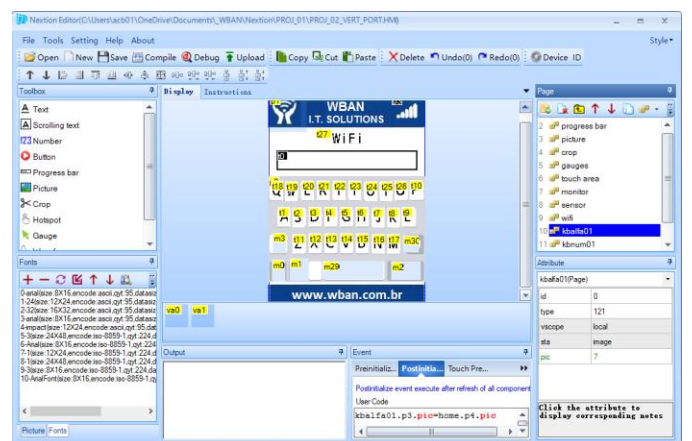

Fig. 8. The setup WiFi screen created with the Nextion editor [13].

For the virtual keyboard was used a figure that represents a keyboard, on the figure and the keys, were created buttons to identify the object that was pressed by the user, this was the system created with the Nextion Editor, records the characters selected by the user in a text box, just above the screen, this to present the user with their options during the configuration.

Three more keys have been created to meet the needs of the user, one to change the characters from uppercase to uppercase, another to be used as input only for numbers, and another for input of special characters, all screens use a text type , which should keep the data selected by the user, then this data is transferred to the main screen of the access point configuration.

Was created also buttons to delete the characters, and a space button, in addition to the Save button, used to store the data entered in a variable and pass them to the main screen of the access point of the WiFi access point, as in a keyboard these functions are defined in each action of the buttons.
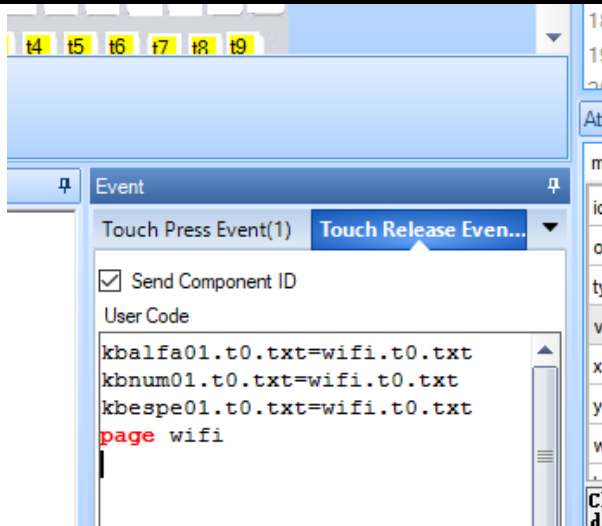
Fig. 9.The Save button commands created with Nextion editor [12].

For the Save button, a basic code has been created to pass the selected data to the main screen, in this case the values that are presented in the field of type text box are passed to the main screen, then the system redirects the user to the access point input screen WiFi.

C. *The algoritm created with the Arduino IDE*

For the construction of the algorithm, the Arduino IDE tool was used, which allows the configuration of the devices, as well as the construction of algorithms that can take care of the project, for more details on the configurations an extra analysis is necessary, available in the works presented by Bento [2] and Arduino [1].

The #include "Wire.h" library was used, for the connection of the devices, the Nextion binaries are available on the GitHub website, the Neo Nextion binaries were used.

```
#include <Nextion.h>
#include <NextionPage.h>
#include <NextionText.h>
#include <NextionButton.h>
#include <NextionVariableString.h>
#include <NextionVariableNumeric.h>
#include <NextionSlidingText.h>
#include <SoftwareSerial.h>
#include <NextionPicture.h>
SoftwareSerial nextionSerial(D3, D4); // BLUE_D3=RX, YELLOW_D4=TX
Nextion nex(nextionSerial);
```

The following algorithm was developed by means of a function to perform the procedure of recording the data sent by the Nextion display after the user presses the Save button.

```
void t9_callback0(NextionEventType type,
INextionTouchable *widget) {
Serial.println(F("Callback..."));
```

```
t9_vb0.setText("Saved-Restart");
char buffer[19];
if (t9_vt0.getText(buffer, 19)) { Serial.println(buffer); }
qsid=buffer;
if (t9_vt1.getText(buffer, 19)) { Serial.println(buffer); }
qpass=buffer;}
```

After creating the algorithms and transferring the code to the devices, it was possible to perform different tests, correcting problems such as incorrectly configured buttons, as well as some loss of communication during the use of the display, thus allowing validation of the solution development, enabling new adaptations.

The data were cleared in the EEPROM of the controller, after the information about the WiFi access point is configured, the device requests that it be restarted, during the initialization the device tries to make the connection with the configured WiFi access point, error or success during operation.

Fig. 10. The project in operation with the NodeMCU12e controller and with the display Nextion touchscreen [2].

As the final result shown in figure 10, there is the input screen of the WiFi access point of the user's residence, in this way it is possible to send information and data collected during the use of the device to be sent to a remote base by the user. internet, making it possible to create reports and remote monitoring.

Some difficulties faced, involve the understanding of the tools as well as the lack of skill gift devices, which after a brief period of study was possible to improve the solution by using more advanced features compared to other existing projects.

V. CONCLUSION

The results allowed to conclude on the possibilities of creating solutions using virtual keyboards with Arduino type controllers for the Internet of Things, making possible the validation of the functionalities, as well as the use and application of the resources developed with the Nextion device and the NodeMCU12e controller, making possible the

construction of a viable solution that can meet the different types of projects.

Due to the great amount of technical content, it is intended that this work may contribute to the development of more complex solutions, collaborating with the scientific society, involving students, researchers and professors, who are interested in constructing simple solutions, which can be complemented depending on the type of project.

With the rapid evolution and growth of solutions for projects that involve the Internet of Things, it is understood that there is still a great need to develop solutions for the academic area, as well as practice in this type of solution, although it is a study that involves interdisciplinarity, since several studies are used, such as the development of algorithms, electrical, data transmission, these factors end up inhibiting new projects, because it is necessary a period of study and practice to understand all the infrastructure used.

ACKNOWLEDGMENT

REFERENCES

[1] A. C. Bento. IoT: NodeMCU 12e X Arduino Uno, Results of an experimental and comparative survey. International Journal of Advance Research in Computer Science and Management Studies, v. 6, p. 46-56, 2018.

[2] A. C. Bento. IoT: Results of an Experimental Survey with Nodemcu, Nextion, DS1307 and NRF24L01+. International Journal of Current Innovation Research, v. 4, p. 969-974-974, 2018.

[3] A. Mouna ; B. Abdelilah ; N. K. M'Sirdi ; E. H. Abarkan. Estimation of the state of charge of the battery using EKF and sliding mode observer in Matlab-Arduino/LabView. 2018 4th International Conference on Optimization and applications (ICOA). IEEE Conferences. pp. 1-6, 2018. Arduino

[4] A. P. Murdan ; M. Z. A. Emambocus. Indoor positioning system simulation for a robot using radio frequency identification. 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA). pp. 986-991, 2018. NodeMCU

[5] A. Taivalsaari ; T. Mikkonen. A Taxonomy of IoT Client Architectures. IEEE Software. IEEE Journals & Magazines. v. 35, Issue: 3, pp. 83-88, 2018.

[6] Arduino. 2018. Official Available at: https://www.arduino.cc

[7] CERP. (2009). IoT – Internet of Thinsgs of European Research Cluster. Internet of things: Strategic Reserach Roadmap, Available at: http://www.internet-of-thingsresearch.eu/pdf/IoT_Cluster_Strategic_Research_Agenda_2009.pdf

[8] D. Naranjo ; P. Córdova ; C. Gordon. Wearable Electrocardiograph. 2018 International Conference on eDemocracy & eGovernment (ICEDEG). IEEE Conferences. pp. 201-205. 2018. NodeMCU

[9] Development of bio-shields for Arduino Uno. 2018 Advances in Science and Engineering Technology International Conferences (ASET). IEEE Conferences. pp 1-5, 2018.

[10] G. Bedi ; G. K. Venayagamoorthy ; R. Singh ; R. R. Brooks ; K. Wang. Review of Internet of Things (IoT) in Electric Power and Energy Systems. IEEE Internet of Things Journal. IEEE Journals & Magazines. v. 5, Issue: 2, pp. 847-870, 2018.

[11] Haoyu Zhang ; Guomin Li ; Yaru Li. A Home Environment Monitoring Design on Arduino. 2018 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS). IEEE Conferences. pp. 53-56. 2018.

[12] Itead. Studio. Nextion Development Software and Informations. https://nextion.itead.cc/

[13] IteadLibNextion. Libraries and examples for Nextion devices. https://github.com/itead/ITEADLIB_Arduino_Nextion

[14] ITU-T. (2012). Internet of Things Global Standards Initiative. Available at: http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=y.2060.

[15] J. Voas ; B. Agresti ; P. A. Laplante. A Closer Look at IoT 's Things. IT Professional. IEEE Journals & Magazines. v. 20, Issue: 3, pp. 11-14. 2018.

[16] K. K. M. Rahman ; M. M. Subashini ; M. Nasor ; A. Tawfik.

[17] M. Al-Kuwari ; A. Ramadan ; Y. Ismael ; L. Al-Sughair ; A. Gastli ; M. Benammar. Smart-home automation using IoT-based sensing and monitoring platform. 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018). IEEE Conferences. pp. 1-6. 2018. Nodemcu

[18] M. Chernyshev ; Z. Baig ; O. Bello ; S. Zeadally. Internet of Things (IoT): Research, Simulators, and Testbeds. IEEE Internet of Things Journal. IEEE Journals & Magazines. v. 5, Issue: 3, pp. 1637-1647, 2018.

[19] M. Frustaci ; P. Pace ; G. Aloi ; G. Fortino. Evaluating Critical Security Issues of the IoT World: Present and Future Challenges. IEEE Internet of Things Journal. IEEE Journals & Magazines. v. 5, Issue 4, pp. 2483-2495, 2018.

[20] M. S. Arya ; K. Anjali ; D. Unni. Detection of unhealthy plant leaves using image processing and genetic algorithm with Arduino. 2018 International Conference on Power, Signals, Control and Computation (EPSCICON). IEEE Conferences. pp. 1-5, 2018.

[21] R. K. Kodali ; V. S. K. Gorantla. RESTful Motion Detection and Notification using IoT. 2018 International Conference on Computer Communication and

Informatics (ICCCI). IEEE Conferences. pp. 1-5, 2018. Nodemcu

[22] R. Muñoz ; R. Vilalta ; N. Yoshikane ; R. Casellas ; R. Martínez ; T. Tsuritani ; I. Morita. Integration of IoT, Transport SDN, and Edge/Cloud Computing for Dynamic Distribution of IoT Analytics and Efficient Use of Network Resources. Journal of Lightwave Technology. v. 36, Issue:7, pp. 1420-1428, 2018.

[23] R. Want ; W. Wang ; S. Chesnutt. Accurate Indoor Location for the IoT. Computer. IEEE Journals & Magazines. v. 51, Issue: 8, pp. 66-70, 2018.

[24] S. M. A. Oteafy ; H. S. Hassanein. IoT in the Fog: A Roadmap for Data-Centric IoT Development. IEEE Communications Magazine. IEEE Journals & Magazines. v. 56, Issue: 3, pp. 157-163, 2018.

[25] S. Ngamsuriyaroj ; W. Chira-Adisai ; S. Somnuk ; C. Leksunthorn ; K. Saiphim. Walking Gait Measurement and Analysis via Knee Angle Movement and Foot Plantar Pressures. 2018 15th International Joint Conference on Computer Science and Software Engineering (JCSSE). IEEE Conferences. pp. 1-6, 2018.

[26] Sajjad Durrani ; Hussnain Arif ; Ehtisham Ali ; Asad Ali ; Shahid Khan ; Hazrat Ali. A smart framework for power distribution and load balancing using Arduino. 2018 1st International Conference on Power, Energy and Smart Grid (ICPESG). IEEE Conferences. pp. 1-6, 2018.

[27] X. Villamil ; T. Guarda ; G. N. Quina. Agile software development for mobile applications and wireless interaction with hardware development board (Arduino). 2018 13th Iberian Conference on Information Systems and Technologies (CISTI). IEEE Conferences. pp. 1-5, 2018.