ISSN 1870-4069

Method for Introducing IoT Project Development Using Free Software Tools

León Torres-Restrepo[1], Alicia Martínez-Rebollar[1], Miguel González-Mendoza[2], Hugo Estrada-Esquivel[1], Fabio Vargas-Agudelo[3]

[1] National Technology of Mexico/CENIDET, Mexico

[2] Monterrey Institute of Technology and Higher Education, Mexico

[3] Antioquia Institute of Technology, Colombia

{leontorres16c, alicia.mr, hugo.ee}@cenidet.tecnm.mx, mgonza@itesm.mx, fvargas@tdea.edu.mx

Abstract. The Internet of Things (IoT) encourages the use of tools that facilitate the development of projects focused on smart cities. These types of projects face significant challenges due to the number of technologies used and the interaction and interrelation between the different systems. There are various methodologies for the development of this type of project, but some are at a very high level to be used by small work teams or people with medium knowledge. This research paper proposes a method for introducing IoT project development focused on small teams or people with medium knowledge. The proposed method works as a guide that covers from the understanding of the business to the implementation of the solution. The method was evaluated through the construction of a prototype that covers the stages of data acquisition, processing and visualization.

Keywords: Application development method, IoT project, internet of things, free software tools.

1 Introduction

The Internet of Things (IoT) has caused transformations that influence aspects such as the economy, the environment, politics and people's daily lives [1]. This encourages the development of intelligent applications capable of interacting with the environment and making decisions based on the information acquired from the different objects [2–4]. However, the development of this type of application involves a high degree of complexity, because it requires increasingly sophisticated techniques of artificial intelligence or automatic learning and generally involves more than one area of knowledge. In this sense, it is important to generate proposals that seek to mitigate the gap that exists between the conceptualization and implementation of the IoT concept.

In this research work, a method is proposed that can be used as a guide for the introduction to the development of internet of things or smart cities applications. The method is detailed through a case of real study in the domain of intelligent transport systems. In the case, a prototype is built that covers the data acquisition, processing and visualization stages. These stages are common in solutions that involve signals from the environment, which are used in the development of data science applications, industry 4.0, smart cities, among others. The use of the method facilitated the construction of the prototype in time and development complexity, thanks to the use of existing free software tools.

The article is organized as follows: Section 2 shows the state of the art; Section 3 describes the proposed method; In section 4 an example of the method is presented to a case study; section 5 shows tests and results. Finally, section 6 presents the conclusions and future work.

2 State of the Art

The development of applications for the internet of things presents a high level of complexity, due to the dependencies and interrelations of the different systems and the technologies involved [5, 6]. The development of this type of applications presents some challenges related to the acquisition, integration, quality or availability of the data applications [7, 8]. Thus, the design of applications for IoT requires understanding and deepening in conceptual, technical and methodological aspects that may differ slightly from traditional development [9, 10]. From the methodological point of view there are some interesting aspects. Por example, there are disciplines focused on the exploitation of data [11, 12], and paradigms focused on exposition of applications and services [13, 14]. However, when reviewing these approaches, several similarities can be observed in terms of their life cycle. i.e., starting with the acquisition of the data and ends with the use of data for a specific purpose [15, 16].

Therefore, the development of applications under the new paradigms is very complex. Some of the traditional tools and methodologies have been adjusted to meet new demands [17, 18]. These adaptations or modifications to the methodologies have reduced the complexity and adaptation of software developers in the construction of the development of new intelligent applications.

3 Method for Introducing IoT Project Development

The proposed method is based on various computational tools and on the benefits offered by internet of things paradigms and smart cities. Figure 1 shows the steps of the proposed method, which are: analysis of the context, requirements analysis, technology analysis, systems design, solution construction and systems implementation. The first stages of our method focus on the conceptualization of the method, while the last two stages focus on the construction of the solution.

The main contribution of our method proposed is focused on the technology analysis stage.

This new proposed stage is where the developer must carry out the analysis of the technology to be used in the development of their application, this directly influences the most appropriate techniques and technologies to use, which allows the reduction of complexity and the time of development of the application. Currently, there are several tools that facilitate the development of Internet of Things applications for developers with different knowledge of these technologies.

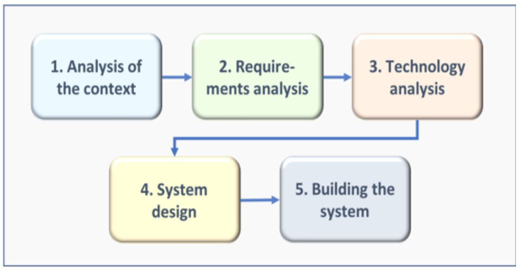

Fig. 1. Overview of the proposed method.

Our method starts with context analysis, which focuses on understanding and modeling the scenario for which you want to develop the IoT application. Once the context is understood, the business requirements and rules that must be considered to obtain the expected functionality are defined and analyzed. The next step is proposed based on experience and its main objective is to allow the identification of data sources and the selection of event detection techniques. In the next step, the design of the prototype architecture is carried out and the final structure of the system to be obtained is modeled. Finally, we proceed with the construction of the system. The next section presents an example of the method's application.

4 Case Study

The validation of our method was carried out in a real case study. This paper presents a smart transport case study which describes a smart application that allows to monitor to handling of a vehicle. When a vehicle is driven in the wrong way, an alert is generated to the users of the application. They may be other drivers or people who are near where the alert was generated. The application detects driving events automatically, which can be: driving at an unauthorized speed, speed changes, driving in the opposite direction, sudden stops of a vehicle, strong turns and road irregularities. However, for reasons of space, only the first two processes will be detailed.

Stage 1. Analysis of the context: in this stage the entities are identified, and the interactions that exist between them. For example, if the following scenario occurs: a car is traveling in an area at an illegal speed. The entities identified in this scenario would be: car, zone and alert. The monitoring of this car in a certain area will allow to identify the attributes of speed limits that the area has, so an alert will be generated when the limits are exceeded. The alert must contain information about the car and the place where this event was generated. In addition, to sending the alert to all users who are close to this event.

Stage 2. Requirements analysis: business requirements and rules are defined at this stage. At this stage, all requirements must be identified considering the frequency of the data, its restrictions, etc. continuing with the scenario of the previous stage: the speed of a car must be analyzed according to the area in which it is located. All areas or roads have a minimum and maximum speed allowed. Any speed value that is outside these limits will be taken as an illegal speed. Monitoring the speed of a car requires the analysis of static and dynamic data, which come from different sources. Dynamic data corresponds to the travel speed and location of the vehicle. The static data correspond to the delimitation of the areas and the speed limits established for each area. Dynamic data can be obtained from different sources (mobile devices, embedded devices, or external devices such as a camera). Static data is taken directly from the traffic regulations.

Stage 3. Analysis of the technology: in this stage, the identification of data sources and the selection of detection techniques are carried out. At this point, it is important to identify the techniques and devices that will be used to monitor the events of the application to be developed. Obtaining information in a smart application can be done in different ways. For example, speed can be determined using a GPS sensor or through a video camera using image processing techniques. Detection of the speed not allowed in a car was carried out through basic mathematical operations. However, building applications for IoT and smart cities generally involves more complex operations that require the use of techniques such as machine learning to obtain more successful results. In our case study application, pattern recognition techniques were used to detect some events identified in the application, such as handling in the opposite direction.

In addition, at this stage the computational tools that will be used for the development of the application must be identified. Free software tools such as: QuantumLeap, and sklearn were used for this case study. QuantumLeap is a generic component of the FIWARE platform, which groups the following modules: Orion Context Broker, MongoDB, CrateDB and Grafana. These components allow managing context information, managing information storage and working with time series to visualize the data in real time respectively. On the other hand, Sklearn is a library developed in Python that contains algorithms focused on pattern recognition. For data transfer, the data models proposed by the FIWARE platform [19] and schema.org [20] were used.

Stage 4. System design: At this stage, the architecture of the prototype to be developed must be defined. The developed smart application allows to monitor the driving of a user through her smartphone. Some aspects that must be considered are those related to the processing and storage capacity of the data acquisition device, for which it must be determined if an external storage of the data is required. As well as the time in the data transfer. However, when using artificial intelligence algorithms and techniques with a high degree of complexity that require large computing capacity, the mobile device may not have enough processing capacity to give immediate responses. In this context, we assume that not all processing can be done in the module that is on the mobile device, or in the cloud. In addition, the design of the modules was carried out with a low level of coupling between them.

This allow us that, in the future the modules can be used independently. Once the abstraction of the stage has been done, we proceed with the design of the architecture.

The proposed architecture for this smart transport application is organized in two subsystems (1) event detection (smartphone) and, (2) monitoring and visualization (cloud). Figure 2 shows the architecture proposed in this case study.

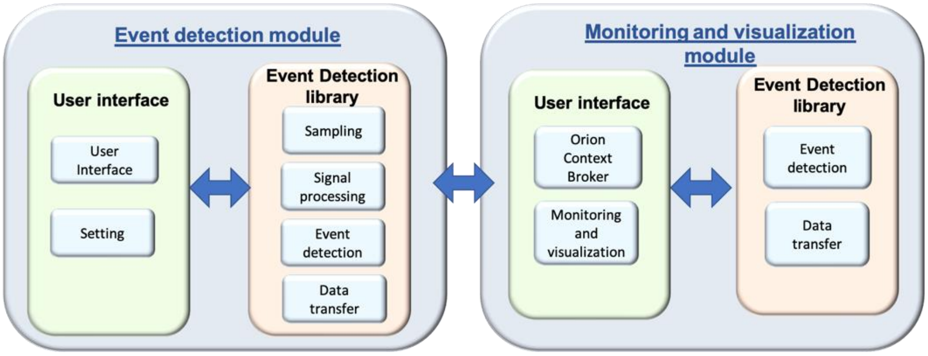

Fig. 2. General Architecture of the smart transport application.

The event detection module gets the data from the smartphone. Therefore, it must be implemented in the smartphone. This module must acquire the data from the sensors, and analyze them to determine if any event has occurred to send information to the monitoring and visualization module. Furthermore, this module must also allow user interaction. With these specifications, an architecture is designed for the event detection module composed of a user interface and a library for event detection. The user interface contains a graphical interface for user interaction and a block that allows the configuration of some parameters such as the connection to the monitoring module. The library is made up of four blocks which cover the process from data acquisition, passes through event detection and ends with data transfer (reprocessed data or alert notifications). The monitoring and visualization module receives the data that comes from the event detection module, stores it, and allows its visualization; and when necessary, processing for event detection is carried out. The design of this module is similar to the event detection module, which is made up of a user interface and an event detection library.

Stage 5. Construction of the system: in this stage, the implementation of the intelligent transport application was carried out. Some of the reference values identified in the application are: the minimum and maximum allowed speed that each of the zones or roads has. This information is taken in real time from the GPS of the smartphone. The first task of acquiring real-time data from the application consists of obtaining every second the GPS position (latitude and longitude), the speed at which the car travels and the time at which the data is obtained. Each data obtained is filtered using the moving averages technique, with the aim of reducing noise. Later all this information is packaged and sent to the distribution block. The second task consist in to obtain and to prepare the data associated with moving the car. In this example, the data of interest is the date, time, longitude, latitude, and speed.

Each data obtained is filtered using the moving averages technique, in order to reduce noise. Then, the data is packaged according to the established parameters and passed to the distribution block. The distribution block takes the received data, adds information if this is necessary (for example, device id) and sends the dataset to the event detection block.

The third task of this stage is the detection of events. In this task, a set of patterns were defined. For example, for the speed change event the following states were defined: At rest, speed increase, speed decrease, constant speed, irregular speed and stop. These patterns allow to identify the type of events that can occur according to the received data set.

For example, if in multiple data sets received, the algorithm detects that the values represent speeding, a "speed not allowed" notification is generated. Notifications can be local or external. Local notifications only inform the user that generated the information, while external notifications send notifications to all users who are close to where the alert was generated.

5 Test and Results

This section presents the results obtained from the tests carried out on the proposed method. The tests carried out were carried out on a group of undergraduate and graduate students from the areas of computing, mechanics and electronics, who should use our method to build intelligent applications with an IoT platform. The applications that the students should build were focused on data collection, automatic event detection or visualization and analysis of historical data. The experimentation process was carried out in the following way: interview to identify the technical skills of the participants, and the development of an intelligent application. Participants should develop an intelligent application starting with the installation of a development environment with an IoT platform; later they should develop the application following with the phases of the proposed method.

The results of the experimentation showed the following data: All participants had basic skills in programming and algorithm design. All the participants knew the concept of IoT, but less than 30% have a clear idea of the concept and the technologies involved. Less than 20% of the participants have been involved in the development of an IoT project. Thus, we can affirm that the identification of free software tools can reduce the implementation time considerably. For example, using FIWARE's Generic QuantumLeap Component it is possible to deploy a data visualization and monitoring environment in less than an hour.

One of the smart applications developed was related to smart transportation. For the tests, three smart phones, a laptop and two virtual machines hosted in the FIWARE Lab Mexico node were used. The module for data acquisition and event detection was installed in the smartphones and the monitoring and visualization module and the database in the virtual machines. The events that we wanted to detect automatically were: speed not allowed and speed changes.

In turn, the following categories were established for speed changes: start of the march, the rest, the stop, the speed increase, the speed decrease and the constant speed. The technique used for the detection of excess speed was a mathematical comparison technique, while for the detection of speed changes three supervised learning algorithms were used: decision trees, Naive Bayes and randomforest. The detection results obtained are the following: speed not allowed 90%, speed changes with decision trees 93.86%, Naive Bayes 71.81% and randomforest 94.12%.

The results obtained show that the reuse of existing software components considerably reduces the complexity and development of IoT applications. However, this does not mean that developing IoT applications is trivial or easy. Instead, current methodological proposals and tools improve understanding and reduce the complexity of new software developments, in addition to considerably reducing software application development times.

6 Conclusions and Future Work

This paper presents a method for introducing IoT application development. The developed method was used to create a prototype that addresses an application in the domain of the internet of things and smart cities. The results obtained in the application tests allow us to infer that the use of the method for the development of this type of solutions is viable. Likewise, the results obtained during the period of application of the method, allowed us to reaffirm the initial idea that advances in the development and implementation of IoT and Artificial Intelligence components decrease the complexity in the development of this type of applications.

Therefore, this makes it easier for people with not so deep knowledge of the internet of things or smart cities to venture into the development of this kind of applications. The main contribution of this research work is the method developed and the case study presented. our example shows two scenarios recurring in the implementing of this type of solution: local processing and external processing. As future work, we are working on refining algorithms for use in other domains.

References

1. Majumdar, C., Lopez-Benitez, M., Merchant, S.N.: Real smart home data-assisted statistical traffic modelling for the internet of things. IEEE Internet Things J., 4662(C), pp. 1–1 (2020)
2. Yi, W., Li, S., Wang, B., Hoseinnezhad, R., Kong, L.: Computationally efficient distributed multi-sensor fusion with multi-bernoulli filter. IEEE Trans. Signal Process., 68, pp. 241–256 (2020)
3. Rafique, W., Zhao, X., Yu, S., Member, S., Yaqoob, I., Member, S.: An application development framework for internet of things service orchestration. 4662(C), pp. 1–14 (2020)

4. Sarbishei, O.: A platform and methodology enabling real-time motion pattern recognition on low-power smart devices. IEEE 5th World Forum Internet Things, WF-IoT 2019 - Conf. Proc., pp. 269–272 (2019)
5. Firmansyah, H.S., Supangkat, S.H., Arman, A.A., Giabbanelli, P.J.: Identifying the components and interrelationships of smart cities in Indonesia: Supporting policymaking via fuzzy cognitive systems. IEEE Access, 7, pp. 46136–46151 (2019)
6. Schoonenberg, W.C.H., Khayal, I.S., Farid, A.M.: A hetero-functional graph theory for modeling interdependent smart city infrastructure (2018)
7. Guzel, M., Kok, I., Akay, D., Ozdemir, S.: ANFIS and Deep Learning based missing sensor data prediction in IoT. Concurr. Comput., pp. 1–15 (2019)
8. Rueda, J.S., Talavera-Portocarrero, J.M.: Similitudes y diferencias entre redes de sensores inalámbricas e internet de las cosas: Hacia una postura clarificadora. Rev. Colomb. Comput., 18(2), pp. 58–74 (2018)
9. Ray, P.P.: A survey on internet of things architectures. J. King Saud Univ. - Comput. Inf. Sci., 30(3), pp. 291–319 (2018)
10. Venticinque, S., Amato, A.: A methodology for deployment of IoT application in fog. J. Ambient Intell. Humaniz. Comput., 10(5), pp. 1955–1976 (2019)
11. Shafiq, S.I., Szczerbicki, E., Sanin, C.: Proposition of the methodology for data acquisition, analysis and visualization in support of industry 4.0. Procedia Comput. Sci., 159, pp. 1976–1985 (2019)
12. Angelov, P.P., Gu, X., Principe, J.C.: A generalized methodology for data analysis. IEEE Trans. Cybern., 48(10), pp. 2981–2993 (2018)
13. Alberti, A.M., Santos, M.A.S., Souza, R., Da Silva, H.D.L.: Platforms for smart environments and future internet design: A Survey. IEEE Access, 7, pp. 165748–165778 (2019)
14. Shah, S.A., Seker, D.Z., Rathore, M.M., Hameed, S., Ben-Yahia, S., Draheim, D.: Towards disaster resilient smart cities: can internet of things and big data analytics be the game changers?. IEEE Access, 7, pp. 91885–91903 (2019)
15. Chen, Q. et al.: A survey on an emerging area: deep learning for smart city data. IEEE Trans. Emerg. Top. Comput. Intell., 3(5), pp. 392–410 (2019)
16. Alsrehin, N.O., Klaib, A.F., Magableh, A.: Intelligent transportation and control systems using data mining and machine learning techniques: A comprehensive study. IEEE Access, 7, pp. 49830–49857 (2019)
17. Liu, C., Nitschke, P., Williams, S.P., Zowghi, D.: Data quality and the internet of things computing. 102(2), pp. 573–599 (2020)
18. Cai, L., Zhu, Y.: The challenges of data quality and data quality assessment in the big data era. Data Sci. J., 14, pp. 1–10 (2015)
19. Home - Fiware-DataModels: https://bit.ly/3ltC5Hu (2020)
20. Place - schema.org Type: https://bit.ly/3lqm6Tf (2020)