

Designing Questions to Teach Python with a Classroom Response System

Luis A. Álvarez-González, Gabriela González-Sáez, and Esteban Huenuman

Grupo de Investigación en Tecnología de Aprendizaje
Instituto de Informática, Universidad Austral de Chile
Valdivia, Chile

emails: {lalvarez@inf.uach.cl, gabriela.gonzalezsaez@gmail.com, esteban.huenuman@hotmail.cl}

Abstract— To do an effective class using a Classroom Response System, two important questions appear: What questions to ask? and How to ask them? This paper tries to contribute with answers for these questions for an introductory course of Python language. For the first question, the Question Driven Instruction (QDI) methodology is introduced. For the second one, the questions are divided in content questions and process questions according to the content objectives and process objectives. Some examples questions are shown. The Classroom Response System (CRS) called “Mobile QTI” is used. “Mobile QTI” was designed for mobile devices (tablets and smartphones), supports eight types of questions, and is under IMS Question & Test Interoperability Specification (QTI). Finally, an experiment is presented in order to validate the methodology. The experiment shows a relationship between Grade Point Average (GPA) and the use of the QDI and the CRS Mobile QTI.

Keywords—*Mobile-learning; Classroom Response Systems; Question Driven Instruction; Python Language.*

I. INTRODUCTION

As a way to improve learning, the research community in educational science has promoted the concept of Active Learning [5][11]. Learning-centered evaluation indicates that it must be continuous, and feedback essential to guide students in the learning process. In order to give response to learning-centered evaluation, the formative assessments contribute to know the learning of the students during the teaching process [3][6][10]. This is in contrast with the summative assessment, which is usually made at the end of a learning unit, and tries to measure how much the student learned. The main purpose of formative assessment is learning and the main purpose of summative assessment is the qualification. Formative assessment is probably one of the best "innovations" in education to improve learning. However, for that to happen there must be feedback as soon as possible. Using Classroom Response Systems (CRS) helps transform classroom dynamics, allowing learning focused assessment. In other words, a CRS, can be used to increase the frequency of questions and reduce response times in the formative assessment, so to have improved and faster feedback. The CRS can help the student-centered learning and active learning in general, but it depends on how the teacher uses it. The CRS is a tool and not “the solution”. A CRS can be used in many different ways and

many of them questionable, such as taking attendance in classes [8][12].

However, the most of the CRS only support multiple choice questions, true/false question and short texts [8] [17]. To support more types of questions it was necessary to develop a new CRS called Mobile QTI [1][18]. Mobile because use mobile devices, and QTI because is under IMS Question & Test Interoperability Specification [14] (IMS QTI Specifications).

The Agile Teaching/Learning Methodology (ATLM) is a systemic approach to teaching/learning that has taken place in several courses of Computer Science at the City University of Hong Kong [9]. As the name implies, ATLM is a methodology for both teaching and learning. Teachers have a well-defined way of educating, in the same way that students have a very clear path to learning. ATLM is a methodology that tries to balance a good teaching with good learning. This approach has been obtained from the software engineering field, where the development process can become complex and dynamic, which often require changes in order to minimize risks. The teaching process in many ways is similar to the software development process, which involves different objectives (sometimes conflicting), completion dates, limited resources and high expectations. Like software engineering, teaching requires detailed planning, monitoring and management of assessments and feedback from participants.

According to Bransford et al. [7] effective learning environments are:

1. Student-centered. Organises learning from prior knowledge of the student.
2. Knowledge-centered. Treat knowledge as an interconnected structure, that can be organised and improved in order to expanded.
3. Assessment-centered. Considers assessment as a continuous process, where education should be commented on while the teaching/learning process takes place. This is contrasted with summative assessment conducted after instruction. So, the first objective is formative assessment [3][4][6].
4. Community-centered. It recognizes that students belong to a community of students in a course of a degree, an institution, a society and promotes collaborative learning

Section II explains why Python as first programming language is recommended. In Section III, a brief explanation of the main functions of Mobile QTI as CRS is given. In Section IV, the relationship between the methodology Question Driven Instruction, formative assessment and the use of CRS is explained. In Section V, some questions used in an experiment are shown. In Section VI, a brief explanation of the experiment and the results are described. Finally, in Section VII, the conclusion and future work are presented.

II. PYTHON AS THE FIRST COMPUTER LANGUAGE

The training of engineers needs to develop logical thinking, which subsequently helps them to plan solutions, algorithms or procedures and also to create and solve problems in general. The development of logical thinking is achieved by learning a computer language, among other subjects. There are several computer languages, but few are recommended as the first language. In several universities, C++ or Java is used as a first language; however, Python is easier to use [16]. For example, the typical first program that every student takes is for screen display "Hello World!"

In C++ corresponds to:

```
#include <iostream.h>
int main()
{
    cout << "Hello World!";
}
```

In Java to:

```
public class HelloWorld
{
    public static void main(String[]
args)
    {
        System.out.println("Hello
World!");
    }
}
```

While Python is simply

```
print("Hello World!")
```

In other words, while in C++ and Java, several statements and explanations are needed, Python uses only one sentence, easy to understand. Additionally, Python is interpreted and available in mobile devices, making it easy to use.

Figure 1 shows an exercise for learning Python lists using a mobile phone. A list of integers is filled and then several lines with asterisk ("*") according to the integer in the list is displayed. The number of asterisks on each line corresponds to the integer value of each item.

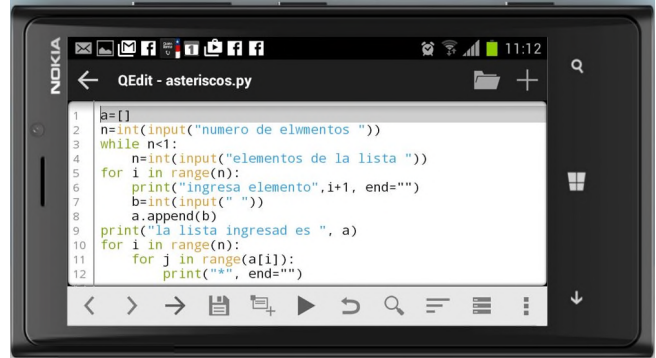


Figure 1. Python program in a mobile phone with Android operating system.

Its availability on mobile devices allows students to practice Python on their own devices, anytime, anywhere.

III. MOBILE QTI

The Mobile QTI is a web player for question under IMS-QTI Specification that can be used as a CRS. The question can be built with either the Web QTI or AquRate, both editors are under IMS QTI Specifications. The Web QTI is a web editor [19] and AquRate [20] need to be downloaded to use it. To develop Mobile QTI, Javascript, PHP and HTML5 were used. The users (teacher and students) need to be registered because they have different functions. After the teacher has launched the questions, the students answer and, in real-time, the answers are relayed to the teacher for inspection [18]. Figure 2 shows the main menu in a mobile device.



Figure 2. The Mobile QTI, a) Teacher's view, b) Student's view

The main menu for teachers shows the following options:

- **Download Interaction** (*Cargar Interacción*): The teachers download questions from his/her laptop or repository.

- **Create a Test (Crear Test):** The teacher creates a test with the questions previously stored in Mobile QTI. The teachers can then select who participates in the test and who doesn't. In others words, a teacher can do several tests to different group of students.
- **Download Test (Cargar Test):** The teacher launches the test.
- **Results (Resultados):** Shows the results of each test per students and class.
- **List of Users (Lista Usuarios):** Shows all the users of Mobile QTI.
- **Administration (Administración):** For the management of questions, test and users.

Figure 3 shows a summary of the students' answers on a class with four questions and ten students. The tick is for a correct answer, the cross for an incorrect answer and dash for an incomplete question. In case the teacher needs to do a detailed analysis, the summary can be exported to an Excel File. Also, the answers of each student can be seen.

	Calciformes	Cinco Sentidos	Partes Ojo	Olfato	
ESTEBAN H	✓	-	-	✓	2 de 4
LUIS A	-	-	✓	-	1 de 4
CATALINA F	✗	✗	-	-	0 de 4
DAVID I	✓	-	✓	✗	2 de 4
FRANCO F	-	✓	-	✓	2 de 4
KAREN A	✓	-	✓	-	2 de 4
VALERY S	-	-	-	-	0 de 4
CARLOS A	-	✓	-	✓	2 de 4
DIEGO N	-	-	-	-	0 de 4
CAMILA A	✓	-	✗	✗	1 de 4
PROFESOR	✓	✓	✓	✓	4 de 4
	5 / 11	3 / 11	4 / 11	4 / 11	

Figure 3. Summary of the students answers.

The teacher needs to download one or several questions and then launch them to the students. For example, a teacher can launch question by question to a class or launch several questions at the same time (test). The teacher can see this live table for each question to know the progress of each student; very useful when you are lunching question by question to a class to know the students understanding. This way, the teacher can see the progress of the students at any time.

Eight types of question are supported by Mobile QTI:

1. **Choice:** The student selects one of several alternatives.
2. **Order:** The student order concepts according to a criteria.
3. **Associate:** The students associated concepts.
4. **Inline choice:** The students need to fill blanks with a word.
5. **Text entry:** The students need to fill blanks with a sentence.
6. **Hotspot:** The students select one of several images.
7. **Graphic order:** The student sort several marks in an image.
8. **Slider:** The students need to move slider between two points bound to the answer.

The question type order is very useful in programming language. Typically in a question, several sentences are displayed and the student needs to order them correctly.

Most of the CRSs support only multiple choice, true/false, and text entry, and do not work under any specification. For this reason, Mobile QTI was developed [1].

IV. QUESTION DRIVEN INSTRUCTION

Active learning and the strengthening the four "centered learning" is possible with the help of Classroom Response System (CRS); however, this is only a technology and its effectiveness depends on the teacher and the methodology to be carried out.

The CRS, as any technology, can be used in many different ways, with varying degrees of impact on learning, for example with diagnostic tests, summative assessment class discussions, etc. However, the Question Driven Instruction (QDI) [2][13], tries to be an alternative to traditional "transmit and evaluate". In QDI the learning activity starts with a question and the periods between feedback should be very short, so that the teacher can monitor student's progress, identify difficulties in learning of each student, and take decisions in order to make an effective class. This methodology is under the notion of agile teaching/learning methodology [9] and personalized learning [15].

Formative assessment is the focus of the QDI methodology and provides quick feedback for the student to "drive" their learning activities and teacher to "drive" their teaching strategies.

A teacher using QDI requires experience and a good knowledge of the subjects that they teach, especially in environments with students of differing levels and speeds of learning. In other words, the teacher needs to be continuously adapting. Many teachers, unconsciously, develop this methodology for smaller classes, but for larger classes is very difficult without the use of technology.

A typical class using QDI begins with a question (a traditional class begins with an exposition). It takes a few minutes for students to respond. Students respond using CRS and the teacher shows a graph of the students' answers. Without disclosing the correct answer, the teacher asks for a volunteer looking to respond with their answer (whether correct or incorrect). The next step is to discuss each result without saying whether it is correct or incorrect. This way the teacher can analyse several related matters. Therefore, it is very important to design good questions in order to cover the learning objective. Also, it is possible to do more questions to explore the matter more profoundly. The teacher can finish with a summary of ideas from the different answers or present a slide to clarify the concepts discussed in the class. Figure 4 shows the different stages of the methodology.

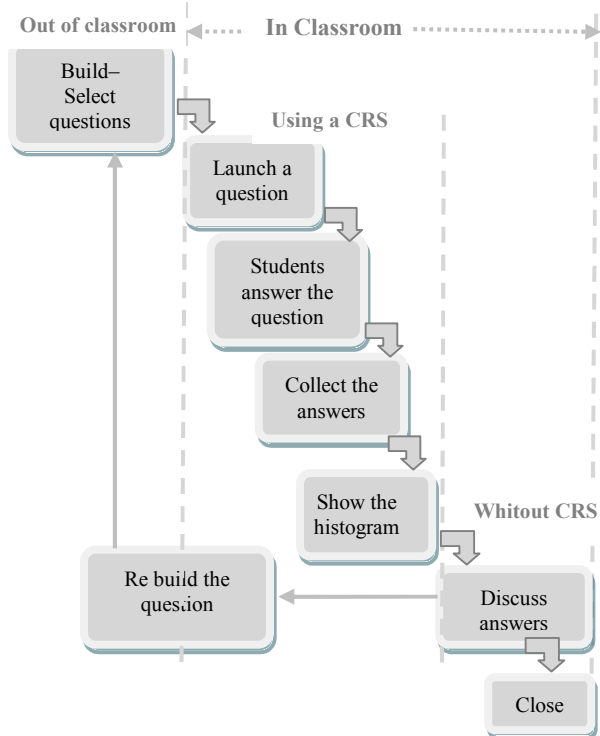


Figure 4. Cycle of the Question Driven Instruction methodology.

The cycle starts building and selecting questions according the learning objective, these questions are “launched” to students, the students answer by group or individually, according to the lesson plan, the answers are collected and a histogram with the answers of each student

is displayed (the right answer is not shown) and the discussion starts. To have a profound discussion the questions must be very content rich. To start the discussion, the teacher asks for a volunteer (ideally with a wrong answer). The volunteer student then argues her/his answer, and then the rest of students discuss it, until the right answer is discovered. The cycle ends with a summary where all the possible answers are analysed.

The time must be sufficient so that most of the students can participate, and no so much as to lose the learning objective. Flexibility is very important. Some questions may be easily understood by some students whilst others may struggle. The teacher needs to adjust the teaching method according to time constraints and the learning objectives. This is the essence of the Agile Teaching. However, all learning material must be available for every student.

This methodology needs to be analysed according to the context, i.e., type of objectives and learning characteristics, for example if the objective is an individual skill, probably another methodology could be more appropriate.

A. Type of questions

Before building or selecting questions, the teacher needs to have declared the learning objective. In general the objectives can be classified into content objectives, process objectives, and metacognitive objectives.

Content objectives: with questions related to content, usually a brief discussion is enough. In programming languages, these types of question can be related to the syntax of a language. In this case, the cycle of QDI can be brief. However, they are very important in order to understand more advanced concepts.

Process objectives: the questions of contents are oriented towards basic knowledge; the process question is oriented about the relationship of this basic knowledge. How are they related? i.e., the answers require a small analysis. The cycle QDI could need more time.

Metacognitive Objectives: the metacognition, generally, relates to building knowledge based on the previous knowledge, e.g., solving news problems. These questions are usually reserved for summative assessments.

V. QUESTION FOR FORMATIVE ASSESSMENT IN PYTHON

To teach introduction to Python language, the contents are structured in four units of learning: a) simple statements, b) conditional statements, c) loop statements, and d) list. For each of these units of learning, the knowledge of syntaxes and semantics (content objectives), and how to use this statements (process objectives) are necessary. So, eight sets of questions were made: two learning objectives (contents and processes) by four units of learning. Each question was discussed with the teaching team, in order to have rich question.

A. Simple statements

The simple statements are related with the use of variables, data type, operators, and built-in function. Here, it is important to know the syntaxes, the semantic, and introduce the simple sequences, to solve small problems.

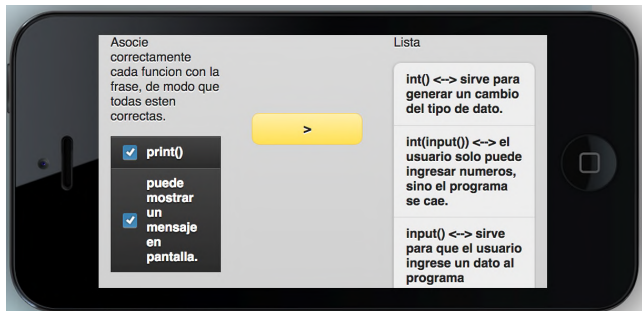


Figure 5a. Question of content for Python language. Associative question.

Figure 5a shows a content question and Figure 5b shows a process question for simple statements. Figure 5a, shows an associative question, where the students relate some function of Python (`int()`, `input()`, and `print()`) with their use.

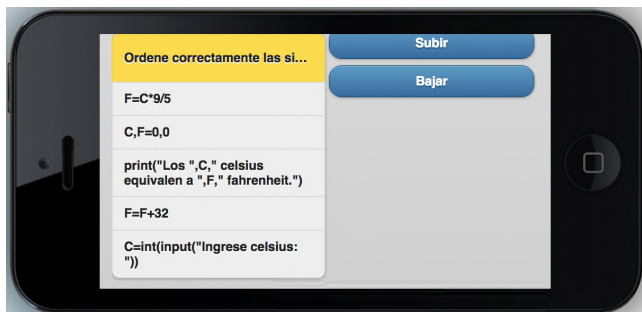


Figure 5b. Question of process. Order question.

After knowing the concepts of variables, data type, and basics built-in functions, the next step is how to build a program in Python. Figure 5b, shows an example, where the student needs to order, several sentences that convert degree Celsius to degrees Fahrenheit.

B. Conditional statements

The conditional statements in Python are `if`, `else` and `elif`, and also these statements can be nested. Figure 6a shows a content question to identify the right use of these statements.

Figure 6a shows typical errors when the `if` statements is used, and Figure 6b shows a program in Python to calculate if a quadratic equation has real roots or complex roots.

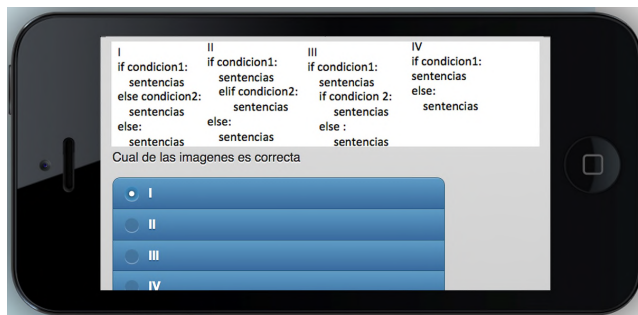


Figure 6a. Content question for statement `if`. Multiple choice question.

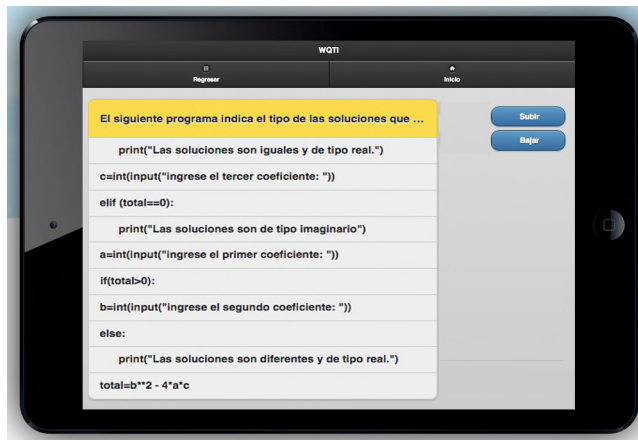


Figure 6b. Process question for `if` statement. Order question.

Here, the important point is that the student can recognise the right sequence, according to the data input and calculus.

C. Loop statements

The `while` and the `for` statements are studied. The first one needs a condition to repeat the loop, the second one needs an index in a range. Figure 7a shows an example of content question and Figure 7b an example of process question. The content questions are oriented to the syntaxes of the statements and the process questions oriented to the execution of the statements.

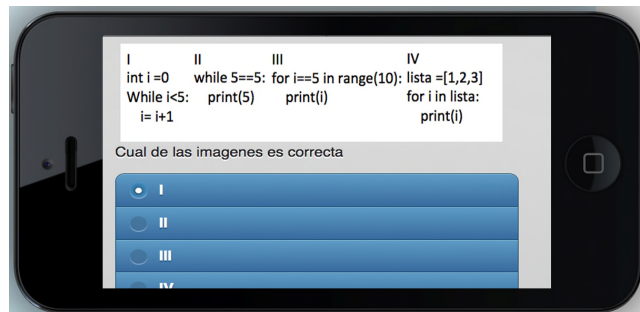


Figure 7a. Choice content question for `while` and loop statements.

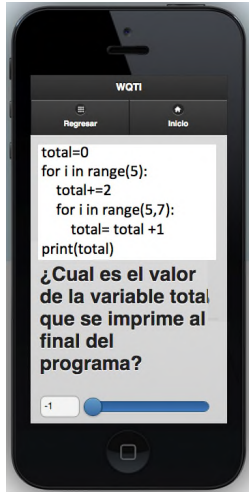


Figure 7b. Slider process question to `for` loop statement.

D. Data Types: List

The lists are necessary as an introduction to structured data in Python. To start with lists, the students need to know some methods as `append()` and some functions as `len()`, and identify the list elements. The content questions are oriented to learn this basic terminology. After this, is necessary to understand how to use a list, to do that, process questions are needed. Figure 8a shows a question oriented to learn the basic methods, functions and identify the elements.



Figure 8a. Content question for lists. Associative question.



Figure 8b. Process question for list. Order question.

Figure 8b, shows the sentences of a program that the students need to order, to obtain the biggest number and the position in the list.

VI. AN EXPERIMENT

The Mobile QTI and the QDI methodology were used with an experimental group to teach Python language in the Universidad Austral de Chile. In Chile, the Grade Point Average (GPA) ranges are from 1.0 up to 7.0 (with one decimal place). Table 1 shows the Chilean GPA.

TABLE I. CHILEAN GPA

GPA	% Achievement	Meaning	Honours
6.0 - 7.0	83% - 100%	Outstanding (7.0)	Highest Honours
5.0 - 5.9	66% - 82%	Good	Honours
4.0 - 4.9	50% - 65%	Sufficient	Passed
3.0 - 3.9	33% - 49%	Less than Sufficient	Failed
2.0 - 2.9	16% - 32%	Deficient	Failed
1.0 - 1.9	0% - 15%	Very Deficient	Failed

https://en.wikipedia.org/wiki/Academic_grading_in_Chile

The contents are organised in three unit of learning, so, three summative assessments were considered (S1, S2, and S3). In the summative assessments, only problems to solve are presented (metacognitive objective). In S1 problems, simple and conditional statements are evaluated, in S2 problems loop statements are used. And S3 problems where the data type: list; including the loop, conditional and simple statements; are needed. Students write their solutions in pseudocode, Nassi-Shneiderman diagram (NSD) [21], and Python language. The assessments are at the same time in several classrooms. The answers made in paper and the lecturers review one by one.

The Mobile QTI and QDI were used only in the second unit of learning. So, the impact is possible to see in the second summative evaluation (S2). Table 2 shows the results in experimental group.

TABLE II. RESULTS OF THE EXPERIMENTAL GROUP

Number of students: 20	S1	S2	S3
GPA	3.5	4.0	3.6
Variance	2.0	2.1	6.1
Increase of GPA respect of S1		14.3%	2.9%
Number of students with grades over 4.0	8 (40%)	13 (65%)	10 (50%)
Increase number of students with grades over 4.0 respect to S1		62.5%	25%

The programming language is not easy to learn, especially for the beginners. For example, in the summative assessment (S1), only eight students had grades greater than or equal to 4.0. For the second unit of learning (S2); with Mobile QTI and QDI; the GPA increased from 3.5 to 4.0 (14.3%), and the students with grades greater than or equal to 4.0 increase from 8 to 13 (62.5%). However, the GPA for the last unit of learning (S3), decrease from 4.0 to 3.6

and the students with grades greater than or equal to 4.0, also decrease from 13 to 10.

VII. CONCLUSION AND FUTURE WORK

The programming language is difficult to learn for engineering students, with a GPA of just “Sufficient” in S2, and “Failed” for S1 and S3.

To define questions for content objectives and process objectives, a procedure is needed in order to cover most of the related material. Motivation also needs to be included.

Using CRS and QDI in the second unit of learning allows an increase of the GPA of S2, but, more importantly, it also increases the number of students with grades greater than or equal to 4.0 (from 45% to 65%).

The variance in the last summative assessment is three times the variance in S1 and S2 (6.1), because several students left the course, and it is important to consider this variable.

Designing effective questions that cover more contents and more learning processes is a big challenge. In fact, it is always possible to improve the questions and their answers. Designing effective questions is a dynamic process, because every course is different from another.

The complete course has 156 students divided in several groups, according to grade, where the professors and programming examples are different. So, it is not possible to consider the rest of the course as control group. In the previous semesters, the course was different, thus, it is also not possible to compare with the previous courses.

An increase in the number of students in the experimental group is necessary to have better conclusions.

A redesign of the experiment with one or more control groups is necessary to obtain more conclusive evidence.

A qualitative evaluation is also necessary, that can consider the relationship between the methodology, mobile devices, and the active engagement of the student in learning. Also, the qualitative evaluation must consider to find out how the students are thinking and solving problems.

VIII. ACKNOWLEDGMENTS

The authors wish to thank the Research and Development Office of the Universidad Austral de Chile through project DID S 2015-20 entitled SEI: Interactive Evaluation System under the IMS QTI Specification. Additional gratitude to the valuable comments from all members of the Research Group on Learning Technologies (www.gita.cl)

IX. REFERENCES

- [1] L. Alvarez González, A. Campos, M. E. De la Maza, and D. Ojeda, “Interactive assessment learning environment system under IMS-QTI specification,” *Education Technologies and Computers (ICETC)*, IEEE Press, Sept. 2014, pp. 7-11.
- [2] I. D. Beatty, W. J. Leonard, W. J. Gerace, and R. J. Dufresne, “Question driven instruction: Teaching science”, in *Audience response systems in higher education: Applications and cases*, 2006, p. 96.
- [3] B. Bell and B. Cowie, “The Characteristics of Formative Assessment in Science Education”, in *Science Education*, vol. 85, no. 5, 2001, pp. 536-553.
- [4] P. Black and D. Wiliam, (1988). “Assessment and classroom learning”, in *Assessment in Education: Principles, Policy & Practice*, vol. 5, no. 1, 1998, pp. 7-71.
- [5] C. C. Bonwell and J. A. Eison, *Active learning: Creating excitement in the classroom*. ASHE-ERIC Higher Education Reports. ERIC Clearinghouse on Higher Education, The George Washington University, One Dupont Circle, Suite 630, Washington, 1991, DC 20036-1183.
- [6] C. Boston, “The concept of formative assessment ERIC Clearinghouse on Assessment and Evaluation College Park MD”, ED470206, 2002.
- [7] J. D. Bransford, A. L. Brown, and R. R. Cocking, *How People Learn: Brain, Mind, Experience, and School*, Eds. Washington, D.C.: National Academy Press, 1999.
- [8] Byrne Richard, “Seven Good Student Response Systems That Work On All Devices”. Available from <http://www.freetech4teachers.com/2014/03/seven-goodstudent-response-systems.html#VWR8ikZlaVA>. [retrieved: April, 2016].
- [9] A. H. W. Chun, “The agile teaching/learning methodology and its e-learning platform”, In *Advances in Web-Based Learning-ICWL, 2004* [Springer Berlin Heidelberg. pp. 11-18,2004]
- [10] E. H. Hobson, “Formative Assessment: An Annotated Bibliography”, in *Clearing House*, vol. 71, no. 2, 1997, pp. 123-125.
- [11] P. W. Laws, “Millikan Lecture 1996: Promoting active learning based on physics education research in introductory physics courses,” *American Journal of Physics*, vol. 65, no. 1, 1997, pp. 14-21.
- [12] C. H. Fies, “Classroom Response Systems: What Do They Add to an Active Learning Environment?,” unpublished doctoral dissertation, University of Texas at Austin, Austin, TX. Available from: <http://repositories.lib.utexas.edu/handle/2152/1859>. [retrieved: April, 2016].
- [13] G. González Sáez and L. A. Álvarez González, “Designing Question to Teach Python in a Interactive Assessment System” *Conferencias Chilenas en Tecnologías de Aprendizaje*. Universidad de Tarapacá, Ag. 2015. Available from: eudev.uta.cl/cclt2015/pdf/python.pdf. [retrieved: April, 2016].
- [14] I. QTI, “IMS Question & Test Interoperability Specification” IMS Global Learning Consortium, 2005.
- [15] E. Alepis and M. Virvou, *Object-oriented user interfaces for personalized mobile learning*, Springer, 2014.
- [16] J. M. Zelle, “Python as a first language”, in *Proceedings of 13th Annual Midwest Computer Conference*, vol. 2, March 1999, p. 145.
- [17] D. A. Carrera Escobar and L. A. Álvarez González, “Free Classroom Response Systems to use with Mobile Devices”. *Conferencias Chilenas en Tecnologías de Aprendizaje*, Universidad de Tarapacá, Arica, Ag. 2015. Available from: http://eudev.uta.cl/cclt2015/pdf/Carrera_Escobar.pdf. [retrieved: April, 2016].
- [18] E. A. Huenuman Villarroel, L. A. Álvarez González, and R. Krause, “Mobile-QTI: An Interactive Mobile System under IMS-QTI Specification”. *Conferencias Chilenas en Tecnologías de Aprendizaje*, Universidad de Tarapacá, Arica, Ag. 2015. Available from: <http://eudev.uta.cl/cclt2015/pdf/mobileQTI.pdf>. [retrieved: April, 2016].
- [19] Web QTI, “Web editor under IMS QTI Specification”, Available from: <http://www.gita.cl/proyectos/web-qli/>. [retrieved: April, 2016].
- [20] G. Alsop, J. Annesley, Z. Cai, A. Campos, M. Colbert, and J. Orwell, “AQuRate Final Report”. JISC Capital Programme. Univ. of Kingston, London, AQuRate, Tech. Rep. 2008.
- [21] I. Nassi, and B Shneiderman, “Flowchart techniques for structured programming”. *ACM Sigplan Notices*, 8(8), 12-26. 1973.