QuadLab

A toolkit for project-based learning in accordance with international standards in automation and robotics engineering

David Zapata, Mario Garzón, João Valente, Antonio Barrientos

Abstract It is frequently claimed that the students must have an active role in building and transforming their own knowledge, and the teacher's labor is to provide the students the necessary tools in order to reach specific learning objectives, included in a course program. This paper presents an aerial robotic system as a toolkit, and proposes a series of activities focused on the learning in automation and robotics. These proposed activities have been designed based upon the project-based learning methodology, and they facilitate the achievement of the learning objectives presented by CEA/ISA and satisfy the international standard ABET. The toolkit and the activities are oriented to impulse the practical teaching, giving the student additional motivation and, in consequence, improving his or her active role. Besides, the toolkit and the activities give the teacher a tool in which it is possible to assess the students learning process.

Keywords Aerial robotic platform · Practical teaching · Project-based learning · ABET

1 Introduction

Due to the rapid dissemination and interest in mini Unmanned Aerial Vehicles (MUAV) that have been observed in the last few years, the number of students that show interest in working in related fields is increasing. However, most of them are undergraduate students, who usually do a short-term stay in the research groups, no longer than an academic semester in most cases at our institution. Furthermore, they are often students from final courses who also have subjects pending, reducing significantly the effective time for developing the projects. In order to optimize the work performed by those students, a series of activities focused on the learning in automation and robotics, and based on the use of an aerial robotic system is presented, those activities have the objective of allowing the student to acquire basic knowledge and capabilities, fulfilling international standards, and also to learn how to control and operate the MUAV.

The methodology used for the aspirant researchers during the past years is as follows: First a task is proposed to the student, then he or she starts to learn how to use the materials and the tools available to carry out the task proposed, and finally he or she starts to work based in a schedule until the goal is reached. Their role in the projects range from low to medium level tasks, such as low-level interfaces, on-board sensors setup processing boards setup; medium-level programming, guidance, navigation and control approaches, among others.

When working with MUAVs the student must also understand the vehicle principles, how to steer it when working on manual mode, also how to send and receive data and commands to the vehicle and finally how to change the setup and perform maintenance. Learning and acquiring expertise in these tasks is often very laborious and requires a lot of time because the students have to overcome many problems during the development and testing of those systems. Taking this into consideration, it would save a considerable amount of time and also improve the final results if a student starts to work with the basic knowledge and understanding of the MUAVs.

The proposed system is oriented to fill this gap in the automation and robotics (AR) engineering program at our institution, providing both students and teachers with a toolkit and a series of activities that are able to complement the necessary knowledge to reach international standards proficiencies in engineering education. In addition, the use of the platform allows the teacher to emphasize in specific academic goals, and can be used by a student without previous experience.

These proposed activities have been designed based upon the project-based learning methodology, following the model suggested by the Northwest Regional Educational Laboratory[17]; and at the same time those activities meet the competences presented by CEA/ISA (Spanish automation committee and the international society of automation)[4] and the international standard ABET [1].

Project-based learning (PBL), in addition to its systemic nature and ability to work both with vertical skills (in our case control, computer technology, electronics -in many of its fields-) and horizontal (teamwork, ability to design experiments, design capacity, creativity, multidisciplinary skills, use of generic resources engineering, planning work), fits well into the design of this learning platform, since this toolkit is divided into small problems or projects that come together to be part of a real application.

Furthermore, the proposals are fully open to be approached by the teacher to the knowledge or skills in which it is desired to emphasize, either by the type of discipline to be targeted or the level of complexity required, as of course, the student must have cognitive foundations that give a starting point to explore possible solutions. This kind of flexibility in the formulation of the problem requires that the platform is modular, so the teacher can include or omit information that he or she gives to the student.

The aerial robotic system is an integrated system with sensors of different types, and with the ability to add others, this means having a multidisciplinary system that can have real application in different fields, such as electronics, embedded systems, control, aeronautics, and robotics. The system is exposed to variations in its environment, such as noise and perturbations; this makes the student to deal with an additional complexity that makes a clear difference between the theoretical concepts and real world. Also, the toolkit treated as a whole system presents a high complexity, but treated as subsystems, the complexity may vary. This allows a wide range of difficulty in the proposed practices, ranging from simple (e.g. linear mono variable) to the study of the whole system. It is also a system that can be approached from different areas of knowledge.

It is a low cost system, which has a very good relationship between cost and performance. This makes it easy for inexperienced users to gain experience without significant economic consequences, and to face these platforms with greater confidence. The type of the system used is highly attractive to new users, as it is a product that uses the latest technology for both academic and entertainment applications, has been increasing in recent years. Being an air system that does not require a fixed base station, you can perform all activities in different geographical areas, eliminating the space and time constraints inherent to traditional methods, likewise promoting outdoor engineering practices.

Finally, the aerial robotic system can be used in various real applications. This brings the student to a training oriented to its future professional activities. It is necessary that the proposed activities will allow the student to respond to the challenges posed to national and international level in the training of engineers. This can be ensured through the application of standards and norms. It is also necessary to have an integrated modular system, such as the activity that the user addressing is not disrupted by obstacles of a technical or management of additional tools necessary for the development of the practice.

This paper is distributed as follows: first, section 2 presents related educational projects in AR engineering field. Then, section 3 establishes the framework in which *QuadLab* is used in the learning process, as well as the definition of the scope in the methodology and standards. Afterward, section 4 exposes the toolkit, describing all elements and showing how it could be used. Finally, after knowing how *QuadLab* works and defining the methodology to be used and the standards to be met, section 5 suggests a series of projects that could be developed with *QuadLab* and shows how those projects satisfy the learning objectives.

2 Related Educational Projects

The availability of practical courses and practices during the formation of an AR engineering student are important to gain experience and understanding of the real systems. It is therefore, the way as pedagogues establish a bridge between the theoretical foundations of autonomous systems and their realistic assessment. Many universities and educational centers made an effort to provide such components and systems in different contexts and backgrounds, ranging from classroom laboratories, contests, to related initiatives.

Nowadays, many AR engineering courses are programmed to be lectured both in classroom and laboratories. Probably, the most common case are control system courses. Herein, the students have the opportunity to study dynamics and control through miniaturized process plants, simulations, or other simplified system built to this end, e.g., [9,19,12]. A good overview about three different control laboratories approaches is also given in [13].

Another outstanding methodology is PBL where students can learn how autonomous systems work - literally speaking - by doing. For instance, courses using

Lego Mindstorms robots as a training platform are reported [15,5], project-learning through robotic contests can also be found in [18,8,2].

For what teaching with aerial vehicles is concerned, there is not much work found about it. One communication from the MUAV team from the University of Applied Sciences Technikum Wien is reported in [10]. The goal for the students is to build an aircraft from the scratch. the motivation, design and development of the system is described.

This work presented herein can be distinguished by the fact that a very well known and affordable platform is used. Therefore, there the student start from an already functional base were it is possible to observe and study some behavior. Moreover, the challenge to work with both off-the shelf, and built from the scratch systems was considered. Being able to work with heterogeneous systems and find out a solution to integrate them in single system is a challenge. Finally, all the activities proposed herein respect a standard. Indeed, they were designed in focusing in the ABET standards for engineering. For the best of our knowledge, our work is one of the first to contribute with a novel understanding of mini quadrotors through education. We believe that is a step forward to the future of AR engineering students.

3 Learning Elements

This section exposes the basic course learning elements and then describes how this work is part of it. First it is fundamental to describe how the courses are designed; Felder and Brent[6] describe three general domains to be covered: *Planing*, *Instruction* and *assessment*. Planing is about to identify and define the learning objectives, Instruction is the way or methods that help the student to reach the learning objectives, and assessment refers to the procedure of determining how well the methods lead to a successful achievement of the learning objectives.

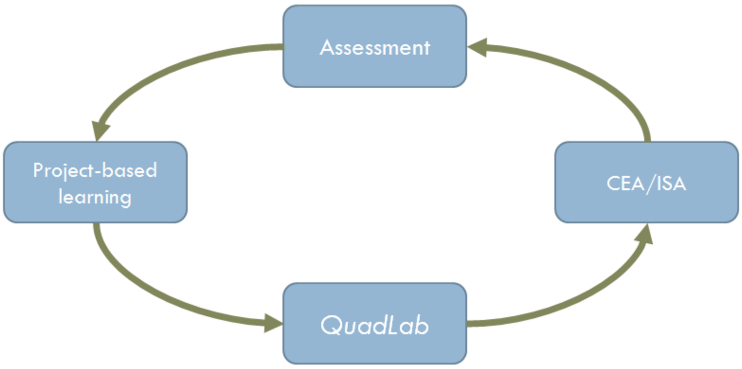


Fig. 1: *QuadLab* Learning scheme

In order to put *QuadLab* into this scheme, there must be characterized each element described before (see figure 1). First step is to define the planing, for this work the learning objectives are given by CEA/ISA, then for achieving these learning objectives it is necessary an instruction or methodology, and here is where the PBL makes use of the robotic platform to address the learning process. Finally, it is necessary to assess how well the learn objectives are achieved by the student. This assess closes the learning cycle, producing a continuous improvement by giving feedback to the methodology and suggesting if the instructions need to be modified in order to obtain a better achievement of the learning objectives. The assessment, besides is an important part in the learning cycle, it is not part of the scope of this work. However, as good evaluation tools that fit well into the PBL and therefore this toolkit, there are: Portfolios, written project reports, oral presentations, memos, interviews, concept maps, among others. The use of multiple assessment methods improves the evaluation results [6], and also the student could do a better self-evaluation, team-evaluation and methodology evaluation. At this point it is important to remark that the toolkit is not strictly linked to the PBL methodology or CEA/ISA learning objectives. The toolkit is totally open to modifications and could be used with other methodologies such as cooperative learning or traditional laboratory (where all the activities are pre-established)

3.1 Project-Based Learning

Project-Based Learning is an alternative to traditional methods of teaching based on the comprehensive development of a project. This project will aim to solve a problem posed by teacher and requires that the student finds resources and then develop activities to solve the problem. This type of training potentiates the binding between *knowing and doing*, as students should address the concepts as they are required for project execution.

Mills et al. [14] make the distinction between the terms *project* and *problem*. PBL typically take more time to complete, besides they are more focused on the application of knowledge, and Problem-based learning to acquire knowledge. Engineering projects in the short term may require a single area of engineering, but the long-term projects require multiple areas and composition of groups with individuals specialized in different areas. As it can be seen, the projects are more related to a professional environment, increasing social skills, such as cooperative learning.

3.2 Standards For Education in Engineering

Two main standards have been studied, the first one is known as *Engineering Criteria 2000* or EC2000. It has been crafted by the Accreditation Board for Engineering and Technology (ABET)[1] as the criteria that should be assessed by the engineering programs in order to obtain the accreditation. The EC2000 specifies 11 learning outcomes, oriented to both technical and professional skills, the list of outcomes is presented in table 1. The second standard was developed by the Spanish Committee for Automation (*Comité español de automática* - CEA) in cooperation with the International Society of Automation (ISA). They have elaborated a document outlining the competences that an student of the technical

	EC2000 learning Outcomes
a	An ability to apply knowledge of mathematics, science, and engineering.
b	An ability to design and conduct experiments, as well as to analyse and interpret data.
c	An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
d	An ability to function on multidisciplinary teams.
e	An ability to identify, formulate, and solve engineering problems.
f	An understanding of professional and ethical responsibility.
g	An ability to communicate effectively.
h	The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.
i	A recognition of the need for, and an ability to engage in life-long learning.
j	A knowledge of contemporary issues.
k	An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

Table 1: List of learning outcomes required by the EC2000 criteria.

Industrial Engineering degree should acquire to fulfill the industry requirements regarding the automation and control area[4].

The reason for selecting those two standards is as follows: The EC2000 is probably the most widely used criteria for international accreditation in engineering programs, therefore its relevance is without question. Nevertheless, the outcomes that are pointed out by that criteria are very generalist making it more difficult to use them as a direct criteria to propose a laboratory project. The CEA/ISA guidelines are used mainly known and used in Spain, but, in contrast with the EC2000, those proposed competences are much more specific and punctual, and they are directly oriented towards the learning of automation and control. This allows to target these competences in a more direct way using laboratory activities.

However, the best results will be obtained if both criteria are aligned, in order to do so, an relationship between the competences of CEA/ISA and the required outcomes pointed out by ABET must be studied. The result obtained will be highly helpful for the e efficient design of the activities proposed in section 5.

	CEA/ISA Competences	**EC2000 Outcomes**
A	Knowledge about fundamentals of automation and control methods	a, e
B	Knowledge and skills for modelling and simulation of systems	b, e
C	Knowledge on automatic regulation and control techniques and their applications in industrial automation	a, b, k
D	Knowledge of the principles and applications of robotic systems	a, e, k, d
E	Applied knowledge of industrial informatics and communications	b, e,
F	Capability to design control and industrial automation systems	b, k, c

Table 2: Relationship between CEA/ISA competences and EC2000 outcomes.

Table 2 shows the relationship between the competences from the CEA/ISA guidelines and the outcomes required by the EC2000 criteria. As it can be observed, the fulfilling of each CEA/ISA competence can help obtaining one or more of the EC2000 outcomes. Since the CEA/ISA competences have a technical focus, even if all of them are obtained, not all of the EC2000 outcomes will be covered, specifically the points *f,g,h,i,j*. This points however can be partially approached using complementary methodology such as team work, documentation, evaluation and oral presentation of the work carried out by the student as well as the results and conclusions that they can obtain from it. It should also be pointed out that the relationship may be subjective and depend on the specific objectives that may be proposed in each project or activity. Moreover each CEA/ISA is subdivided in several points, and the laboratories can only target some of those points.

4 Robotic platform *QuadLab*

The aerial platform base kit involves two mainly parts, the MUAV and the ground station control GCS. The MUAV used to this laboratory is a quad-rotor type because of its stability, safety and controllability; The model adopted is a low cost AR.Drone Parrot.

AR.Drone's prefix AR comes from *"Augmented Reality"* , that means it is designed for gaming and interactivity [3], in other words, is a product that is not designed for civilian or military applications but for fun and entertainment, and it is therefore using low-cost sensors (and which is economical in the market).

As a commercial project, issues like price, safety, ease of use and repair are very important, and with the quad-rotor inherited characteristics, fit accurately in academia. Seeing that it is easy to use and designed for a mass audience, does not require the students to have any experience, and that somehow generates confidence regarding security concerns. For the low cost of the robot and its parts, it is an ideal tool for testing, since it is intended for a student to acquire knowledge from tests and experiments (and this increases the chances of failure and/or errors), it would be easy economic and replace affected parts, which has an additional advantage, as it increases the level of student interaction with the robot.

All these features and its high stability make obvious that the student will be more focused on the objectives for practice and have not to worry about technical issues deeper (low-level control, communication drivers, data acquisition) or different from those that are required for the preparation of laboratory activity. This section gives a review of both hardware and software *QuadLab* components. For more detailed information about the whole system refer to [20].

4.1 Mini-UAV

The MUAV has a weight between 380 and 420 grams depending on the hull type used; It can fly at a maximum speed of 18 meters per second and a fly autonomy near to 12 minutes. Its on-board computer system is a processor ARM9 RISC 32-bit 468 MHz with 128 MB DDR RAM memory, Linux OS, and it is communicated via Wi-Fi whose scope on outdoors could be between 50 and 100 meters.

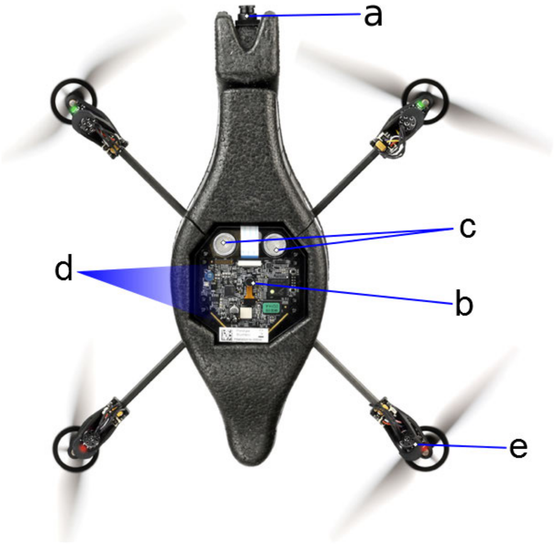


Fig. 2: AR.Drone hardware

It is equipped with an ultrasonic sensor for altitude measuring with a range of 6 meters (Figure 2.c); an IMU of 3 axis accelerometer, a 2 axis gyroscope (pitch and roll) and a single axis gyroscope of precision (yaw)(Figure 2.d); and two video cameras, one looking horizontally with 640x480 pixels of resolution, 93 wide-angle diagonal and a 15 fps frequency (Figure 2.a) and the other looking vertically with 176x144 pixels of resolution, 64 wide-angle diagonal and a 60 fps frequency (Figure 2.b). Although the cameras have a very low resolution, they allow to process the recognition of the main objects in the environment and to measure altitude and velocity for its own hovering control among other features. On the safety side, the AR Drone has an automatic locking of propellers in the event of external contact and an emergency state that stops all the motors (Figure 2.e).

Using a Wi-Fi ad-hoc connection, through UDP/TCP ports, the MUAV sends navigation data, status, and the images captured by the cameras (one at a time), and receives control commands and configuration parameters. For more details, refer to [3].

4.2 Enhanced System

An electronic circuit has been added in order to improve the MUAV capabilities, expanding the number of applications and providing more controllability and robustness to the MUAV, as well as the capacity to add more advanced laboratories. This circuit was designed to collect data from one (or multiple) external sensor (e.g. a GPS and/or an altimeter) and send to GSC through a wireless connection.

The final MUAV has been endowed with an external wireless GPS system, which adds location information (latitude, longitude and altitude). The addition of a GPS allows knowing the absolute position and programming the MUAV to return to base (taking advantage of the MUAV's automatic taking off and landing features), among others capabilities. Figure 3 shows two different prototypes of enhanced MUAVs, using different brands of GPS. In order to have the drone weight between the original range, the hull is removed.

(a) Prototype 1

(b) Prototype 2

Fig. 3: MUAV prototypes used for the development of practices.

Figure 4 shows the final configuration of the aerial platform; In the marked *UAV* box are the hole AR.drone system and the additional plug-in mentioned, that comprises GPS and communication unit. The GSC box contains another communication unit (which is paired with the one in the MUAV). This unit sends all information in transparent mode to a software application, who collect data and integrates all the telemetry of the MUAV and works as a user interface.

To send the measured data to the base station, the XBee wireless modules, which have an average range of 40 meters indoors and 120 outdoors in direct line of vision are used, this range may vary depending on the type of antenna that has the module. They are characterized by low power consumption and low cost, as well as the GPS receiver, it have a communication interface UART by means of which are configured. Additionally, they have digital inputs and outputs, and auxiliary analog inputs.

4.3 Ground Control Station

Besides the MUAV itself, the robotic platform must have a GCS which works not only as the interface between the MUAV and the operator, but also as a data collector for analysis or study purposes. The design of the GCS implies essentially two modules, one to communicate with the MUAV (send, receive and collect data) and another module to communicate with the user (i.e. graphical user interface GUI). It is evident that both modules are necessary in order to use the robotic platform, but the level of complexity of each one depends of the target that will be presented to the student. The fact that there could be different levels of complexity gives the teacher the flexibility to manage the difficulty of the assignments.

In order to establish communication with the AR.Drone, there is a software development kit (SDK) provided by Parrot. By mean of this SDK, it is possible

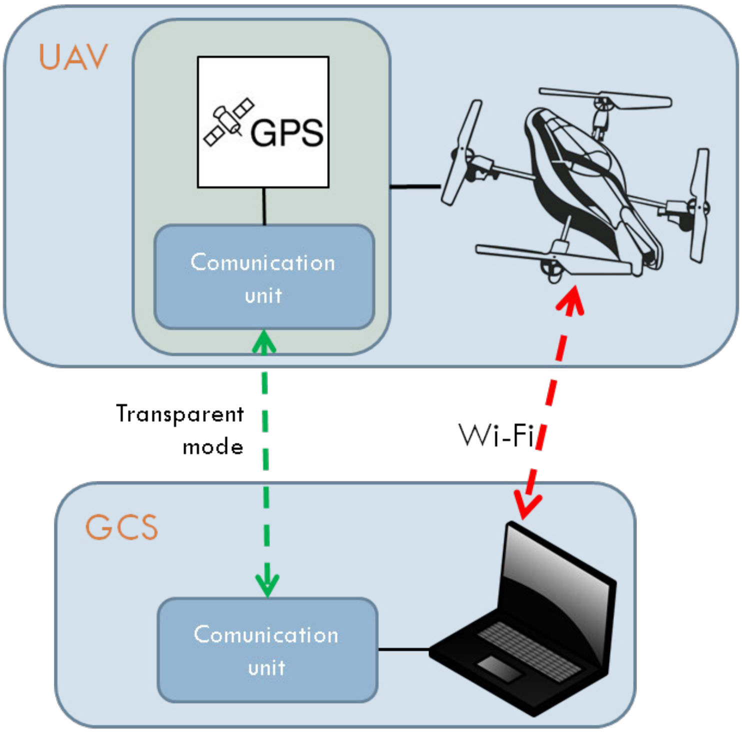


Fig. 4: Overall system diagram

to use the TCP/UDP ports to access and control remotely the drone. There is a port dedicated for any service described above (Navigation data and status, video streaming, control commands and configuration data). The AR.drone SDK also features pattern recognition and tracking. Despite very basic, these features are useful to develop new control algorithms.

This tool is oriented to game developers (as an entertainment project) so the use of the SDK requires high skills in programming. This could be a disadvantage specially for new students. Looking forward for a more friendly framework, ease of use and with more graphical tools (thinking in the GUI), this work takes advantage of *QT*, a modular, cross-platform and adaptable application framework that fits very well the BPL.

Using *QT*, an user interface has been developed and integrated, which integrates the visualization of the MUAV data, GPS data (way-points visualization) and an interface for input devices (e.g. a joystick or a gamepad) that send control commands to the MUAV aside from automatic control algorithms. The importance of this interface is, as mentioned previously, that it is completely modular and adjustable.

Figure 5 shows the GCS's software architecture of *QuadLab*. The central box is the core or main process and each surrounding box represents a thread or service. This implies that each service could be enabled or disabled. The green arrows symbolize services who use TCP/UDP channels (communication with the MUAV)

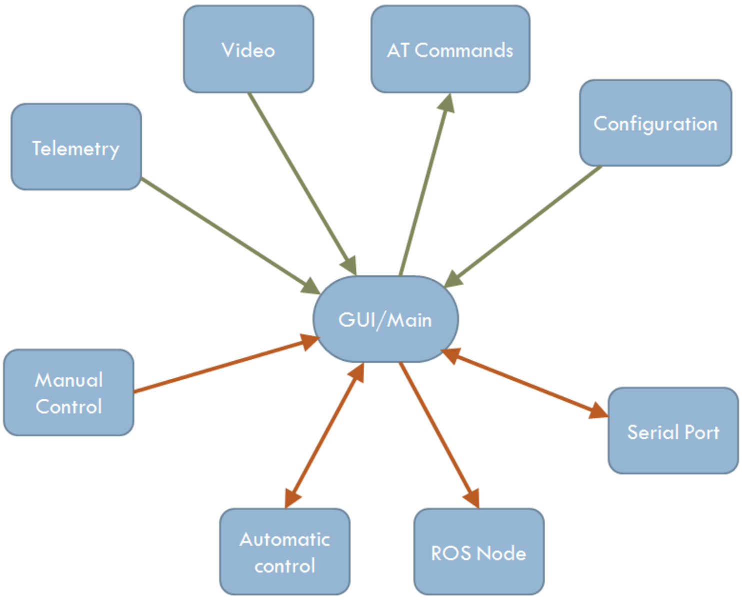


Fig. 5: GCS software architecture.

and the red arrows internal services. The main process is closely tied to the GUI and is in charge of manage all information. Then there are four modules providing all the interface with the MUAV, the "AT commands" module serves as the channel to send all information to the MUAV (i.e. configuration data and flying commands), then The "video", "telemetry" and "configuration" are only reading modules.

The Telemetry module receives datagrams from the MUAV with the status, sensors measurements (mentioned in section 4.1) and results from image analysis (this will be discussed thereafter) each 65 ms. Additionally to the navigation data, the video module receives images from both cameras, but one image from one camera at a time; in other words the user has to choose which camera is in use for capturing images. The images are compressed and transmitted in datagrams at 15fps with a resolution of 320x240 pixels, this means images from frontal camera (640x480) are scaled down and from vertical camera (176x144) are filled with null pixels. After receiving the image, this module decodes and adjusts the image to be sent to the main process for different purposes.

The serial module gives the possibility to connect devices with UART interface. It was intended to communicate the wireless devices of the add-on in the MUAV, but certainly it is totally open to connect any device since the module is configurable (bit rate, parity, stop bits etc.). In *QuadLab* case, this service is used for the NMEA protocol to have access to the GPS data, therefore this module also decodes the incoming frames and puts the info in a structure to be sent to the main process.

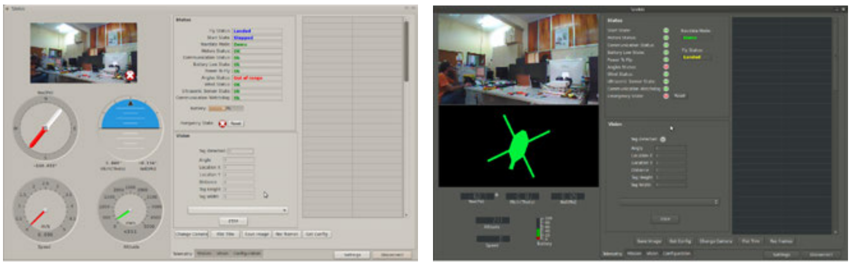
The manual control module permits to connect an input device as a gamepad for free flying or to take control in case the automatic control does not work correctly. In the automatic control service there can be implemented and configured algorithms or rules for autonomous navigation. Above the "AT commands" that serve as an interface for sending datagrams to MUAV command module was introduced, is important to clarify that this module does generate the datagrams from the control information that comes from Manual or Automatic control modules.

The "ROS node" module is a special feature that makes possible the integration with external robotic systems. It is based on ROS, an open-source modular framework that provides software libraries, drivers, tools, as well as novel algorithms that help to design complex and efficient robotic systems. The code is maintained by an extended international community and can also be re-used [16].

ROS has a message-passing philosophy, which means that each individual ROS package created is able to publish and to subscribe messages of different types, such as commands or sensor reading. In a ROS-based system it is also possible to enable communication between nodes running on different computers[7].

4.4 Graphical interface

This section describes the graphical interface, This interface was made for helping out the development of the different activities proposed (section 5). For that reason, this interface is modular, scalable and totally open. Figure 6 shows two different possibles user-interfaces, as it can be seen it is possible to add, modify or remove different types of elements. Those elements will be explained next.



(a) Interface 1

(b) Interface 2

Fig. 6: Different developed interfaces.

There are three mainly areas in the interface: video area that shows images from cameras; status area when the nautical angles are showed , battery and connection status and emergency stop; and finally the tap area that comprises:

- **Telemetry**, shows all navigation data above mentioned. In this tap it is showed when the system sets an alarm and what type of alarm occurs. It also has a special box for system identification, PID parameters for automatic flying and vision telemetry which is activated when the front camera detect one of the predefined patterns showed in figure 7, these features are used in section 4.5.2.

- **Mission control**, shows everything related to way-points navigation (figure 8). It shows the current coordinates of the MUAV, then shows information about the current waypoint (target coordinates, altitude and angle), and finally shows graphically and in a georeferenced map, the complete set of waypoints (a.k.a. mission) and the MUAV's current position. All the waypoints and configuration about the mission is introduced to the GUI by a XML file, this tab was designed for the activity 4.5.3.
- **Configuration**, has in it some tools for supporting the learning process, including reading of internal parameters of the AR.drone, sending specific PWM value to each motor, managing of serial port, exporting KML file (for view the mission in Google Earth), coordinates converter, among others.

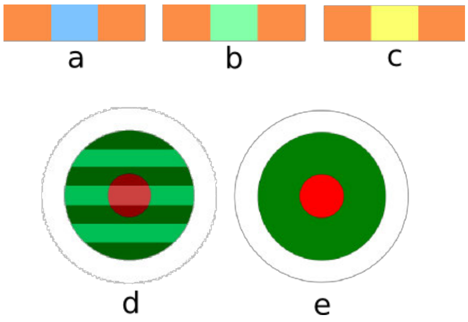


Fig. 7: Predefined patterns.

Fig. 8: Mission interface.

There is an file system associated to the application where, besides the source code of it, there are different folders in which it is saved the log files (where all the flying data is stored), captured images, image maps (for the georeferenced interface) and mission files (XML mentioned above).

4.5 Toolkits

This section shows how the designed robotic platform could be used by mean of solving small projects (for more technical information refer to [20]), which will be proposed in the next section as projects for the students to solve. Once more it is remarkable that those activities are open to modifications, as well as there is more than one way to suggest and solve each activity.

4.5.1 System Identification

For this practice, it has been used a simple AR.Drone model structure(figure 9) based in the presented model by Krajník et al. [11]. Taking into account that the AR.Drone's internal control guarantees the output angles and vertical speed, this model takes as inputs the pitch and roll reference angles as well as yaw and vertical reference speeds, and as a outputs the pitch, roll and yaw angles, the altitude and x and y axes speeds. Also it is considered that movement on each axis is independent of the others axis (e.g. x-axis movement only is affected by the pitch angle).

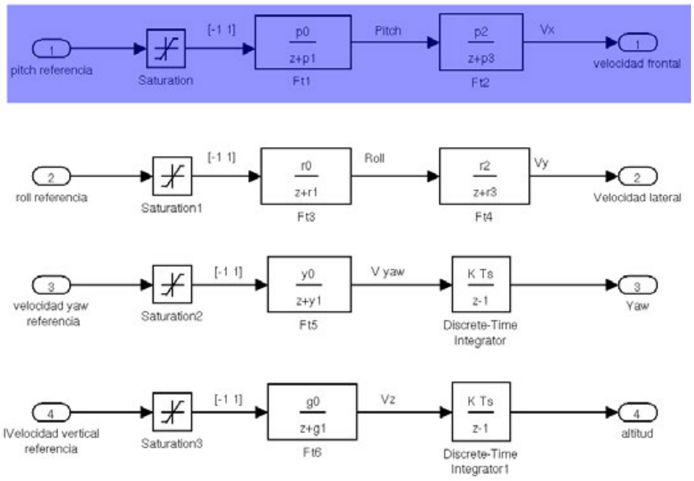


Fig. 9: Simplified MUAV Model.

This work only presents system identification of the forward-backward movement (blue shadow in figure 9) since it is the same procedure for the other movements. The first step is to give the MUAV an input sequence and then read the *log* file for the system responses. Using a time series model (e.g. ARMAX) it is possibly then to estimate a valid model for the system, figure 10 shows the step response of the real system and the estimated model.

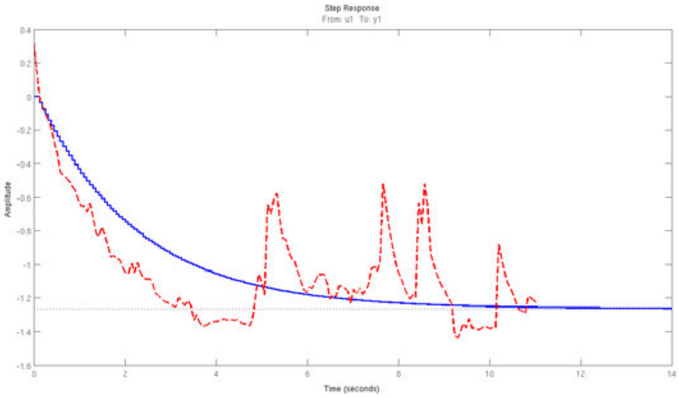


Fig. 10: Step response, real vs. model.

4.5.2 Pattern tracking

The idea of this activity is, using the frontal camera and the drone's internal pattern recognition system, the MUAV has to recognize one of the patterns showed in figure 7, and then tune in a controller to track the pattern.

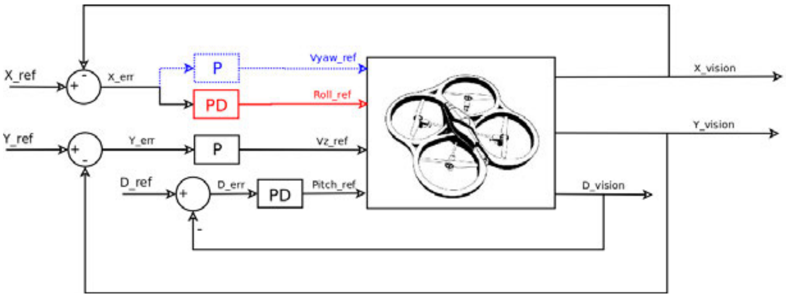
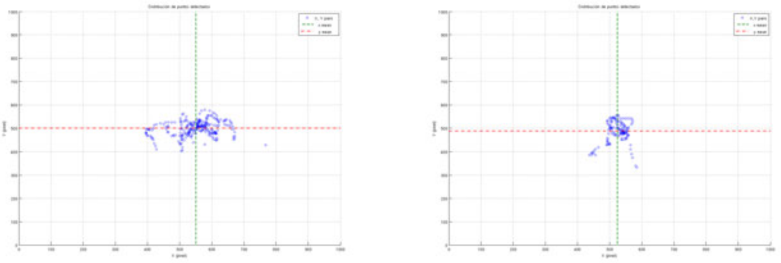


Fig. 11: Control scheme for pattern tracking.

Figure 11 shows the control scheme for pattern tracking, where the inputs are the desired position of the pattern into the image and the distance between the pattern and the MUAV, and outputs are the current pattern position and distance. In this specific case it is desirable that the controller sets the pattern centered in the image; because of the cameras resolution are different, the image is scaled to a $1000x1000pixel$ matrix, then to keep the pattern centered the reference position (x and y) must be $500pixel$ in both vertical and horizontal positions. For the distance reference it has been set in $150cm$ in order to avoid light interference and noise.

The procedure for this activity starts giving the GCS the pattern to be identified following by to check the correct recognition, then in the same tab, it could be tuned up the controllers and read the data in the log files for analysis.



(a) Roll angle control

(b) Yaw angle control

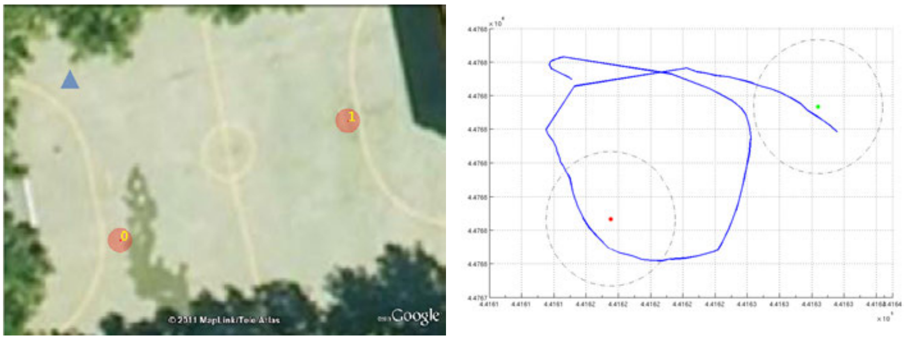
Fig. 12: Pattern position inside the image.

There are two ways to do pattern tracking, and it has to do with the horizontal control or holonomics. It can be done by modifying the roll angle (holonomic system) through a proportional controller(blue block in figure 11) or the yaw angle (nonholonomic system) through a proportional-derivative controller(red block in figure 11). Both types of control gave good results as the figure 12 shows, but the yaw angle controller (figure 12b) is more accurate and stable.

4.5.3 Waypoint navigation

This activity requires the MUAV to do an autonomous navigation visiting predefined waypoints. Due to the enhanced system, the trajectory could be traced by simple separated position controllers in x and y axes (UTM coordinates); even so, the system lacks compass, making a requirement that the MUAV heads north in order to relate the x movement with the roll angle.

Figure 13 shows a general overview of the mission, viewed in the GUI and the real trajectory. The configuration data is loaded through a XML file that contains the coordinates, altitudes, delay times and margin of errors of each waypoint. As well as general configuration and the maximum and minimum coordinates that define the mission area (useful for georeferencing) among others. In this specific activity the mission has only two waypoints with a tolerance of $4m$ in each waypoint (due to the GPS has an error of $\pm 3m$). In figure 13b it can be seen that the MUAV seems to be lost at the beginning of the trajectory, this is because the GPS measurement quality (would work better with a Kalman filter) and the mission area is only about $800m^2$. It is recommendable to keep the GPS reading data for about 10 minutes before start the mission. Even so, the MUAV gets through the waypoints and land in the second waypoint.



(a) Mission in the GUI (b) Real trajectory

Fig. 13: Waypoint navigation mission.

5 Activities and tutorials

This section proposes a series of projects/activities that can be developed using the toolkit presented in this work. Those activities have been designed taking into account both the capabilities of the platform described in section4 as well as the standards and requirements discussed on section 3.

The projects or activities proposed are highly related with the developments presented in section 4.5 so both the students and the teachers can benefit from the tools that are already available. Moreover, the projects are presented in a modular manner, and for some of them some previous developments are necessary. However, this does not implies that any of them cannot be developed independently from the rest, nor does it implies that there is a pre-defined order in which the activities can be carried out.

It should also be pointed out that, both the activities and the solutions can be taken as guidelines. They are designed to cover a very wide group of subjects and there is not a great number of details. Moreover, there may be several variations, additional requisites or limitations given to the students. This has been done according with the purposes of the toolkit which is to be flexible and with the ability of adapting to different learning objectives.

5.1 Design and construction of a data acquisition circuit, and its integration with the MUAVs communication system

The activity is oriented to design and implement a data acquisition and wireless communication system. The data is obtained from one or more sensors that will be mounted on-board the MUAV and they should be sent to a ground base station, where they can be processed on-line or stored. Both the module and the protocol used to transmit the data should be designed by the student according to the type and number or sensors, sampling frequency, and other parameters that should be defined. The design of electronic circuit must take into account among other requirements: size, weight and power supply. The operation of the circuit must not interfere with the flying capabilities or the communication system of the AR.Drone.

Project name	Design and construction of a data acquisition circuit, and its integration with the MUAVs communication system.
Objectives	Design and implement a system for data acquisition and wireless communication. The system should be designed so it can be embedded into the MUAV, and therefore it should compliant with the specifications of the aerial vehicle in terms of its weight and power limitations.
Previous Knowledge	– Physics from the first course of engineering and sciences. – Micro-controllers. – Basic programming. – Basic digital electronics.
Tools	– Electronic design tools (e.g. Eagle, KiCad). – Sensors or measurements tools to be integrated. – Micro-controllers, communication modules and other electronics components.
Detailed Activities	– Definition and selection of the sensor(s) that are going to be used. – Definition and selection of the data acquisition methodology. – Selection of the wireless communication technique (Technology, frequency, etc.). – Design of the power unit. – Design and mounting of the electronic circuit. – Tests. (Data acquisition, Data processing and Communication).

5.2 Design, Programming and Integration of a Basic Ground Control Station

An application to communicate with the MUAV, that read its telemetry and control it should be developed. It must use as a base the open-source developments available, such as the AR.Drone SDK, the different ROS drivers, and the QT libraries. An initial approach to those tools is necessary in case the student is not familiar with them, then the basic threads for communication and control should be designed and implemented, and taking that as a base more functionalities can be added to the system. This will allow the student to develop the software modularity among other concepts. It will also be the base for future activities and applications that will use the interface as an starting point.

Project name	Modelling and Identification of a Dynamic System.
Objectives	Designing and programming a simple ground control station. The program must be able to communicate with the MUAV and act as an user interface. The telemetry data, video feedback and external sensor data should be processed and displayed. It should also include the possibility of teleoperation of the MUAV using a joystick or gamepad.
Previous Knowledge	– ROS Framework. – C++ Programming. – QT Programming. – Joystick/gamepad handling over ROS.
Tools	– Framework and libraries from QT. – ROS Framework. – Parrot SDK.
Detailed Activities	– Initial Approach, study and start of the parrot's SDK Driver for the AR.Drone. – Creation of a QT project and linking of the main libraries. – Communication with the MUAV. – Integration of the telemetry readings and video feedback in the application. – Integration of input devices and sending control commands. – Integration of external sensors readings from the project described in Sec. 5.1. – Storing of telemetry and external sensor readings as well as of video screen shots. – Sending additional commands (Change camera, flat trim, reset, etc.)

5.3 Modelling and Identification of a Dynamic System

This project proposes the creation of a dynamic model for the MUAV. A method for performing the identification must be proposed, then the type of the model should be defined and its corresponding parameters must be computed. In order to do this, the student must first determine the input sequence that will be sent to the MUAV (for both identification and validation). Then the received output data can be used to estimate the parameters of the model, after that the model must be compared with the real output and according with those results determine if the proposed model is suitable for the case. The main characteristics such as stability or response time can be obtained, and a control law can also be defined.

Project name	Modelling and Identification of a Dynamic System.
Objectives	Propose a dynamic model of the MUAV and identify the type of system, its order and the corresponding parameters using different techniques. Also, both the input data and the validation approach should be defined.
Previous Knowledge	– Dynamic systems modelling. – C++ and/or MatLab programming. – Control theory basics, transfer functions, open loop response, frequency spectrum response.
Tools	– Control and Data acquisition software (e.g. GCS developed according to Sec. 5.2) – Numeric computation software (e.g. MatLab, Octave).
Detailed Activities	– Design and propose a dynamic model for the MUAV. – Select one or more parameters to identify. – Design an methodology to send the control commands to the MUAV and to store the necessary telemetry output and integrate them into the Ground Control Station. – Generate an input sequence for a given time lapse (Type of sequence, time step and duration must be determined by the student), send it to the MUAV and store the output data. Repeat the process for the validation data. – Obtain or estimate the parameters of the model proposed previously. – Compare the results of the estimated model against the real data, and obtain the main characteristics of the model.

5.4 Following an object detected by a camera

This project proposes the development of a control for the MUAV in order to follow a pre-determined object. The detection of the target is not part of this project, therefore the system for detection included in the AR.Drone drivers will be used, by doing so, the MUAV can send information regarding the detection or not of the target and the position (x,y) in the plane of the image. This can be used as input to keep the detection on the center of the image plane using a controller proposed by the student (P, PI or PID). The parameters for the controller must be estimated using the output of the identification task proposed on section 5.3. This will allow the student to analyze the sources of error and difficulties that appear when working with such complex systems, and the techniques to overcome those limitations.

Project name	Following an object detected by a camera.
Objectives	The objective is to design and implement a controller for the MUAV in order to follow an object detected by the MUAV's frontal camera.
Previous Knowledge	– Dynamic of systems and transfer functions. – Control of dynamic systems. – Response on the frequency spectrum and filtering. – Programming in C++ and MatLab.
Tools	– Control and Data acquisition software (e.g. GCS developed according to Sec. 5.2) – Numeric computation software (e.g. MatLab, Octave).
Detailed Activities	– Design a control schema and establish the reference set points in order to have the identified object in the center of the image plane. – Design and implement an additional module of the ground control station that is able to read the data from the MUAVs detection module, and send back the control commands. – Perform the tuning of the controller parameters on-line or using previously stored telemetry data. – In case it is necessary, perform a filtering process on the received telemetry data, before is sent as feedback to the controller. – Test the performance of the controller first using linear and then planar movements.

5.5 Design and implementation of a waypoint navigation control system

This project requires that the student develops a system for autonomous navigation of the MUAV using a way-point controller. The trajectory can be predetermined using a priori-known way-points or autonomously computed from a coverage area or any other similar task. Once the way-points are established they must be followed in a strict order, by sending each one to the position controller. Those functionalities are also to be embedded into the Ground Control Station, where it should be possible to input some parameters or additional information. This will require several areas of knowledge to be used therefore preparing the student for more realistic and complex developments.

Project name	Designing and implementation of a waypoint navigation control system.
Objectives	Design and implement a waypoint navigation control system, with or without trajectory controlling, integrate the controller with the available information.
Previous Knowledge	– Basic autonomous navigation concepts. – Position and/or velocity controlling. – Tuning of automatic controllers. – Classical control structures (FeedForward, ratio, cascade).
Tools	– Expanded MUAV prototype including GPS sensor and data acquisition hardware (e.g. The circuit proposed in project 5.1) – Ground control station with user graphical interface. – GIS application (e.g. *Google Earth, OpenStreetMaps*). – Numeric computation software (e.g. MatLab, Octave).
Detailed Activities	– Define the area to be covered or the way-points that must be visited. – Define a task to be performed in each way-point (e.g. Wait for a number of seconds, take an aerial image, record data from sensors). – Design and program a basic mission controller with (Start, pause, resume, cancel, etc.) and integrate it into the GCS. – Design and implement a navigation strategy in order to reach each way-point, the position controller can be of different complexity. – Integrate the controller or navigation module into the GSC.

5.6 Discussion

It is possible to estimate how the proposed projects achieve the learning objectives, and in consequence, how they satisfy the ABET criteria points established in section 3.2.

Table 3 shows the percentage of the learning objectives covered by each project. It is noteworthy that the idea is not to reach a 100% of coverage, but cover as more learning objectives as possible in a reasonable percentage; as well as to measure how well the project addresses the achievement of the objectives. These data was measured by relating each project with each theoretical contents contented in each CEA/ISA learning objective [4], and then calculating the whole percentage per objective.

		Projects				
	CEA/ISA Learning-objectives	**P1**	**P2**	**P3**	**P4**	**P5**
A	Knowledge about fundamentals of automation and control methods.	14.3	-	50	71.4	28.6
B	Knowledge and skills for modelling and simulation of systems.	-	-	100	-	40
C	Knowledge on automatic regulation and control techniques and their applications in industrial automation.	-	-	54.4	-	63.6
D	Knowledge of the principles and applications of robotic systems.	-	33.3	44.4	22.2	44.4
E	Applied knowledge of industrial informatics and communications.	50	50	25	25	62.5
F	Capability to design control and industrial automation systems.	60	-	40	20	40

Table 3: Percentage (%) covered by each activity

The first appreciation is that the learning objective *E* is covered by all the projects although some percentages are not good enough. Objectives A, D and F are well covered too. Objectives B and C are briefly covered but with good percentages.

From the project perspective, it can be seen that projects *P3* and *P5* cover most of the learning objectives, but have the problem that they require the previous projects to be achieved. For the other projects, besides they do not cover many objectives, they made emphasis in one specific objective.

Table 4 shows how well each project addresses the ABET outcomes (see table 1). This table was generated by crossing the tables 3 and 2 and is based in the course assessment matrix suggested by [6]. The projects *P1* and *P2* address not all the outcomes moderately, but the projects *P3*, *P4* and *P5* address the outcomes substantively in most of the outcomes. This means that those projects who are focused in solving a real application, have more likelihood of address more outcomes.

		EC2000 Outcomes					
	Project	**a**	**b**	**c**	**d**	**e**	**k**
P1	Design and construction of a data acquisition circuit, and its integration with the MUAVs communication system.	1	2	2	-	1	2
P2	Design, Programming and Integration of a Basic Ground Control Station.	2	-	-	2	2	2
P3	Modelling and Identification of a Dynamic System.	2	3	-	2	3	2
P4	Following of a object detected by a camera.	3	1	1	1	3	1
P5	Design and implementation of a waypoint navigation control system.	2	2	2	1	2	2

1=project addresses outcome slightly, 2=moderately, 3=substantively

Table 4: Projects assessment (ABET),

6 Conclusions

This platform is presented as an alternative to the traditional laboratories used in teaching of automatics, which generally consist of high complexity, high cost and hermetic systems, and where it is necessary to take supervision to ensure the integrity of both, the system and the user. This paper proposes an open platform, taking some advantages, such as low cost, space and time constraint, and in some cases, the need of supervision and planning. In addition to the characteristics and properties that the AR.Drone owns, it has been implemented an external circuit to improve its performance. There have been presented two prototypes which were tested outdoors in order to integrate a GPS measurement. As a result, there have been obtained stable systems, but GPS accuracy is not good enough for the MUAV workspace; a possible solution (and a future work) is the application of a Kalman filter. A development of a modular ground control station for the robotic system has been performed with the following features: connection to the robot, teleoperation, autonomous control, data acquisition and processing of telemetry and video data, interface for identification, and integration with other ROS-based robotic platforms. Additionally, it contains an interface dedicated to the control and supervision of a mission by waypoints. This paper presents a simple interface for identifying and implementing a controller from a defined model of the AR.Drone. It has also been made a tracking control of an object based on the information provided by the drone cameras. Finally, this work presents a tracking control of a waypoint mission, in which it has been used a extended-UAV prototype and a dedicated interface within the ground control station. As a result, it has been obtained a correct operation for illustrating basic concepts of control systems. Furthermore, although it is possible to perform a mission by waypoints, the platform presented many problems and restrictions on environmental conditions and accuracy of GPS. The modifications to the MUAV, as the design and development of the user interface and the ground control station, and the design and implementation of autonomous flight controllers for tracking and control for waypoint missions, have been proposed as projects for student. These activities, based on the development of this platform, are framed within a set of learning objectives and the PBL methodology. Activities or projects have been compared with the learning objectives and ABET outcomes showing that the more closer the project is to a real application, the greater likelihood of covering more outcomes.

The PBL methodology fits very well in combination with methodologies which presents to the student an activity where he or she works in a multidisciplinary team in a collaborative environment to propose and implement a solution. In this way, students not only achieve the learning objectives of an active part in the discovery of knowledge, learning and meaningful thoughts, but also reinforces the social and professional goals, which forms a fundamental part of their future in the area of environment engineering.

For future work, mainly is planned to improve the way-points tracking laboratory as an extension, also to integrate another controller devices (i.e., radio-control, kinect, wii controllers) and to improve haptics feedback adding vibration to the controller devices.

Acknowledgements This work was supported by the Robotics and Cybernetics Group at Technical University of Madrid (Spain), and funded under the projects ROTOS: Multi-robot system for outdoor infrastructures protection, sponsored by Spain Ministry of Education and Science (DPI2010-17998), and ROBOCITY 2030, sponsored by the Community of Madrid (S-0505/DPI/000235).

References

1. ABET: Criteria For Accrediting Engineering Technology Programs (2013). URL `http://www.abet.org`
2. Ahlgren, D., Verner, I.: Socially responsible engineering education through assistive robotics projects: The robowaiter competition. International Journal of Social Robotics **5**(1), 127–138 (2013)
3. Bristeau, P.J., Callou, F., Vissière, D., Petit, N.: The navigation and control technology inside the AR.Drone micro UAV. In: the 2011 IFAC World Congress, pp. 1477–1484 (2011)
4. CEA, ISA: Recomendaciones generales para impartir las competencias de automática en los títulos de grado (2012)
5. Cruz-Martín, A., Fernández-Madrigal, J., Galindo, C., González-Jiménez, J., Stockmans-Daou, C., Blanco-Claraco, J.: A {LEGO} mindstorms {NXT} approach for teaching at data acquisition, control systems engineering and real-time systems undergraduate courses. Computers & Education **59**(3), 974–988 (2012)
6. Felder, R.M., Brent, R.: Designing and teaching courses to satisfy the abet engineering criteria. Journal of Engineering Education **92**(1), 7–25 (2003)
7. Garzón, M., Valente, J., Zapata, D., Barrientos, A.: An aerial-ground robotic system for navigation and obstacle mapping in large outdoor areas. Sensors **13**(1), 1247–1267 (2013)
8. Huang, H.H., Su, J.H., Lee, C.S.: A contest-oriented project for learning intelligent mobile robots. Education, IEEE Transactions on **56**(1), 88–97 (2013)
9. Johnson, S.H., Luyben, W.L., Talhelm, D.L.: Undergraduate interdisciplinary controls laboratory. Journal of Engineering Education **84**(2), 133–136 (1995)
10. Kittenberger, T., Brodl, L., Vavra, N.: Experiences using autonomous model airplanes for embedded control education and for bachelor and master theses projects. In: Proceedings of the 1st international conference on Robotics in Education, RiE2010, pp. 177–182. FEI STU, Slovakia (2010)
11. Krajník, T., Vonásek, V., Fišer, D., Faigl, J.: AR-Drone as a Platform for Robotic Research and Education. In: D. Obdržálek, A. Gottscheber (eds.) Research and Education in Robotics - EUROBOT 2011, *Communications in Computer and Information Science*, vol. 161, pp. 172–186. Springer Berlin Heidelberg (2011). URL `http://dx.doi.org/10.1007/978-3-642-21975-7_16`. 10.1007/978-3-642-21975-7_16
12. Lee, P., Allen, R., Cole, G., Shastri, S.: A modular laboratory for process control and process engineering. Journal of Process Control **13**(4), 283–289 (2003)

13. Ma, J., Nickerson, J.V.: Hands-on, simulated, and remote laboratories: A comparative literature review. ACM Comput. Surv. **38**(3), 7:1–7:24 (2006). DOI 10.1145/1132960.1132961. URL `http://doi.acm.org/10.1145/1132960.1132961`
14. Mills, J., Treagust, D., et al.: Engineering education-is problem-based or project-based learning the answer? Australasian Journal of Engineering Education **4**(1), 16 (2003)
15. Pomalaza-Ráez, C., Groff, B.H.: Retention 101: Where robots go students follow. Journal of Engineering Education **92**(1), 85–90 (2003)
16. Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., Ng, A.: ROS: an open-source Robot Operating System. In: ICRA Workshop on Open Source Software (2009)
17. Railsback, J.: Project-based instruction: Creating excitement for learning. Northwest Regional Educational Laboratory (2002)
18. Verner, I.M., Ahlgren, D.J.: Fire-fighting robot contest: Interdisciplinary design curricula in college and high school. Journal of Engineering Education **91**(3), 355–359 (2002). DOI 10.1002/j.2168-9830.2002.tb00715.x. URL `http://dx.doi.org/10.1002/j.2168-9830.2002.tb00715.x`
19. Wankat, P.C.: Integrating the use of commercial simulators into lecture courses. Journal of Engineering Education **91**(1), 19–23 (2002)
20. Zapata, D.: Una propuesta basada en proyectos para la enseanza practica en automática y robótica. Master thesis, Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid (2012)