

Designing a Project for Learning Industry 4.0 by Applying IoT to Urban Garden

Javier Hormigo, and Andrés Rodríguez

Abstract—The fast evolution of technologies forces teachers to trade content off for self-learning. Project-based learning (PBL) is one of the best ways to promote self-learning and simultaneously boost motivation. In this paper, we present our experience of designing a PBL course for Industrial Informatics. We provide all the designing details that could help other teachers to reproduce a similar experience. Furthermore, we report on the impact that this project has on student satisfaction and motivation. The new Industry 4.0 topic allows us to carry out complete projects, integrating different technologies and tools. Moreover, the selection of open-source and standard free technologies makes easy and cheap the access to hardware and software platforms used. Surveys taught us that tuning the courses towards developing real projects on the field, has a large impact on acceptance, learning objectives achievements and motivation towards the course content.

I. INTRODUCTION

THIS paper is an extended and enhanced version of the paper presented at TAEE2018 [1], which presents the implementation of a Project Based Learning (PBL) methodology in the subject of Industrial Informatics. Here we provide new details of the design of the course that can be very helpful for someone who wanted to reproduce a similar experience. Specifically, we present a detailed description of the deliverables that the student had to hand in and the rubric we provided in order to guide them to get a professional result. In contrast, other details described in [1] have been summarized to make room for the new content.

Industrial Informatics is a 60-hour course corresponding to the fourth year of Electronic, Robotic and Mechatronic Engineering degree in the Industrial Engineering School at “Universidad de Málaga”. Typically, there are about 40 students in the course. In this course, we introduce the new “Industry 4.0” [2] through concepts like Internet of Things (IoT), cyber-physical systems and cloud computing.

There are multiple reasons for selecting a PBL methodology [3][4]. First, taking into account the fast evolution of technology in this field, preparing the students for self-learning should be a priority for any engineering curriculum. Second,

being a last year subject, a more practical approach is sought. Third, it has been demonstrated that PBL greatly improves motivation of students [5][6]. This quality is very important since our students are typically reluctant to embrace self-learning. On the other hand, cooperative learning is also used, since nowadays teamwork skills are also fundamental for any engineer. Besides that, by working in groups, the students can accomplish a complete project and it also facilitates monitoring the work progress by the teachers.

To be really effective, the PBL methodology requires of a credible and tangible project[4]. We thought that an urban garden could be an ideal place to design a cyber-physical system by deploying several IoT nodes. These IoT nodes could gather data from the garden and also perform some actuation. The collected data could be also processed in the cloud and machine learning could be used to take the different decisions. Then, we arrange a cooperation with the Plant Biology Department, which holds an urban garden, to launch a project that could be useful for both, their students and ours. Furthermore, inter-discipline collaboration may be also an interesting way for enriching the experience.

Another important aspect of the project is the use of accessible open source software/open hardware platforms. The extensive community of developers who use these open systems offers a large number of online examples, tutorials, videos, etc. These are materials that our students prefer better than classic textbooks or reference manuals. In addition, the use of such accessible hardware platforms stimulates natural curiosity and the desire to try and to experiment on their own. Many of them show interest in acquiring these materials for using it at home on small do-it-yourself projects.

The rest of the paper is organized as follows: Section II describes the project assigned to the students including deliverables and the rubric for evaluation; in Section III, we summarize the planning of the course and the activities for evaluation; in Section IV, we show the results and discuss problems and possible solutions; finally Section V presents the conclusion of this work.

II. PROJECT DESCRIPTION

In this section, we summarize the project assignment and the different technologies involved. A detailed explanation of these matters, including the complete statement delivered to the students, can be found in [1]. Furthermore, we provide new detailed description of the individual deliverables that the student had to hand in, and the different rubrics which were used to guide them to complete the task successfully.

Manuscript received December 14th, 2018; revised February 13th, 2019; accepted March 14st, 2019 (Spanish version received March 27th, 2019; revised March 29th, 2019; accepted May 8th, 2019). Date of publication June, 2019; date of current version May 31st, 2019.

The authors are with the Department of Computer Architecture, Universidad de Málaga, 29071 Málaga, SPAIN e-mails: fjhormigo@uma.es, andres@uma.es

There exists a Spanish version of this paper available at <http://rita.det.uvigo.es/VAEPRITA/V7N2/A4.pdf>.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

DOI (Digital Object Identifier) Pending



Fig. 1. Urban garden at Science School facilities

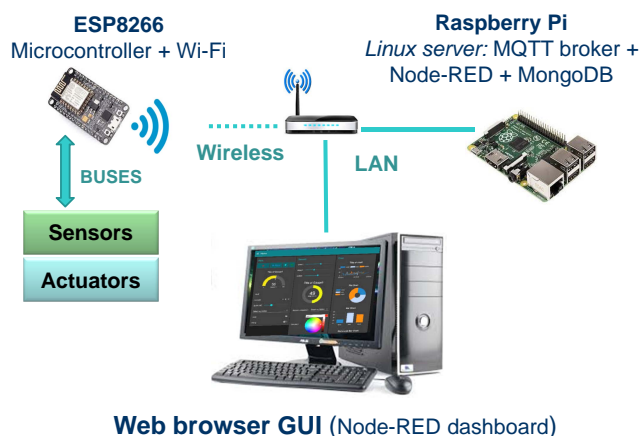


Fig. 2. Project system main architecture

A. Project statement

The project was introduced as a real collaboration with the Department of plant biology to improve their studies in their urban garden (Fig.1). The project consisted of designing and implementing a system for monitoring and control of urban gardening. This system should have included:

- IoT nodes based on the Wi-Fi module ESP8266: a "sensor node" to measure and transmit wirelessly fundamental parameters (temperature, humidity, light exposure...), and a "actuator node" receiving orders via the wireless communication network to control valves for irrigation and other actuators;
- A "base station", based on the Raspberry Pi, with a SCADA system and a database. It also serves as a message broker for IoT communication (using MQTT Protocol) and receives the information from the "sensor nodes", stores and upload it to the cloud;
- A graphical web interface for interaction with the users, which is hosted in the "base station" and programmed using Node-RED.

Along with the description of the system (see Fig.2), detailed functional and non-functional requirements are provided. Furthermore, the project statement contains a list of deliverables with their corresponding deadlines and their weights in the final score, and a list of available components in the laboratory.

B. Involved Technologies

The project is developed using several open-source software programs and open-hardware platforms:

- ESP8266 platform for IoT nodes. It is an open-hardware development board with a microcontroller and a Wi-Fi stack. The board is programmed using Lua scripts, thanks to the open-source firmware NodeMcu, that provides a complete set of ready-to-use modules [7]. The firmware is flashed using the development environment (ESPlorer IDE [8]) through the on-board USB interface.
- Raspberry Pi platform for base station. The Raspberry Pi boards are running the latest Raspbian operating system.
- Node-RED programming tool for the SCADA. Node-RED is an open-source flow-based programming tool [9]. It provides a browser-based editor for programming by wiring flows together using a wide range of nodes.
- MQTT protocol for communication between IoT nodes and base station. MQTT is a machine-to-machine (M2M) IoT connectivity protocol, based on messaging and using a publish/subscribe model. It requires installing an IoT message broker on the base station using the open-source MQTT broker Eclipse Mosquitto [10]. Within the messages, we chose using JSON (JavaScript Object Notation) [11] to format the data.
- MongoDB database. It is a free, open-source, No-SQL database, which stores data in JSON-like documents [12].

C. Detailed description of deliverables

Despite being students in their fourth years, our students needed a clear and specific guidance to keep the project development on track. Besides the project statement reproduced on [1], we provided them with a specific description of each deliverable and a rubric for its evaluation. This information was provided for both types of deliverable: prototypes and documentation. Our goal for the rubric were more formative than evaluative, and as consequence, the rubrics were more qualitative than quantitative. The deliverables were scheduled to help the student to reach the final deadline on time. In general, the documentation of a part of the system were deliberately programmed before the prototype of that part in order to make the students thinking ahead and planning. Below, we reproduce the most representative examples of deliverable specifications

1) *Technical Document of the Sensor Node*: This was the first thing they had to hand in. Following, the content of the document we handed out:

What should the document contain?

- The context and general summary of the operation performed by the node.
- The list of sensors that will be used, with a description and the reason to use them.
- The scheme of the electrical connections of the sensor node.
- A flowchart of the operations of the sensor node.
- The list of functions that will be used to communicate with the sensors, including input/output parameters and the operation description.

- List of functions that will be used to communicate with the base station, including input/output parameters and the operation description. Note: This part will be completed when we study the communication protocols.
- Other functions you think they are necessary for the correct operation of the node
- Other technical considerations not described above.

How will the document be evaluated?

- Well written and clear
 - Well: The document has no spelling mistakes. It uses simple language that is perfectly understood. The document is well edited: it has a nice appearance; the sheets, drawings, etc., are numbered; the different sections are clearly marked. It shows that they have taken it seriously.
 - Enough: The document is quite easily understood, although there is some room for improvement. I have found some errors, probably attributable to an oversight. It could have been better with a little more effort.
 - Insufficient: I have found several spelling mistakes, and I have not understood many of the things that are said in the document. The document is quite neglected. It shows that they have not worked hard enough.
- Adequate content
 - Good: The document includes a complete information on all the points that were requested. They also explain and justify briefly the decisions taken.
 - Enough: The document includes all the points, but the information is not complete in no more than three of them, or some point is missing. I see some obscure aspects in the description of some points. Some key decisions are not justified.
 - Insufficient: there are more than one point missing or the description is quite incomplete.
- Good approach and effectiveness
 - Good: The approach is good. I cannot think of any improvements. The scheme and operations are clear and reasonable. The functions that will be used are well described, with a clear comment on what they do. With this information, I believe that I would not have any difficulty to carry out the design myself.
 - Enough: I think the proposal is good, although it admits some improvements. I understand what they want to do, but if I had to implement it, I would have to ask for some additional clarifications.
 - Insufficient: The approach is not good. I do not understand the purpose or necessity of some of the elements, or some of the procedures and functions. If I had to implement the application, with this information, I would not know how to start.

2) *First prototype of the sensor node*: This was the first prototype. We specified the functional specifications and also more general desirable characteristics in this type of system through the rubric. Students are not used to think of these important features. Following the document we handed out:

What should it does?

This prototype must be fully functional from the sensors point of view, although it does not need to communicate with the base station yet. Therefore, it must have all the connections and be able to read the information of all the sensors at a given frequency and store and/or print said information.

What should be delivered?

- The updated technical documentation, including a section at the beginning with a list of the modifications, corrections or improvements made to the first version.
- The file (or files) for programming the sensor node.
- In class, it has to be demonstrated that the node works correctly.

How is it going to be evaluated?

- Correctness
 - Well: the system always works well with all sensors.
 - Enough: the system has ever failed on one or two sensors.
 - Insufficient: the system fails very frequently. It is clear that it is not right.
- Robustness
 - Well: the system does not freeze considering all the typical errors that may appear (disconnection of the sensor, the measured values are outside the logical range, ...). I have not gotten it to hang up.
 - Enough: It is reasonably robust. It was not easy, but in one or two cases it froze.
 - Insufficient: the system is not robust at all. It freezes frequently with typical mistakes.
- The code is well organized and commented
 - Well: The code is well organized. It is very easy to find the point of the system that has to be changed in order to make some modification in the functionality. Each function has an initial comment that explains what it does, and what the parameters are. In addition, especially complicated code points have a sufficiently clarifying comment. The variables, procedures and functions have names that help to understand what they are used for. The code is well indented.
 - Sufficient: It is reasonably well organized and commented, although in some cases, the structure of the functions could be better. The comments are sufficient, although I miss some further clarification in the code at some points.
 - Insufficient: The structure of the code has no logic, and there are no comments (or those do not clarify anything). The code is not well indented. I could not modify this code to add some new functionality or fix an error.
- Appropriate and updated documentation
 - Good: The documentation allows us to understand the operation of the system and to reproduce it easily. It has been properly updated from the previous version.
 - Enough: The documentation allows us to reproduce the system and more and less understand its opera-

tions, although there is some unclear or unspecified point. It has been updated with respect to the previous one, although some last minute changes are not registered.

- Insufficient: The documentation has a lot of dark spots that does not allow to reproduce and to understand the system or it has not been updated with respect to the previous version.

3) *Test, calibration and safety plan*: This documentation was proposed to make them think of aspects, which are usually present in real projects, but it is difficult to introduce them in the academic world. Again, we want them to think ahead and to plan before doing. Following we reproduce only the description of the documents, since the rubric is very similar to the one presented for the documentation of the sensor node: What should the document contain?

- Test plan of the sensor node. (What procedures I will perform to verify that the node is working correctly and is sufficiently robust).
 - Performance tests: it read sensors well, and sends and responds well to messages.
 - Robustness tests: it responds adequately to failures and does not freeze (the environment does not work as the node expected: a sensor is disconnected or sends erroneous data, an expected message is not received, ...)
- Test plan of the actuator node (What procedures I will perform to verify that the node is working correctly and is sufficiently robust).
 - Operation tests: it opens and closes the relay, and it sends and responds well to messages.
 - Robustness tests: it responds appropriately to failures and does not freeze (the environment does not work as the node expected: the relay is disconnected, an expected message is not received, ...).
- Test plan of the base station. (What procedures I will perform to verify that the base station works correctly and is sufficiently robust).
 - Functional tests: It sends and responds well to messages, stores data properly in the database, interacts correctly with the user ...
 - Robustness tests: Responds appropriately to failures and does not stay hung (the environment does not work as the base station waits Examples: an expected message is not received, the node sends data or erroneous messages, the user does unexpected things, ...).
- Test plan of the complete system. (What procedures I will perform to verify that the system works according to the specifications that were established and it is sufficiently robust).
 - Performance tests: it collects and stores the data correctly, it interacts correctly with the user ...
 - Robustness tests: it responds appropriately to failures and does not freeze (the environment does not work as expected: a node freezes, the node sends data or

erroneous messages, the user does unexpected things ...).

- Calibration plan (What procedures I will perform, or the end user will have to perform to assure that measurements obtained are expressed correctly in the units chosen).
 - Security plan: List of vulnerabilities in our systems (if someone wanted to bother us, what could they do to make our system not work properly) and possible solutions (ideas to improve the security of our system to avoid the vulnerabilities described above)
- 4) *Final prototype*: The information provided for handing in the final system. It includes similar points to previous deliverables, but the rubric also includes two new ones: one for self-evaluation of the project and another to evaluate the final system interfaces. Here, we only reproduce the new points of the rubric:
- Capacity to self-evaluation
 - Well: It clearly highlights the specifications that are met and the limitations and errors of their implementation. It also indicates the differentiating aspects of its design.
 - Sufficient: It does not indicate all the limitations of their system. There are some specifications that it is not clear whether they are fulfilled or not. The differentiating aspects of its design have not been sufficiently highlighted.
 - Insufficient: It has not been indicated enough what works and what does not work. Compliance with the specifications is diffuse and the representative aspects of its design have not been highlighted.
 - Friendliness
 - Good: The user has no doubt, at any time, about how to interact with the application, what data should be entered, and how to interpret the results and messages of the application.
 - Sufficient: The messages and information provided by the system are enough to work well. However, on some occasions I have had some doubts about what needs to be done or how to do it.
 - Insufficient: The user has constant doubts about what the application is asking for, and it is difficult to understand the results and messages on the screen.
- 5) *Demonstration-presentation of the final system*: The students had to deliver a final presentation. This is relatively new for many of the student. Hence, to help them, we provided a detailed description of what we expected from them in terms of both content and quality. Following, we reproduce the document:
- What should be presented in 10 minutes?
- Showing the operation from the user interface: configuration (sampling frequency); information obtained from the different sensors (current and stored in the database); other functions and operations.
 - Explaining if the defined standard for communication is met and if any extension has been made, what for; the measures taken to save energy, and for making the system robust.

- Showing and explaining, in general, how the node and the base station have been implemented.
- What things do not work properly yet and possible improvements.
- Answering the possible questions.

How is it going to be valued?

- It shows how the system works
 - Well: The presentation has been done in a way that shows which aspects work correctly and which are the limitations of the system. It highlights the specifications that are met and those that are not, as well as the differentiating aspects of their implementation.
 - Sufficient: The presentation has been made in a way that shows which aspects work correctly but does not clarify all the limitations of the system. It is not clear if there are some specifications that are not fulfilled. The differentiating aspects of its implementation have not been sufficiently highlighted.
 - Insufficient: The presentation has not sufficiently demonstrated what works and what does not work. Compliance with the specifications is diffuse and the representative aspects of its design have not been highlighted.
- Clarity of the explanation
 - Good: they have clearly explained each of the points and answer the questions adequately
 - Sufficient: they have explained well in general but a couple of points or questions have remained obscure
 - Insufficient: They do not explain clearly
- All the points are addressed in the stipulated 10 minutes
 - Well: They have touched all the points sufficiently in the stipulated time
 - Sufficient: They have been explaining too much some points and another one they have not been able to explain it. They have missed some points
 - Insufficient: They have not controlled time well and have not been able to finish. They have not explained several points.
- Understand and master what they are explaining
 - Well: Everyone in the group understands what they are saying and answers correctly to the questions
 - Sufficient: All of those who have spoken understand more or less what they are talking about but they have some doubt in some aspects. They have not answered some questions correctly. A member of the group did not control any point.
 - Insufficient: Several students have constant doubts, they do not dominate what they are describing or have done. They do not answer properly.

III. COURSE PLANNING, DEVELOPED ACTIVITIES AND EVALUATION

This course is organized in 15 weeks with 4.5 hours of in-person class. There is only one session per week. The students formed 3- or 4-student groups to develop a complete system. After working on more general topics for three weeks,

the project was developed in 12 weeks. For learning about each topic, except for some introductory exercises and an initial insights into each technology and tool, the students used mainly the examples, tutorials, and videos available online. These learning is organized through several jigsaws [13][14]. First, each topic is divided in four parts and each one is assigned to a student within each group. Before the next class, each student hand in a summary of their topic, including questions and doubts. In class, the students with the same part meet together in groups to became an expert in that part and to prepare a presentation. After that, the base groups meet together to explain each others theirs parts. Finally, an individual test is performed.

The evaluation of the course was defined following [15] in this way: 15% for handing in all deliverables on time (failing more than 20% of deliverables means not passing the course); 25% from a test of basic knowledge which is compulsory to pass with an A, but there are several sittings (this part is grade according to the number of sitting utilized); 50% from the grade of the project; and 10% from participation in class and within their own group.

In the first four weeks of the project, the students had to focus on developing the "sensor node". At the end of this period, they showed us the first prototype of the sensor node, which was able to read the information from a few sensors. For two weeks, they were focused on the communication between the sensor node and the base station. After a failed attempt for agreeing a standard syntax and protocol for all groups, we defined it. During the following three weeks, they developed the communication between the sensor node and the base station using the messages and very simple protocols defined in our standard. At the same time, they used node-RED at the base station to process the received messages from the IoT nodes and send others messages back. At the end of this period they had to hand in a prototype of the system which was peer-reviewed in two different ways: using the sensor node and base station of the same group, and also using the sensor node of one group with the base station of the group that was evaluating.

After Christmas holidays, they had the first sitting of the test of basic knowledge. This test consisted in a few very simple theoretic questions and basic programming exercises using Lua and node-RED. On the other hand, they had two weeks to finish up the project. In the second last week of the course, they presented their projects and showed the working prototype in front of the teachers and their classmates. They also had to answer some general questions about their designs. In the last session, we asked for an extension of their own project on an individual basis. We proposed the same extension for all students and they had to suggest the required changes in both the IoT node and the base station. This exercise was performed in class as an exam. Although this exercise seems completely individual, it should actually encourage to help their group-mates because there was a component of the mark which depended on the fact that all the members of the group pass the exercise. In practice, if this happened, they double the mark. However, only two groups achieved this goal.

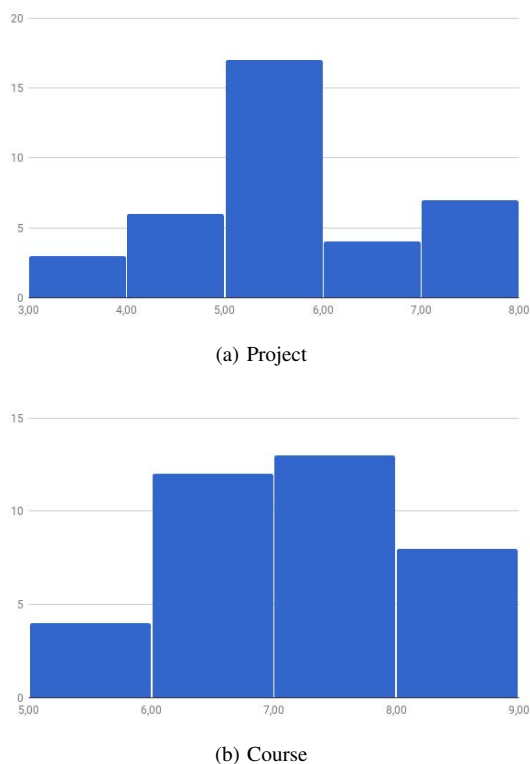


Fig. 3. Distribution of the final marks obtained for the project and the course

IV. RESULTS AND DISCUSSION

The results have been bittersweet. Fig. 3a shows the distribution of the marks obtained in the project for each student. Most of the students handed in just acceptable projects and even one group a borderline one. Furthermore, the marks of the projects went down because only a few of the students accomplished the individual extension.

After including the rest of components, most of the students improved their marks. Fig. 3b shows the distribution of the final marks obtained in the course for each student. Only about 50% of student passed the basic test in the first sitting, although, in general, the rest of students only failed in one part. About 40% of them passed the test in the second sitting and only 3 students (8%) passed the test in the third sitting. In the end, none failed the basic test.

Only 2 students withdrew from the course. One of them failed ill and had to quit. In both cases, that happened at the beginning of the course. There is also one student that never attended a class.

Another interesting information is the number of hours out of class that each student used in this course. Near to the end of the course, we asked them to estimate anonymously the mean of the hour per week they had spent in this course at home. The result of the survey is shown in Fig. 4. According to the number of ECTS, they should have spent about 6 hours per week and most of them said to use between 4 and 8 hours. Even so, the mean, 5.2 hours, is a little low and extreme cases are too disperse for being satisfactory. Hence, some measures should be added to correct this in next courses.

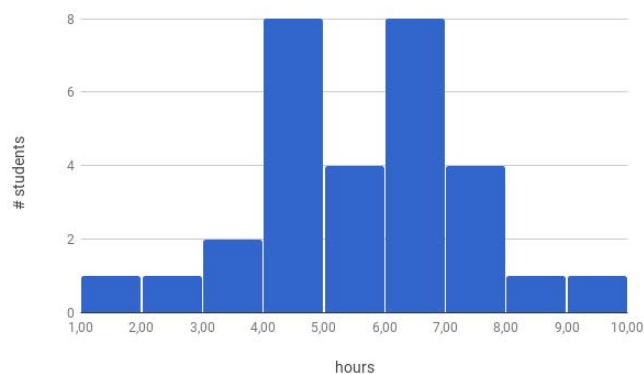


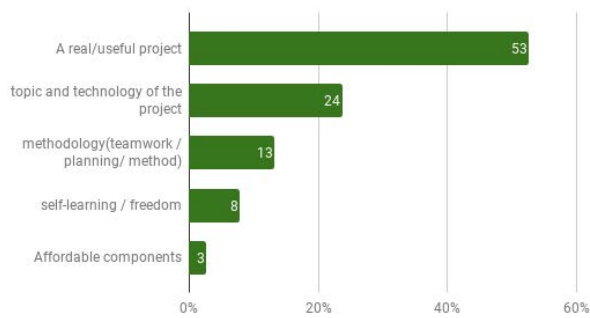
Fig. 4. Distribution of the number of hours out of class per student

A. Student perception

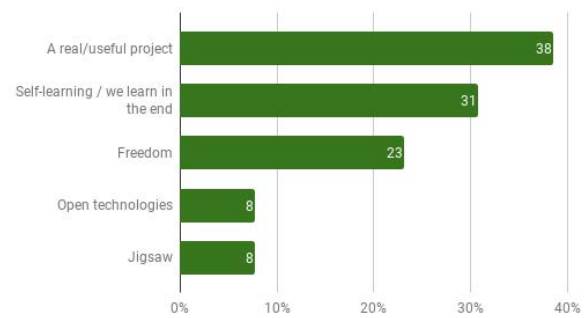
We have conducted three quick surveys along the course. The student had to write anonymously the best and the worst aspect of the course so far in two minutes. The first survey was conducted after providing them with the project statement. They had time to read the document and discuss the project with their teammates. The results of the survey are summarized in Fig. 5. In Fig. 5a, it is easily observed that they were excited about the project, especially with the fact that it sounded (and it was) a real project. They appreciated that the project could be useful to someone. The topic and technologies required also appealed to them, and to a lesser extent the methodology and freedom. However, they were also worried about the amount of work required to develop the project and the necessity of learning new technologies. It is alarming that 25% of student thought that "learn new content" is the worst thing of the subject, when they supposedly enrolled in this course to learn this content. On the other hand, almost 20% of the students do not see any negative aspect.

The second survey was conducted after working on the project for several weeks. The results are shown in Fig. 6. The students were working on the sensor node using Lua as the programming language. Most of them were having important difficulties to debugging their codes. The project was still in the first position as the best of the course, but self-learning was close behind, being claimed by 31% of the students (see Fig. 6a). In contrast, another 33% of students considered self-learning and the lack of enough help as the worst of the course (see Fig. 6b). That could be explained by the fact that the first group of students had overcome the steps described in [16] but the second group had not. They consider that the second important problem is the fact that they could not bring the equipment to home and the lack of time in the laboratory. This topic was a continuous source of complaint.

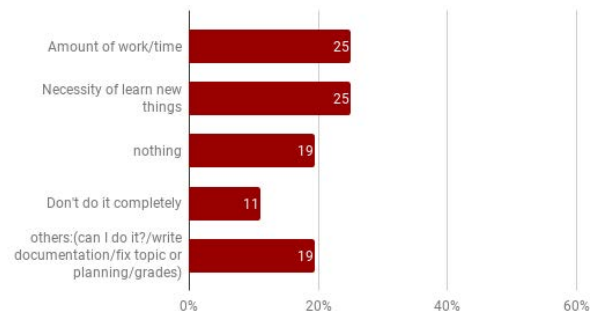
The results of the third survey are shown in Fig. 7. It was conducted after finishing the project and before defending it in front of the teachers. As in the previous surveys, working in a real and useful project was the most appreciate characteristic of the course, claimed for about 40% of the students (see Fig. 7a). Furthermore, the topic and technologies involved in the project were claimed by 16% of students. Self-learning was



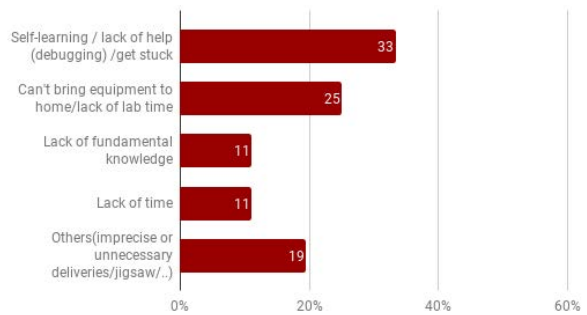
(a) Best



(a) Best



(b) Worst



(b) Worst

Fig. 5. Student perception after giving them the project statement

Fig. 6. Student perception after working on the project several weeks

the second one with 22% of votes, but there are more students (28%) who thought that self-learning was really the worst of the course (see Fig. 7b). This was only slightly outnumbered by the idea of not being able to bring the equipment to home or not enough lab time. Another important group of students (22%) think that the planning was not good or some activities made them waste time, like surveys. Surprisingly, only a few students (9%) thought that it was too much work to cope with.

B. Teacher perception

From our point of view, students were very reluctant to read the documentation we provided with. In many cases, when they were solving an exercise, they simply put together several examples without adjusting them to their own problem. They abuse the use of "trial and error", and they don't want to use any methodology to debug their codes. Furthermore, students had serious difficulties to understand the way Lua works, based on events. They keep trying to write functions sequentially, thinking wrongly that until one function is not completely finished, the next it is not going to be executed.

It is also very difficult to make them follow written instructions properly. As shown in Section II, we provided exhaustive and precise instruction for each document or code they had to hand in, including the format and the rubric for evaluation. However, they fail following simple rules, even when they know they are going to be evaluated based on these rules. Despite of that, we think these rubrics have been very useful, because, in other courses where we did not used them, the results were much worst.

Regarding a more practical issue, we have found some difficulties in evaluating Node-RED flows designed by the

students. When you have 40 individual solutions proposed by students and presented as exported flows in JSON format, you have a tough work ahead to test them. You have to manually import every flow in your Node-RED programming tool, evaluate the design and then the functional aspects. We have found a very practical solution for next year: designing a Node-RED flow to automatically perform the functional testing of the student's designs. We can test the answer of the student design and store on a collection in the database the results for every student assignment, making easier the evaluation process.

C. Required Improvements

It is clear that the students really like the project and this, along with teamwork, encourages them to follow the course. However, some adjustment is required to improve their experience and the quality of the learning (the project). First, they need specific material to learn how to debug systematically a program (or system). More progressive exercises for each technology have to be introduced previous to work directly in the project, especially for Lua. We should control better their individual progress in a week term basis. To do that we are planning to use an individual diary, where they will have to write down the tasks they work on and the hours used.

Another important issue we should address is the problem of bringing to home the prototype components. This year we did not allow them to do it mainly because, in our opinion, they had a lot of work to do at home. They had to read the documentation we provided, to plan the hardware/software architecture, to design the functions, to generate the asked documentation, etc. In other words, they had to think before

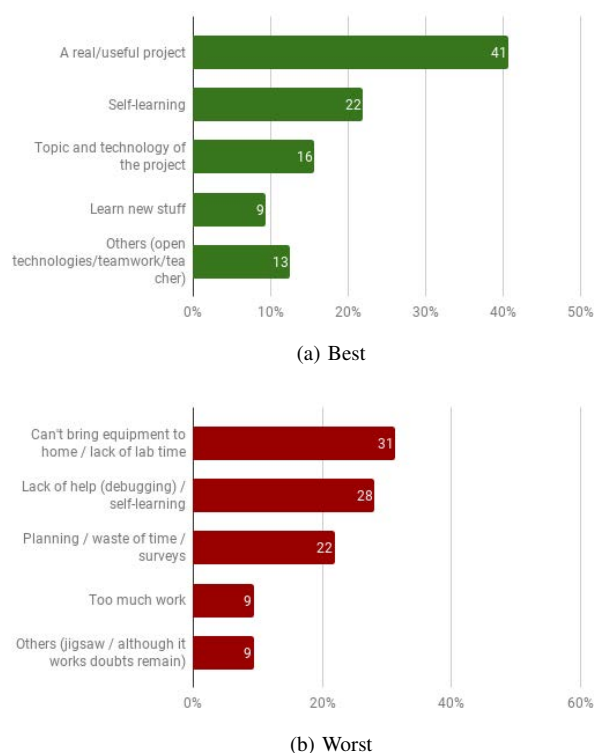


Fig. 7. Student perception after finishing the project

doing. However, in their way of working (trial and error), those matters were a waste of time. There were also other reasons, such as preventing the situation of not being able to work in the lab because they forgot to bring the equipment back. Furthermore, if they really thought they need it at home, why did not buy it if it cost about 5 euros? We think that this complaint was more a self-excuse than a real problem. The easiest solution would be to let them bring the material to home but we do not think that it going really to help them. We better have to find a way to reinforce the task of planning and thinking ahead.

We think that utilizing open-hardware and open-software was a good decision and students appreciated it. However, taking into account that our students have serious problems with Lua, we are considering to shift to a more traditional way of programming using C through Arduino environment. They are used to these tools and it may be easier for them.

Despite all difficulties, we think that PBL is the better way of confronting the challenge of promoting self-learning and boosting motivation. We will need several cycles to adjust the planning and small details of the methodology, but we think we are in the good way.

V. CONCLUSION

This paper presents the design of a PBL course for Industrial Informatics. All designing details are provided to facilitate reproducing a similar experience. We observed a great positive impact on the student satisfaction and motivation, although we had to confront the students' reluctance to face their learning process autonomously. Above all, we are satisfied with the

initial results obtained and that leads us to keep on improving the PBL course design for next courses.

ACKNOWLEDGMENTS

We want to thank Prof. Miguel A. Quesada Felice and Antonio J. Matas Arroyo for their helpful advising on the Urban Garden topic. This work has been supported by project TIN2016-80920-R, from the Spanish Government and by Project PIE17/085 from Universidad de Málaga.

REFERENCES

- [1] J. Hormigo and A. Rodríguez, "Project based learning on industrial informatics: Applying iot to urban garden," in *2018 XIII Technologies Applied to Electronics Teaching Conference (TAAE)*, June 2018, pp. 1–9.
- [2] A. Gilchrist, *Industry 4.0: The Industrial Internet of Things*, 1st ed. Berkely, CA, USA: Apress, 2016.
- [3] L. Gil-Sánchez, R. Masot, and M. Alcañiz, "Teaching electronics to aeronautical engineering students by developing projects," *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, vol. 10, no. 4, pp. 282–289, Nov 2015.
- [4] A. F. Jiménez López, M. C. Prieto Pelayo, and . Ramírez Forero, "Teaching image processing in engineering using python," *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, vol. 11, no. 3, pp. 129–136, Aug 2016.
- [5] J. Larmer, J. R. Mergendoller, and S. Boss, *Setting the standard for project based learning: a proven approach to rigorous classroom instruction*, 2015.
- [6] J. Rodríguez, I. Esparragoza, J. R. Ocampo, and R. Viganò, "Study of motivation of engineering students on multinational design projects," *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, vol. 12, no. 4, pp. 218–226, Nov 2017.
- [7] (2018, Jan.) NodeMcu connect things easy. [Online]. Available: http://www.nodemcu.com/index_en.html
- [8] (2018, Jan.) Integrated Development Environment (IDE) for ESP8266. [Online]. Available: <https://esp8266.ru/esplorer/>
- [9] (2018, Jan.) Node-RED. [Online]. Available: <https://nodered.org/>
- [10] (2018, Jan.) Eclipse mosquito. [Online]. Available: <https://mosquitto.org>
- [11] (2018, Jan.) JSON lightweight data-interchange format. [Online]. Available: <https://www.json.org/>
- [12] (2018, Jan.) Welcome to the MongoDB Docs. [Online]. Available: <https://docs.mongodb.com/>
- [13] M. A. Kousa, "Jigsaw cooperative learning in engineering classrooms," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, March 2015, pp. 58–62.
- [14] F. Martinez-Rodrigo, L. C. Herrero-De Lucas, S. de Pablo, and A. B. Rey-Boue, "Using PBL to improve educational outcomes and student satisfaction in the teaching of DC/DC and DC/AC converters," *IEEE Transactions on Education*, vol. 60, no. 3, pp. 229–237, Aug 2017.
- [15] M. Valero and J. Domingo, *Técnicas de Aprendizaje Cooperativo y Aprendizaje Basado en Proyectos*, 2017.
- [16] R. M. Felder, "We never said it would be easy," *Chem. Engr. Education*, vol. 29, no. 1, pp. 32–33, 1995.



Javier Hormigo received the M.Sc. and Ph.D. degrees in telecommunication engineering from the Universidad de Malaga, Spain, in 1996 and 2000, respectively. He joined the Universidad de Málaga in 1997, where he is currently an Associate Professor with the Computer Architecture Department. His current research interests include computer arithmetic, specific application architectures, and field-programmable gate array.



Andrés Rodríguez received the M.Sc. and Ph.D. degrees in Computer Science from the Universidad de Malaga, Spain, in 1995 and 2000, respectively. He joined the Universidad de Málaga in 1996, where he is currently an Associate Professor with the Computer Architecture Department. His current research interests include high productivity programming models, parallel programming, heterogeneous computing, and multiprocessor architecture.