

PSICO-A: A Computational System for Learning Psychology

Javier González Marqués

Department of Basic Psychology II, Complutense University, Madrid, Spain
Email: javgonza@psi.ucm.es

Carlos Pelta

Department of Basic Psychology II, Complutense University, Madrid, Spain
Email: pelta.carlos@gmail.com

Abstract—PSICO-A is a new educational system, based on the web, for learning psychology. Its computational architecture consists of a front-end and a back-end. The first one contains a design mode, a reflective mode, a game mode and a simulation mode. These modes are connected to the back-end, which is composed of a rule engine, an evaluation module, a communication module, an expert module, a student module and a metacognitive module. The back-end is the heart of the system analysing the performance of pupils. PSICO-A assembles Boolean equations introducing algorithms such as those of Levenshtein, Hamming, Porter and Oliver. The system design used the programming language PHP5 for a clear and fast interface. PSICO-A is an innovative system because it is the first system in psychology designed for assessing the value of computer-based learning games compared with simulations for teaching the subject. Other systems use virtual environments for teaching subjects like mathematics, physics or ecology to children but the role of digital games and simulations in learning psychology is to date an unexplored field. A preliminary analysis of the motivational value of the system has been performed with sample of undergraduate students, verifying its advantages in terms of to encouraging scientific exploration. An internal evaluation of the system, using the game mode, has been conducted.

Index Terms—PSICO-A, Psychology, Education, Intelligent Tutoring Systems, Digital Games, Simulations.

I. INTRODUCTION

The term ‘Intelligent tutor system’ (ITS) was coined by Sleeman and Brown [1] and designates a software system that uses techniques of artificial intelligence to represent knowledge and interacts with students so they learn the concepts. ITS has evolved from a mere Skinnerian instructional proposal [2-5] towards the design of virtual environments of experimentation according to a constructivist viewpoint of learning [6]. Very recent and relevant examples are MetaTutor [7], Betty’s Brain [8] or REAL [9]. MetaTutor is a learning environment designed to foster students’ learning about the human circulatory system. MetaTutor is designed to

train self-regulated processes [10, 11] that relate to metacognitive monitoring and learning strategies for handling task difficulties. Using student trace data and think-aloud protocols, the authors provide insight into the student’s thought processes. Students predominantly use strategies for acquiring knowledge from the system, and they only occasionally employ monitoring strategies to verify what they have learned [12].

Betty’s Brain is based on the learning by teaching paradigm [13-15]. This computational system uses a combination of speech, text and animation, implementing several types of activities. Betty is a virtual agent and students teach Betty by constructing a concept map representation [16]. Students check their teaching by asking Betty questions, which she answers using causal reasoning through chains of links [17]. Students can observe how well they have taught Betty through a set of questions chosen by the mentor agent, Mr Davis. Mr Davis not only provides feedback to Betty and the student but also provides advice, when asked. Two types of self-regulation strategies are posited by Betty’s Brain [18]: (a) information seeking, whereby students search the available resources in order to expand existing knowledge, and (b) information structuring, whereby students structure the information through causal relationships to build their concept maps. It has been demonstrated that the Betty’s Brain system facilitates students’ science learning by promoting very productive cognitive and metacognitive processes [19]. Students who utilised this system constructed better concept maps than students who used a non-teaching version or traditional methods [20].

The REAL or Reflective Agent Learning environment [21, 22] is an interactive learning environment that allows students to construct imaginary worlds, generating simulation games and reflecting upon their demonstrated understanding of a subject. REAL contains a reflective agent or student model which models students’ level of competence. The reflective agent can be released in the simulation environment running in the students’ imaginary world [23]. Thus, the reflective agent can be viewed as the pupil’s own embodiment within the simulation. An expert agent consists of propositional networks and procedural rules in the form of *if-then* clauses and the pedagogical agent

decides which students' misconceptions to look for and what feedback to give. The feedback is sent to the user in the form of thought boxes. It has been well established that REAL is a motivational learning environment and encourages reflection by giving users several cognitive tools to explore problematic situations.

PSICO-A includes features proceeding from the implementation of those computational systems. For example, PSICO-A is a system based on agents, uses the idea of Novak's conceptual maps [24], is inspired by representational theory [25] and gives relevance to metacognition [26]. Other pedagogical influence derives from Karpicke and Blunt's research on free retrieval practice for learning [27]. PSICO-A is an innovative system, however, because it is the first one teaching psychology to undergraduates. Other systems use virtual environments for teaching children anatomy, ecology or mathematics, for instance, so why not design intelligent learning environments for teaching psychology? PSICO-A is also the first system to assess the value of computer-based learning games compared with educational simulations, today a key issue in the field of the application of computational systems to education.

PSICO-A is a web-based system since the future design of intelligent tutoring systems is on the Internet. The advantages of this approach are obvious: first, a bulk storage; second, the flexibility to reconfigure the design. In fact, PHP5 is the language used to achieve a clean and fast interface in PSICO-A. The back-end of the system contains an Analyser that uses algorithms of similarity and distance for finding the presence of concepts. In this paper, we present the computational architecture of PSICO-A to provide the generic framework of the system. We then describe its implementation and next, we expose the experimental design. We conducted two experiments with several samples of high school students. The first one revealed an external evaluation and the second one revealed the real functionality of PSICO-A by realising an internal evaluation. Finally, we propose some conclusions and future directions for this line of research.

II. ARCHITECTURE OF THE SYSTEM

PSICO-A, developed in PHP5 language [28], consists of a front-end, which is the area in which the student interacts with the system, and a back-end or teacher interface (where the teacher performs the Analyser settings and gets performance data from students). In the front-end we find a home screen subdivided into the following components: on the middle left is a 'Concept Map Editor' which displays a 'File Menu', a 'Mode Menu' and a 'Help Menu'. The Concept Menu follows the theoretical principles established by Novak [24] and consists of a series of boxes occupied by concepts and connectable by the student through the use of various arrows. The mode is displayed in 'Game', 'Simulation' and 'Reflection'. The game consists of an animation in which MOUSI (our virtual agent) travels through a maze

(following the experimental design by Crespi [29], measuring the effect of the amount of reward on the speed with which a rat would cover a certain course; this experiment was very relevant for the development of drive-reduction theory by Hull [30], which is the subject of our Didactic Unit). The simulation presents MOUSI running through a corridor (as in the original experiment by Crespi) with different incentives and velocities. Whereas simulations 'are structured environments, abstracted from some specific real-life activity, with stated levels and goals' [31], computer-based learning games are 'applications using the characteristics of video and computer games to create engaging and immersive learning experiences for delivering specified learning goals, outcomes and experiences' [32]. Computer games are interactive experiences that are fun to engage in while building awareness and educational simulations usually develop skills and capabilities more rigorously. Reflection Mode allows the students to verify the accuracy of their learning. When they click on the 'Learning' button (center top), the Didactic Unit window disappears and 'Prior Knowledge', 'Notepad', 'Confidence Judgments', 'Metacognitive Judgments' and 'Evaluation' buttons come up.

The back-end or teacher interface is at the heart of how our computational system works. Its core is a student performance Analyser that allows the interpretation and verification of their texts in natural language. The Analyser allows them to enter word lists. In the Editor of the lists and in the concepts it is essential to introduce a code identifying the elements introduced. As regards lists, not only single words can be entered but whole sentences, always separated by commas. Such lists, being saved, are compared by the system with the sequences of terms introduced by the student in the Notepad. The lists are then used in equations of concepts. Concepts must be identified by a different code in each case, their name and a description of themselves. Each concept represents either a correct concept (which is expected to be present) or the wrong one (which is expected to be absent). The system determines if a concept is present or absent in the student's response, executing an equation defined by the teacher for that concept. The equation consists of a series of functions (and parameters), concatenated with the logical operators $\&$ =AND, $|$ =OR and $!$ =NOT. Each function returns a logical value (1=TRUE, 0=FALSE) and brackets and parentheses can be used to group them, as is done conventionally in mathematical or logical expressions. Therefore, for 'E', expressing equation, 'F', function, and 'C' concept, $E(C)=(F1|F2)\&F3$ is analysed as meaning that if C is considered to be present if F3 and at least one of the functions F1 and F2 give TRUE. Each function is applied to each of the words of the response.

The functions can receive parameters. If the function applied to a word or group of words gives the value TRUE, the function is valid for that word or group of words and is marked on the Analyser. If, on the contrary, the function is found to be invalid, the concept does not appear. Clearly, articles and pronouns are discarded and

we are only interested in nouns, adjectives, verbs and adverbs. Nor is there sensitivity to capital letters. Three of the functions also have as argument a number (n) and make use of distance algorithms or similarity measure between chains of signs: Similarity is the function ($\text{similar}(n, \text{list})$), Levenshtein is the function ($\text{levenshtein}(n, \text{list})$) and Hamming ($\text{hamming}(n, \text{list})$). Two functions, Accuracy (list) and Quasi-Accuracy (list) use basic algorithms. Finally, Stemmer (list) uses the Porter algorithm.

As regards distance between chains the intention is to measure the differences between them. Oliver's algorithm [33] discusses the similarity between two chains, returning the same number of characters to a degree less than or equal to a variable ' n '. Its formulation in our Analyser is: $\text{similar}(n, \text{list})$ returns TRUE if the examined word is considered a degree of similarity less than or equal to ' n ' in at least one word of the list.

The Levenshtein algorithm [34] has many uses, from the detection of plagiarism to the analysis of the DNA strands. It can be applied to chains of different lengths and reflects the number of deletions, insertions or substitutions required to transform a source string into a target string. For example, if the source string is 'drove' and the target string is 'drive', the Levenshtein distance is one, since a single transformation is necessary for both chains to be equal (exchange the 'o' for the 'i' in this case). The greater the value of the distance, the greater the difference between the chains analysed. The expression of Levenshtein's algorithm in our Analyser is as follows: $\text{levenshtein}(n, \text{list})$ returns TRUE if the word is considered to have a 'Levenshtein distance' less than or equal to ' n ' in at least one word on the list.

Hamming's algorithm [35] is defined only for chains of the same length, expressing the number ' n ' of the places where two chains are different. It is formulated in our Analyser as: $\text{hamming}(n, \text{list})$ returns TRUE if the word examined has a 'Hamming distance' less than or equal to ' n ' in at least one word on the list.

Accuracy identifies identical chains and it returns TRUE if the examined word is in the list of words.

Quasi-Accuracy locates identical chains, excluding vowels, and it is formulated as returning TRUE if the examined word is in the list of words without checking the vowels.

The most commonly used algorithm for a function of 'stemmer' or reduction of a word to its root or stem is that of Porter [36], following adaptation by Estrella and Duboue [37] to Spanish. It can extract the suffixes and prefixes of different words that express a common content. In our Analyser the stemmer (list) returns TRUE if the examined word is approved by the Stemmer algorithm in at least one of the words on the list.

With all these features in view, we exemplify how to construct and interpret an equation in our Analyser. So, [$\text{Accuracy}(\text{list1})$ AND $\text{Quasi-Accuracy}(\text{list2})$] | $\text{levenshtein}(3, \text{'word1, word2'})$ is interpreted as though

the concept is considered present in the response if and only if the words of 'list1' are in the exact form of the response AND the words of 'list2' are in almost the exact form of the response, OR if a word in the response contains a Levenshtein distance of three with respect to 'word1' and/or 'word2'.

III. IMPLEMENTATION

The main screen of PSICO-A contains a 'Concept Map Editor' displaying a 'File Menu', a 'Mode Menu' and a 'Help Menu'. In the Maps Concept Design window, buttons appear which generate them. On top is the button entitled 'Concept' which will distribute boxes on the screen and assign concepts to them. In the 'Concept' box a bar of concepts appears. Next to the 'Concept' button are the indicative buttons of 'Relationship', and there are three kinds of relations in form of arrows: a conventional continuous arrow designates a type of causal relation; a white-tipped arrow denotes a type of hierarchical relationship; and a non-continuous arrow designates a descriptive class of relation. As the student chooses the concepts, the boxes containing the same on the screen are drawn out and the student traces the connections between them using the above types of arrows. The concept boxes can be moved, deleted, and changed in size. The screen Concept Map Editor displays the number of concept boxes and arrows of various kinds that the subject has inserted at the end of the task, specifying the names. This is shown in Fig. 1.

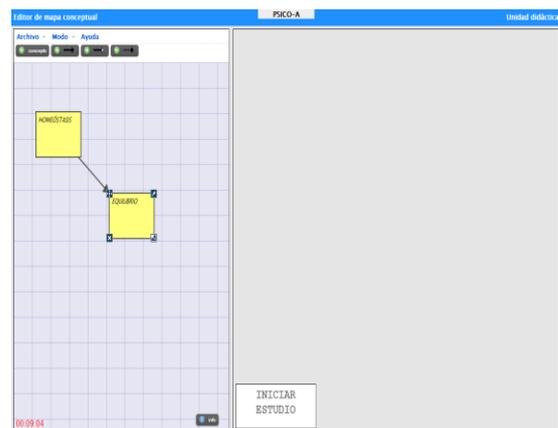


Figure 1. Concept Map Editor.

File displays all the basic components of any conventional file of a program: these are 'New' (which allows you to design a new map), 'Open' (which opens the maps already constructed and stored), 'Save as' (which preserves maps already designed), 'Print' (which prints all system screens), 'Edit My Tab' (where the student writes their name and surname and introduces the password) and 'Exit'.

The game consists of an animation in which MOUSI, our virtual agent, will travel through a maze, as can be observed in Fig. 2.



Figure 2. Game screen.

After the game, the student can see their success in achieving the defined goals. A simulation with MOUSI travelling through a corridor according to the parameters of the experiment by Crespi [29] can be observed in the following Fig. 3.

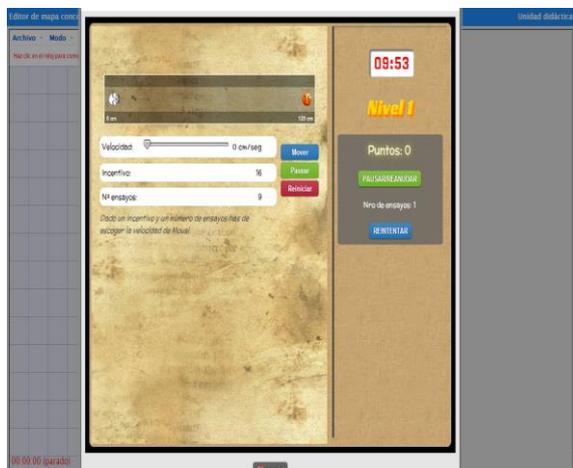


Figure 3. Simulation screen.

The system formulates a series of questions. The responses are scored in the Reflection Mode and the adjusted score Analyser of the system back-end is also reflected in the Learning Assessment Module. This module also finds the correct response or no response from the ‘feedback’ given to the student in Reflection Mode, rephrasing the question, in case of failures.

In the left window, the Help Menu resolves the most common concerns of the system application that a subject may have. In the central part of the main display area is a ‘start survey’ which, when activated, generates above a button entitled ‘Learning’ and a summary of contents, and to the right, the Didactic Unit to be studied. In ‘Prior Knowledge’, the subjects have to state whether they knew anything about the topic before studying it and what (background information). The Notepad allows the free recovery. Our Analyser back-end allows the fidelity of the student’s memory to be judged. This is one of the strengths of PSICO-A: comprehensive

training conjugation with retentive learning. Our computer system applies the combination of a number of functions measuring the similarity and distance between words and chains of words (Accuracy, Quasi-Accuracy, Similarity, Hamming, Levenshtein and Stemmer) to assess the fidelity of the student’s retention in relation to the textual content studied. The ‘Confidence Judgments’ button displays a window that asks the subject to enter their degree of confidence (as a percentage) in their learning of the Didactic Unit [38].

The Metacognitive Judgments area consists of 10 questions based on the questionnaire of the Global Metacognitive Model by Mayor, Suengas and González Marqués [39], and the student must choose and check one of four options that are recorded.

In the members zone intended for students in the back-end and next to the name of each is an area that collects the performance data. Here the data are collected from the Notepad, the concepts located there, the time devoted to study (in seconds), the percentage entered in the Confidence Judgments area, the score obtained in the questionnaire on Metacognitive Judgments, the number of questions answered correctly as identified by the Assessment window and collected from the reflection mode and, finally, the total score in the game mode and in the simulation mode.

IV. METHOD

We evaluated the PSICO-A application using an external evaluation based on [40] and an internal evaluation method.

A. External Evaluation: Results and Discussion

This targets the motivational value of the system to students. It addresses the following questions: (1) does PSICO-A increase your interest in psychology?; (2) do you think that PSICO-A can increase understanding of the subject?; (3) would you like to learn more about psychology? Twenty-two students ($M=17.3$) from the IES ‘Francisco Giner de los Ríos’ High School (Madrid) participated in one session. They worked individually on PSICO-A. Subjects first read the learning material (a lesson about drive-reduction theory by Hull) for 10 minutes. Then they did a pre-test that evaluated the level of their understanding. After a brief orientation on PSICO-A, students started to run the system using the game mode. Later, students worked on a post-test and ten of them participated in an interview for five minutes. The session lasted 55 minutes. User activity processes were saved in each computer and retrieved as data sources.

Eighteen students observed that the system was friendly and that the game was great fun. They generally had no problem using the tools although the design of the conceptual map caused them some trouble (perhaps motivated by the use of three types of relations). Four students had prior knowledge but the knowledge was imprecise and based on biology lessons. The localisation of concepts retrieved from the Notepad was high

($M=14.3$ for 20 concepts) corresponding to a high degree of confidence (72%). The metacognitive results obtained were less striking ($M=5.9$). Without a doubt, metacognitive reflection is not usual in the educational system. In fact, students paid a lot of attention to the questionnaire, spending a lot of time completing it. All of the students finished the game, obtaining 1,420 points on average. In reflection mode five students were not able to obtain five points out of 10. On the Likert scale of one to five (Yes/No) the results showed excellent student evaluation of PSICO-A for $n=22$. The response categories and their values are as follows (see Table I):

- 1=strongly disagree.
- 2=disagree.
- 3=neutral.
- 4=agree
- 5=strongly agree.
- 4.5=Yes.
- 1.5=No.

TABLE I. Results On Psico-A In Post Test

QUESTIONS	MEAN
1) Does manage PSICO-A increase your interest in Psychology?	4.5
2) Do you think that PSICO-A can increase the understanding of the subject?	4.2
3) Would you like to learn more about Psychology?	4.3

We conducted a simple external evaluation of PSICO-A and participants showed great interest in the system. To investigate student perceptions of the system, the interviews were conducted in a semi-structured format [41]. Most students said that the game was fun and believed that the system was useful. Most subjects, according to the results in Table 1, think PSICO-A could be an interesting scaffolding from which to learn more about psychology in the future. Some students commented on their desire to study the subject. For example, one student commented: "Just something I am interested in, really, and I would enjoy it. So I thought if I do something I enjoy I am more likely to better at it." Interestingly, there was also evidence that students may have higher expectations for the subject using the system: "... it is not like other subjects where you sit down and you work just doing the same type of exercise. That makes sense, it is always different." As in a typical classroom, however, many students displayed weak metacognitive thinking. Many of them read the items of the metacognitive questionnaire slowly and perhaps some of their responses were casual. Some students expressed frustration when the system introduced the metacognitive questionnaire: "It was annoying, it kept saying 'you have to think this...you have to study so'." Although all of the students received a brief introduction on how to complete the questionnaire, many students were unsure of how to proceed, asking on what items to pick out. Some students failed to understand the precise meaning of the different arrows in the concept map. This requires multi-tasking and could be rather confusing for potential learners. As expected,

most students used the feedback included in Reflection Mode. It appears then, that the presence of the feedback helped keep students focused on the activity: "...I did not know what to choose for a while so I asked and I got on track." Some students also commented that the amount of concepts from the Didactic Unit was too extensive: "...all this information...they were like, 'home óstasis', 'incentivo' and..."

B. Internal Evaluation: Results and Discussion

Three classes ($M=17.6$), taught by the same teacher, were assigned to experimental groups randomly. Three conditions were used in the study. In the first condition (Group TBG, $n=24$) or Text-Based Group, students were provided with the test-based resource or classical teaching and received a lecture-based teaching unit from the textbook with a duration of 55 minutes. The class included presentation of the subject, review and resolution of queries posed by students. After the explanation, they indicated their percentage of confidence in the learning of the subject and filled in the metacognitive questionnaire adapted to PSICO-A.

In the second group (Group PBG, $n=21$) or PSICO-A Based Group, students obtained their resources through PSICO-A.

Students in the third group (Group PGG, $n=24$) or PSICO-A plus Game Group obtained their resources from PSICO-A incorporating the game.

The meetings were performed in parallel, i.e., the same day at the same time, and two teachers attending the sessions in the computer room.

A pre-test and a post-test, facilitating objective comparison of the learning, was administered to all students. They consisted of 30 multiple choice questions (each offering three options). Table II reveals the post-test gains for the three groups:

TABLE II. Post-Test Gains For Tbg, Pbg And Pgg

TEST	GROUP	N	PERCENTAGE
Pre-test	TBG	24	65
	PBG	21	61
	PGG	24	60
Post-test	TBG	24	74
	PBG	21	82
	PGG	24	86

ANOVA revealed that there were significant gains in pre- to post-test scores for all three groups ($F(1,66)=48.015$, $p<.001$). A significant interaction was found between scores and group: $F(2,66)=3.367$, $p<.039$. One-way ANOVAs revealed that there was a significant difference between groups on the number of concepts generated in the Notepad ($F(2,29)=5.781$, $p<.008$) but no significant difference in the number of correct definitions generated ($F(2,29)=2.434$, $p<.106$). A Tukey HSD comparison between groups performed on the number of concepts in the Notepad, showed that the PGG group generated significantly more concepts than

either the TBG and PBG groups.

The analysis indicates that students learning outcomes were related to whether they used the computational system or not. There is a positive change from the pre-test to the post-test. It would be worth further examining whether students using PSICO-A have smaller learning outcome immediately after the session. Students using computer game in PSICO-A seem to remember significantly better the studied concepts compared to students using more traditional teaching methods or even compared to students using PSICO-A without game. This could, however, be related to motivational differences between groups. Perhaps the PGG group could be said to have overall a stronger intrinsic motivation for using the game. The game does not interfere with explicit generation, that is, it improves the retention process but it does not improve the understanding mechanism. In a future study, it is necessary to analyse the retention of students in the course for eliminating potential differences in the initial baseline. Also an internal evaluation of the system will be required to provide a comparison between games and simulations to foster learning in psychology.

V. CONCLUSIONS

In this article we have presented an Intelligent Tutoring System, called PSICO-A, for learning psychology. It is a pioneer educational system combining digital games and simulations. Programmed in PHP5 language, its computational architecture consists of a front-end and a back-end. The first contains a design mode for building concepts, a reflective mode, a game mode and a simulation mode. These modes are connected to the back-end, which is the core of the system and analyses the performance of the pupils. PSICO-A assembles Boolean equations introducing algorithms such as those of Oliver, Levenshtein, Hamming and Porter.

PSICO-A takes as a model recent and powerful computational educational systems like MetaTutor, Betty's Brain and REAL but its design is based on many pedagogical influences, such as conceptual maps by Novak [24] and representational theory by Black [25], introducing images and mental models as a representation of knowledge and its influence on the construction of virtual worlds for learning, the relevance of metacognition, Kapicke and Blunt's findings about free retrieval practice for learning and, of course, the constructivist paradigm in education [42,43].

We conducted an external evaluation of the system and verified the motivational value of the system. Besides, it was conducted an internal evaluation comparing the learning outcomes of three experimental groups. Students in the first group were provided with text-based resources and the other conditions involved students who worked in collaboration with PSICO-A (plus game and without game). PSICO-A improved the learning and the combination with the game increased the generativity of concepts but further studies are

required. In a future, however, an internal evaluation will be required to provide a comparison between games and simulations using PSICO-A.

ACKNOWLEDGMENT

We acknowledge to Dr. Andrés Leonardo Sclipa for his invaluable aid for the development of the system.

REFERENCES

- [1] Sleeman, D. and Brown, J.S. (Eds.), *Intelligent Tutoring Systems*, Academic Press, London, 1982.
- [2] Polson, M.C. and Richardson, J.J. (Eds.), *Foundations of Intelligent Tutoring Systems*, Lawrence Erlbaum Associates, Mahwah, NJ, 1988.
- [3] Urretavizcaya, M., "Sistemas inteligentes en el ámbito de la educación", *Revista Iberoamericana de Inteligencia Artificial*, 5: 5-12, 2001.
- [4] Cataldi, Z., Salgueiro, F., Britos, P., Sierra, E. and García Martínez, R., "Selecting pedagogical protocols using SOM", *Research in Computing Science Journal*, 21: 205-214, 2006.
- [5] Nkambou, R., Bourdeau, J., and Mizoguchi, R. (Eds.), *Advances in Intelligent Tutoring Systems*, Springer-Verlag, Berlin, 2010.
- [6] Ausubel, D., Novak, J., and Hanesian, H., *Educational Psychology: A Cognitive View* (2nd Ed.), Holt, Rinehart & Winston, New York, 1978.
- [7] Azevedo, R., Witherspoon, A., Chauncey, A., Burkett, C., and Fike, A., "MetaTutor: a metacognitive tool for enhancing self-regulated learning". In R. Pirrone, R. Azevedo, and G. Biswas (Eds.), *Proceedings of the AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems* (pp. 14-19). AAAI Press, Menlo Park, CA, 2009.
- [8] Biswas, G., Schwartz, D.L., Leelawong, K., Vye, N. and TAG-V, "Learning by teaching: A new agent paradigm for educational software", *Applied Artificial Intelligence*, 19: 363-392, 2005, <http://dx.doi.org/10.1080/08839510590910200>.
- [9] Bai, X., Black, J.B., and Vitale, J., "REAL: Learn with the Assistance of a reflective agent", *Agent-based systems for human learning Conference*, Hawaii, 2007.
- [10] Zimmerman, B.J., and D.H. Schunk (Eds.), *Self-regulated learning and academic achievement: Theory, research and practice*, Springer-Verlag, New York, 1989.
- [11] Bembennuty, H., *Self-regulated learning: New directions for teaching and learning*, Jossey-Bass, San Francisco, 2011.
- [12] Azevedo, R., Witherspoon, A., Graesser, A., McNamara, D., Chauncey, A., Siler, E., Cai, Z., Rus, V., and Lintean, M., "MetaTutor: Analyzing self-regulated learning in a tutoring system for biology". In V. Dimitrova, R. Mizoguchi, B. du Bolay, and A. Graesser (Eds.), *Building learning*

- systems that care: from knowledge representation to affective modelling (pp. 635-637), IOS Press, Amsterdam, 2009, <ftp://129.219.222.66/pdf/MetaTutor%20Analyzing%20SelfRegulated%20Learning%20in%20a%20Tutoring%20System%20for%20Biology.pdf>.
- [13] Gartner, A., Conway, M. and Riessman, F., *Children teach children. Learning by teaching*, Harper & Row, New York, 1971.
- [14] Barr, R. and Tagg, J., "From teaching to learning: a new paradigm for undergraduate education", *Change*, 27: 12-25, 1995.
- [15] Grzega, J., and Schöner, M., "The didactic model Ldl (Lernen durch Lehren) as a way of preparing students for communication in a knowledge society", *Journal of Education for Teaching*, 34: 167-175, 2008, <http://www.joachimgrzega.de/GrzegaSchoenerLd.pdf>.
- [16] Blair, K., Schwartz, D., Biswas, G., and Leelawong, K., "Pedagogical agents for learning by teaching: teachable agents", *Educational Technology and Society, Special Issue on Pedagogical Agents*, 2006.
- [17] Davis, J.M., Leelawong, K., Belyne, K., Bodenheimer, R., Biswas, G., Vye, N., and Bransford, J., "Intelligent user interface design for teachable agent systems", *International Conference on Intelligent User Interfaces* (pp. 26-33), ACM, Miami, Florida, doi: 10.1.1.14.8457.pdf.
- [18] Kinnebrew, J.S., and Biswas, G., "Modeling and measuring self-regulated learning in teachable agent environments", *Journal of E-learning and Knowledge society*, 7: 19-35, 2011.
- [19] Tan, J., Beers, C., Gupta, R., and Biswas, G., "Computer games as intelligent learning environments: a river ecosystem adventure". In C.-K. Looi et al. (Eds.), *Artificial Intelligence in education*, IOS Press, Amsterdam, 2005.
- [20] Biswas, G., Roscoe, R., Jeong, H., and Sulcer, B., "Promoting self-regulated learning skills in agent-based learning environments". In S.C. Kong et al., *Proceedings of the 17th International Conference on Computers in Education*, Hong Kong, Asia-Pacific Society for Computers in education, 2009.
- [21] Bai, X., and Black, J.B., "REAL: a generic Intelligent Tutoring System framework". In C. Crawford et al. (Eds.), *Proceedings of Society for Information Technology and Teacher Education International Conference* (pp. 1279-1283), AACE, Chesapeake, 2005.
- [22] Bai, X., and Black, J.B., "Enhancing Intelligent Tutoring Systems with the agent paradigm". In V.V.A.A., *Gaming and simulations: Concepts, methodologies, Tools and Applications* (pp. 46-66), IGI Global, London, 2011.
- [23] Black, J.B., "Imaginary worlds". In M.A. Gluck, J.R. Anderson, and S.M. Kosslyn (Eds.), *Memory and mind*, Lawrence Erlbaum Associates, Mahwah, NJ, 2007, [JblackImagination\(2\).doc](#).
- [24] Novak, J., *A theory of education*, Cornell University Press, New York, 1977.
- [25] Black, J.B., *Types of knowledge representation*, CCTE Report, New York, Teachers College, Columbia University.
- [26] Dunlosky, J. and Metcalfe, J., *Metacognition*, SAGE, London, 2008.
- [27] Karpicke, J.D. and Blunt, J.R., "Retrieval practice produces more learning than elaborative studying with concepts mapping", *Science*, 331: 772-775, 2011, <http://dx.doi.org/10.1126/science.1199327>.
- [28] Lerdorf, R., Tatroe, K., and MacIntyre, P., *Programming PHP*, O Reilly Media, New York, 2006.
- [29] Crespi, L.P., "Variation of incentive and performance in the white rat", *The American Journal of Psychology*, 55: 467-517, 1945.
- [30] Hull, L.C., *Principles of behavior: an introduction to behavior theory*, D. Appleton-Century Company, New York, 1943.
- [31] Aldrich, C., *Learning online with games, simulation, and virtual worlds*, John Wiley & Sons, San Francisco, 2009.
- [32] De Freitas, S.I., "Using games and simulations for supporting learning", *Learning Media and Technology*, 31: 343-358, 2006.
- [33] Oliver, I., *Programming classics: implementing the world's best algorithms*, Prentice-Hall, Saddle River, NJ, 1994.
- [34] Levenshtein, V., "Binary codes capable of correcting deletions, insertions and reversals", *Soviet Physics Doklady*, 10: 707-710, 1966, <http://profs.sci.univr.it/~Liptak/ALBioinfo/files/levenhstein66.pdf>.
- [35] Hamming, R.W., "Error detecting and error correcting codes", *Bell System Technical Journal*, 29: 147-160, 1950, <http://www.lee.eng.uerj/hamming.pdf>.
- [36] Porter, M.F., "An algorithm for suffix stripping", *Program*, 14: 130-137, 1980, <http://dx.doi.org/10.11>.
- [37] Estrella, P. and Duboue, P.A., "Experiments on language normalization for Spanish to English machine translation", *Revista Iberoamericana de Inteligencia Artificial*, 9: 23-37, 2005, <http://dx.doi.org/10.4114/IA.V9I26.843>.
- [38] Nelson, T.O., and Narens, L., "A new technique for investigating the feeling of knowing", *Acta Psychologica*, 46: 69-80, 1980.
- [39] Mayor, J., Suengas, A., and González Marqués, J., *estrategias metacognitivas: aprender a aprender y aprender a pensar, Síntesis*, Madrid, 1993.
- [40] Bai, X., Black, J.B., Vikaros, L., Vitale, J., Li, D., and Xia, Q., "Learning in one's own imaginary world", *American Educational Research Association*, Chicago, 2007, doi: 10.1.1.135.1024 (1). pdf.
- [41] Drever, E., *Using semi-structured interviews in small-scale research*. The Scottish Council for Research in Education, Edinburgh, 1997.
- [42] Bruner, J., *Actual minds, possible worlds*, Harvard University Press, 1980.

- [43] Glaserfeld, E., *Constructivism in education*, Pergamon Press, Oxford, 1989.



Dr. Javier González Marqués is Chair of Psychology at the Complutense University of Madrid (Department of Basic Psychology II). In research, his current interests include theoretical and applied neuroscience, neurology,

cognitive functions and metacognition. Dr. González Marqués received his PhD degree in Psychology from the Complutense University of Madrid, Spain. He is a foundational member of the Master of Neuroscience organized by the Complutense University and member of numerous scientific societies.



Mr. Carlos Pelta has BA degree from Autónoma University and BS from UOC, Spain. He had been working for the application of Artificial Intelligence to Psychology for over 10 years. He is currently preparing a PhD in

Psychology at the Department of Basic Psychology II, Complutense University.