Guzmán, E., Conejo, R., & García-Hervás, E. (2005). An Authoring Environment for Adaptive Testing. *Educational Technology & Society*, 8 (3), 66-76.

An Authoring Environment for Adaptive Testing

Eduardo Guzmán, Ricardo Conejo and Emilio García-Hervás
Departamento de Lenguajes y Ciencias de la Computación. Universidad de Málaga.
E.T.S.I. Informática. Bulevar Louis Pasteur, 35. 29071, Málaga. Spain
guzman@lcc.uma.es
conejo@lcc.uma.es

ABSTRACT

SIETTE is a web-based adaptive testing system. It implements Computerized Adaptive Tests. These tests are tailor-made, theory-based tests, where questions shown to students, finalization of the test, and student knowledge estimation is accomplished adaptively. To construct these tests, SIETTE has an authoring environment comprising a suite of tools that helps teachers create questions and tests properly, and analyze students' performance after taking a test. In this paper, we present this authoring environment in the framework of adaptive testing. As will be shown, this set of visual tools, that contain some adaptable features, can be useful for teachers lacking skills in this kind of testing. Additionally, other systems that implement adaptive testing will be studied.

Keywords
Adaptive testing, Authoring environment, Item Response Theory, Adaptability

Introduction

Testing is one of the most widely used tools in higher education (Brusilovsky & Miller, 1999). The main goal of testing is to measure student knowledge level in one or more concepts or subjects, i.e. in pieces of knowledge that can be assessed. This kind of assessment has been used for student knowledge diagnosis in adaptive educational systems, such as EML-ART (Weber & Brusilovsky, 2001) or DCG (Vassileva, 1997), but most of these systems are based on heuristic-based testing techniques. However, there is another kind of test, namely the adaptive test, which is based on a theoretically-sound theory, the *Computerized Adaptive Testing* (CAT) theory (van der Linden & Glas, 2000). This theory defines which questions (called *items*) are the most adequate to be posed to students, when the tests must finish, and how student knowledge can be inferred from students' performance during the test. To this end, CAT uses an underlying psychometric theory called *Item Response Theory* (IRT) (Hambleton et al., 1991).

Adaptive test elicitation is a task that requires a special effort on the part of the teacher, since the construction of this kind of test must be accomplished in accordance with some features. These features must be kept to ensure the correct operation of adaptive tests. For instance, teachers must ensure that the stem of one item does not provide any clue to correctly answering other items, i.e. items must be independent of each other. Additionally, adaptive testing selection techniques must have a significantly large set of items available, with a wide range of difficulties. These requirements demand that adaptive testing systems have an authoring environment that helps teachers construct items and tests. This kind of system needs some tools to analyze student test session data in order to check if the set of items contains the necessary properties. Unfortunately, only a small number of systems are able to generate adaptive tests (Brusilovsky & Miller, 1999). Furthermore, references to these systems do not provide information about their authoring interfaces, and consequently, we cannot know if they include adaptive and/or adaptable features.

SIETTE (Conejo et al., 2004) is a web-based system for adaptive test generation. Moreover, this system is able to deliver conventional (heuristic-based) tests. Through a web interface, students can take tests for self-assessment, where item correction is shown after each item, with some kind of optional feedback; or teachers can make grading tests in order to assess their students, even for academic purposes. To construct and modify the test contents, SIETTE offers an authoring environment. This is a suite of tools that mainly permits teachers to edit tests. This environment includes a tool for analyzing student performances.

This paper is aimed at showing the authoring environment of the SIETTE system. The next section briefly explains what adaptive tests are. Next the components of the SIETTE architecture will be shown. Afterwards, the test editor will be described, showing its operation mode and its adaptable capabilities (Oppermann et al., 1997) together with the result analyzer, i.e. a tool that allows teachers to study student performance in tests they have taken. A brief state-of-the-art description of systems implementing adaptive testing is also included and an evaluation of the authoring tool in terms of its effectiveness and user satisfaction. Finally, in the last section, the conclusions of this work are summarized.

What is an adaptive test?

CAT theory tries to mimic the usual assessment procedure followed by a human teacher. That is, it first gives the student an item of medium difficulty. If the student answers correctly, it then administers an item that is a little more difficult and if not, it administers a less difficult item. This process should be repeated until the teacher considers that he/she has enough evidence to determine the student's knowledge level. In CAT theory, this process has been automatized. Items are posed one by one. After posing an item, a temporary student knowledge level estimation is achieved. In terms of this estimation, the next item to be posed is chosen in such a way that this estimation will be more accurate. In more precise terms, an adaptive test can be seen as an iterative algorithm that starts with an initial estimation of the student's knowledge level, and comprises the following steps:

1. All the items that have not been administered yet are examined to determine which is the best item to ask next, according to the current estimation of the student's knowledge level.
2. The item is asked, and the student responds.
3. According to the answer, a new estimation of the knowledge level is computed.
4. Steps 1 to 3 are repeated until the test stopping criterion defined is met.

IRT postulates that there is a relationship between the student's knowledge level and the probability of successfully answering an item. This interdependent relationship is probabilistically expressed by means of a function called *Item Characteristic Curve* (ICC). Accordingly, this function collects, for each knowledge level, the probability that a student with this level will correctly answer the item. If this probability function is available for every item of a test, the student's knowledge can be directly inferred. In CAT theory, IRT is used to estimate the student's knowledge level, in order to determine the next item to be posed, and to decide when to finish the test. This theory ensures that the student knowledge estimations obtained do not vary in terms of the items used in the estimation process. The models most commonly used as ICC functions are the family of logistics models of one (1PL), two (2PL) and three parameters (3PL). All of them can be expressed by the following equation:

$$P(\mathrm{u}_\mathrm{i} = 1 \mid \theta) = c_i + (1 - c_i)\frac{1}{1 + e^{-1.7\, a_i(\theta - b_i)}} \qquad \textbf{(1)}$$

where c_i is the guessing factor, b_i is the item difficulty and a_i is its discrimination factor. The guessing factor is the probability that a student with no knowledge at all will answer the item correctly. The difficulty represents the knowledge level in which the student has the same probability of passing or not the item, besides the guessing factor. The discrimination factor is a value proportional to the slope of the curve and represents a measure of how the item contributes to estimating the knowledge level. The formula just shown in Equation 1 expresses the 3PL model. When the guessing factor is always assumed to be zero, the 2PL model is obtained. If, in addition, we consider the discrimination factor equal to 1, we obtain the 1PL model.

The main advantage of adaptive tests is that they are fitted to students individually. This means that the number of items posed is different for each student, and depends on his/her knowledge level. In consequence, students neither get bored from being given very easy items, nor feel stressed for being asked very difficult items. In addition, different sets of items are posed to different students. Therefore, this reduces the possibility of cheating. In contrast, the main disadvantage of an adaptive test is that its construction is costly. Each ICC must be determined (calibrated) before the test can be applied. To this end, a large student population must be given this test non-adaptively, after which the calibration can be accomplished using this data.

SIETTE architecture

SIETTE allows eliciting and delivering CAT through web interfaces. It can work as a standalone assessment tool or inside other web-based adaptive systems as a diagnosis tool. It is a multilingual system, currently available in Spanish and English, but open to include other new languages. Figure 1 represents the system's architecture. It comprises two main parts: the student workspace and the authoring environment.

The *student workspace*: This is the place where students can take tests. The main component of this part is the *test generator*, which is in charge of delivering tests. Two interfaces can be used to access tests generated:

- *Student classroom*: Here, students can take tests for self-assessment, and teachers can administer tests for grading.

- *Interface for external connections*: This interface permits SIETTE to work as a diagnosis tool inside other web-based adaptive hypermedia educational systems. A simple protocol (Guzmán & Conejo, 2002a) has been specified and implemented for this purpose.

The *authoring environment*: It consists of a suite of tools used by teachers. They allow content creation and updating, as well as analyzing the performances of students that have taken tests. This suite is composed of the following tools:

- The *test editor*: With this tool, teachers can create subjects. Different sets of items can be defined in relation with the topics of each subject. Teachers can also define different tests that involve the subject topics.
- The *result analyzer*: This tool helps teachers analyze student performance.
- The *item calibration tool*: As shown, ICC functions predict the behavior of students that answer the corresponding item. They are determined by a set of parameters. These parameters are inferred by calibration techniques (Glas, 2000). In this part of the architecture, some of these calibration techniques are being developed. Unfortunately, this tool is currently under development and, therefore will not be addressed in this paper.

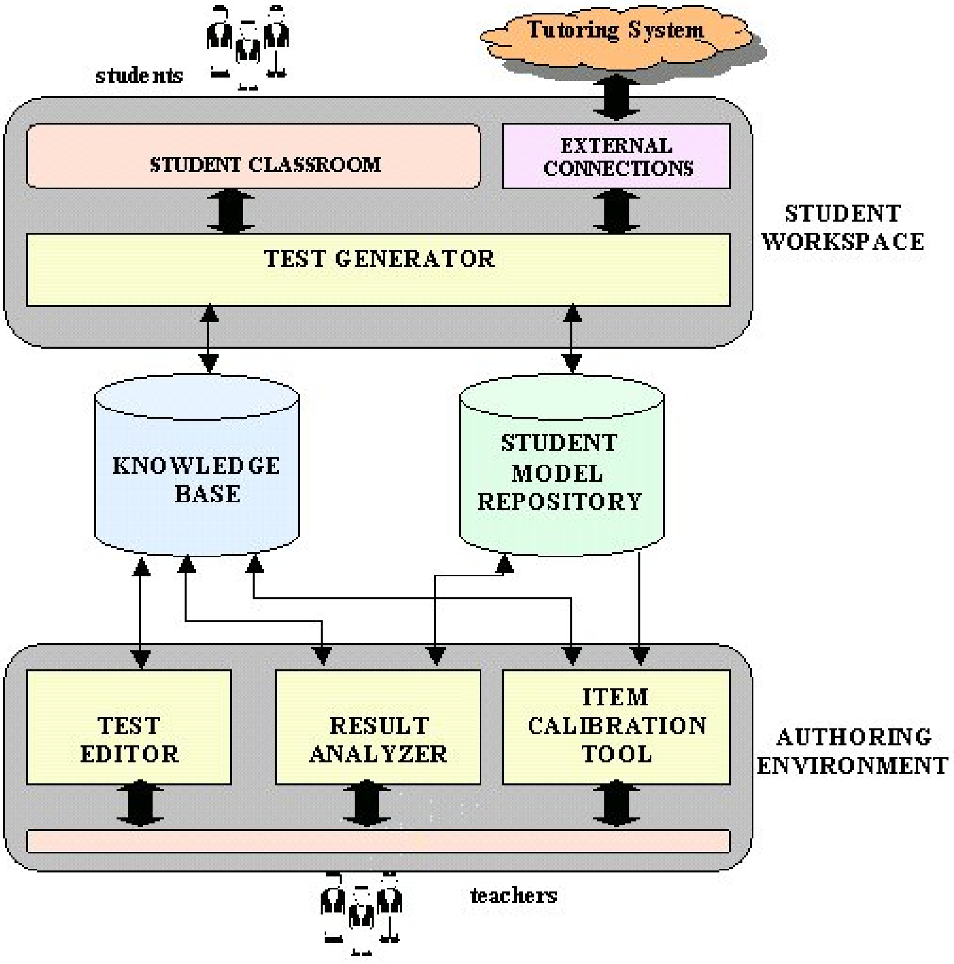
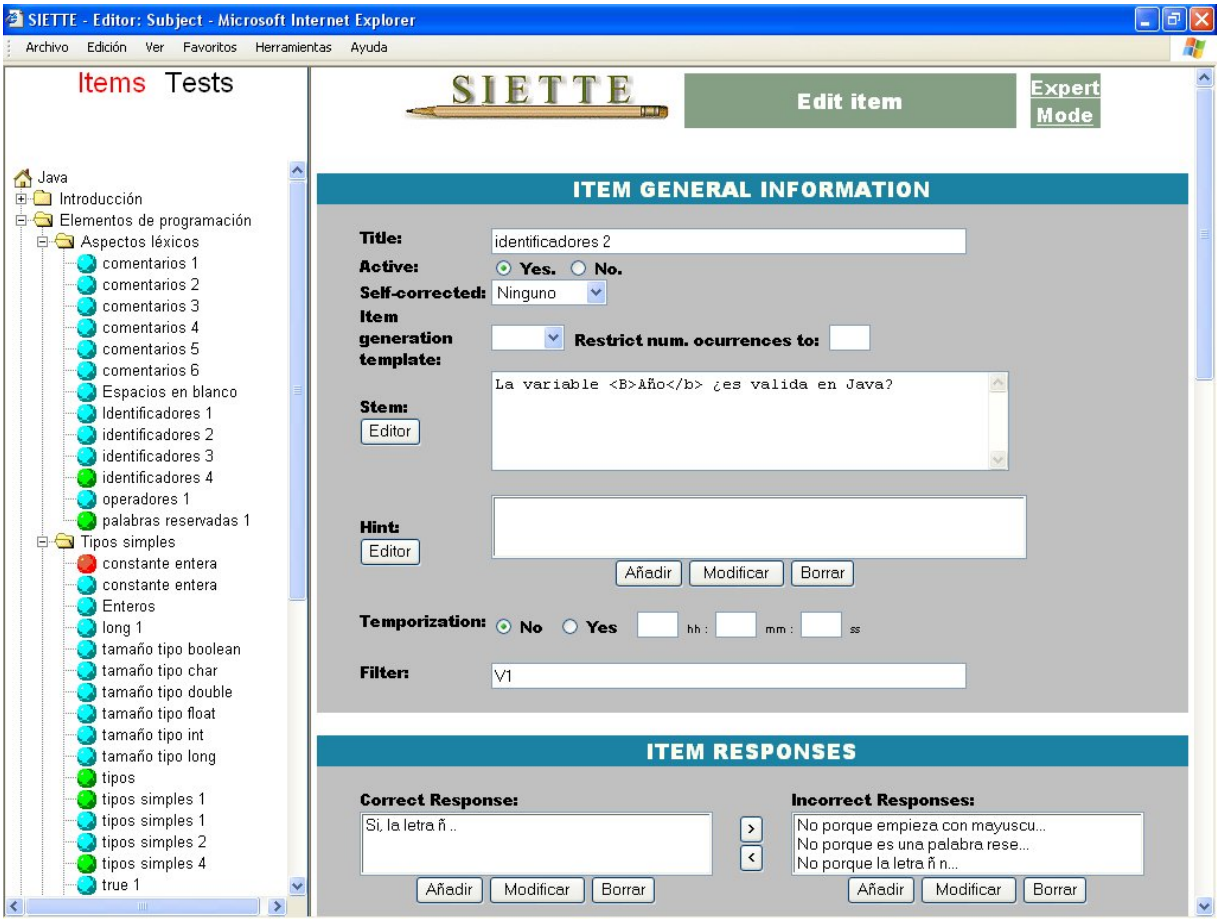


Figure 1. SIETTE architecture

The test editor

In SIETTE, teachers can define different subjects. The curricula of subjects, tests and items are stored in the knowledge base. Subject curricula are structured forming acyclic graphs of topics. Therefore, a subject can be divided into topics. Each topic can also be divided into subtopics and so on. As a result, each curriculum can be seen as a granularity hierarchy (McCalla & Greer, 1994), where topics are related to their subtopics via aggregation relations. Items are assigned to topics in such a way that if an item is assigned to a topic, this item is used to assess the student's knowledge level in the topic. Items can be assigned to any topic of the hierarchy, including the subject, since it can be seen as a global aggregation of the whole curriculum. At last, tests are defined in terms of topics. If a test involves a set of topics, after a testing session, SIETTE is able to return a student's knowledge estimation for each test topic, and for each one of their descendant subtopics at any level (Guzmán & Conejo, 2004b).

In order to access this tool, teachers must be provided with an identifier/password pair given by the system administrator. A snapshot of this tool, after selecting a subject to edit, is shown in Figure 2. It is divided into two main frames. The left one is the curriculum hierarchy tree. Two different views of this tree can be seen: items or tests. When the "items" option is selected, the tree shows the subject curriculum hierarchy, composed of the topics and their items. Topics are represented by folders, and items by colored balls. The ball color differs in terms of the kind of item. If the "tests" option is selected, the tree shows the tests that have been defined for this subject. Under each test, the curriculum of the test topics is shown. Finally, the aspect of the right side depends on the element selected in the tree. Subject, topic, item and test information can be added, modified or deleted through this frame.



Java
Introducción
Elementos de programación
Aspectos léxicos
comentarios 1
comentarios 2
comentarios 3
comentarios 4
comentarios 5
comentarios 6
Espacios en blanco
Identificadores 1
identificadores 2
identificadores 3
identificadores 4
operadores 1
palabras reservadas 1
Tipos simples
constante entera
constante entera
Enteros
long 1
tamaño tipo boolean
tamaño tipo char
tamaño tipo double
tamaño tipo float
tamaño tipo int
tamaño tipo long
tipos
tipos simples 1
tipos simples 1
tipos simples 2
tipos simples 4
true 1

Title: identificadores 2
Active: ● Yes. ○ No.
Self-corrected: Ninguno
Item generation template: Restrict num. ocurrences to:
Stem: La variable <B>Año</b> ¿es valida en Java?
Hint:
Temporization: ● No ○ Yes
Filter: V1

Correct Response: Si, la letra ñ ..
Incorrect Responses:
No porque empieza con mayuscu...
No porque es una palabra rese...
No porque la letra ñ n...

Figure 2. The test editor

The element parameters of the editor can been seen in two different ways, depending on the teacher profile. The test editor adapts the content presentation in terms of the teacher profile. Two different teacher stereotypes (Kay, 2000) are managed: novice and expert. In terms of their mastery in using this tool, teachers can select either one stereotype or the other, and are free to change it at any time. The difference between the two is the degree of detail of the information shown. In the novice profile, some information is hidden. When a teacher with this profile is editing an element, some of its parameters will take values by default. The expert profile has been conceived for teachers with more advanced mastery of the system and/or in the use of adaptive tests.

Different teachers can access the same subject. For each subject, there is a teacher who is its creator and has all permissions granted. Through the editor, he/she can grant different permissions to other teachers. In consequence, the set of actions a teacher can accomplish is adapted to the permission he/she has for the subject. These permissions are item reading or item modification, curriculum modification, test addition or test modification, etc. The editor takes into account teacher permissions in order to adapt the interfaces it shows teachers. This is done by hiding/showing the actions the teacher can accomplish. For instance, if a teacher cannot modify items, when he/she is editing an item the update button is hidden.

Test definition

Chua Abdullah (2003) has pointed out the types of knowledge that teachers must take into account to achieve effective testing assessment: (1) *what to test*, in other words the parts of the domain knowledge to be tested; (2) *who to test*, this is the student model; (3) *how to test*, namely the item selection criterion and the student assessment method. We have added another one: (4) *when to finish the test*, i.e. the test finalization decision. In adaptive testing, this last decision is vital, since it will determine the accuracy and, as a result, the reliability of student assessment. In SIETTE's test definition stage all these concerns are expressed by test configuration parameters. The first concern (*what*) is represented by the topics involved in the test, and the number of knowledge levels in which the students will be assessed. Although the real number domain is used in IRT, in SIETTE a discrete domain is used for the sake of simplicity. Accordingly, if the number of knowledge levels is equal to *K*, students will be classified between *0* and *K-1*.

The *who* is clearly the student represented by a student model. Student models in SIETTE are essentially probability distribution curves which contain, for each knowledge level, the probability that student knowledge will coincide with this knowledge level. For each topic assessed in a test, SIETTE keeps a student distribution curve. When creating a test, SIETTE provides teachers with the possibility of selecting the prior probability distribution their students will have before posing any item.

Finally, the *how* and *when* concerns will be discussed in the following three subsections:

Item selection criteria

SIETTE provides two different adaptive item selection criteria:

- *Owen's adaptive criterion*: It uses a discrete version of Owen (1975) item selection approach. It selects the item that minimizes the expectation of the variance of the posterior student knowledge distribution.
- *Difficulty-based criterion*: Owen found a simplification of his previous selection criterion (*op. cit.*), whose performance is very near to the former, and which is very simple to apply. It selects the item whose difficulty (a parameter of the ICC) is the nearest to the current student knowledge level estimation.

Student assessment techniques

In the *how* concern, we do not only have to consider the item presentation order; it is also necessary to decide what mechanism must be used to infer student knowledge level. In SIETTE, the adaptive assessment methods are based on a discrete Bayesian mechanism in which student knowledge probability distribution is calculated after posing each item *i* (Equation 2). The estimation made after posing the last test item becomes the final estimation.

$$\mathrm{P}(\theta|\mathrm{u}_1,..,\mathrm{u}_i) \propto P(\mathrm{u}_i=1|\theta)^{u_i}(1-P(\mathrm{u}_i=1|\theta))^{(1-u_i)}P(\theta|u_1,..u_{i-1}) \qquad \textbf{(2)}$$

In Equation 1, $P(\theta|u_1, ..,u_{i-1})$ is the temporary student estimation before answering item *i*, and $u_i=1$ indicates that the student has answered the item correctly. $P(u_i=1|\theta)$ is the ICC for item *i*. As mentioned before, it expresses the relationship between the item's correct answer and knowledge levels. Once the new estimation distribution is calculated, the student knowledge level can be inferred in two different ways in terms of the adaptive criteria used: *modal*, that is, the most likely level; or *expectation-based*, where the estimated knowledge level is equal to the expected probability distribution value.

In SIETTE, items are assigned to the topics they assess. If an item *Q* is used to assess a topic *T*, applying the aggregation relations defined in the curriculum, item *Q* can be used to assess all the topics preceding *T*. In order to manifest this relation, each item has an ICC for each topic it can assess. Accordingly, after a single test, the system is able to return the state of student knowledge in the test topics and in all their descendants (Guzmán & Conejo, 2002b).

Test finalization criteria

In order to ensure test finalization, and to avoid item overexposure, a maximum item number is defined for each test. While a test is being administered, every time an item is selected, this upper limit is compared to the number of items already administered. If this last number is equal to or greater than the limit, the test has to finish. While this condition is not satisfied, test finalization can be decided by one of the following criteria:

- student's knowledge estimation variance is less than a certain threshold;
- student's estimated knowledge level probability is greater than a certain threshold;
- or, for temporized tests, the time limit has been reached.

Whereas the two former criteria are purely adaptive, the last one, although non-adaptive, can be applied to adaptive tests as an alternative mechanism to avoid very long tests. SIETTE offers the possibility of configuring tests to be temporized. To this end, teachers only have to set the test time limit through the editor.

Additionally, other configuration parameters can be set for each test: its availability can be restricted to one or more groups of students; filters can be configured to restrict the items that can be administered in each test; teachers can allow students to retake a test at the same stage they stopped, if the test has been suspended for any reason (for instance, connection failure); etc.

Item definition

In SIETTE teachers are supplied with several types of items with which to construct their tests:

- *True/false items*, where students have to select just one answer.
- Multiple-c*hoice items*, where students must select an answer or none.
- *Multiple-response items*, where more than one answer can be correct.
- *Self-corrected items*, which are little programs, implemented by means of java applets or flash, that allow teachers to include more sophisticated exercises. They are corrected on their own, and this correction is given to SIETTE.

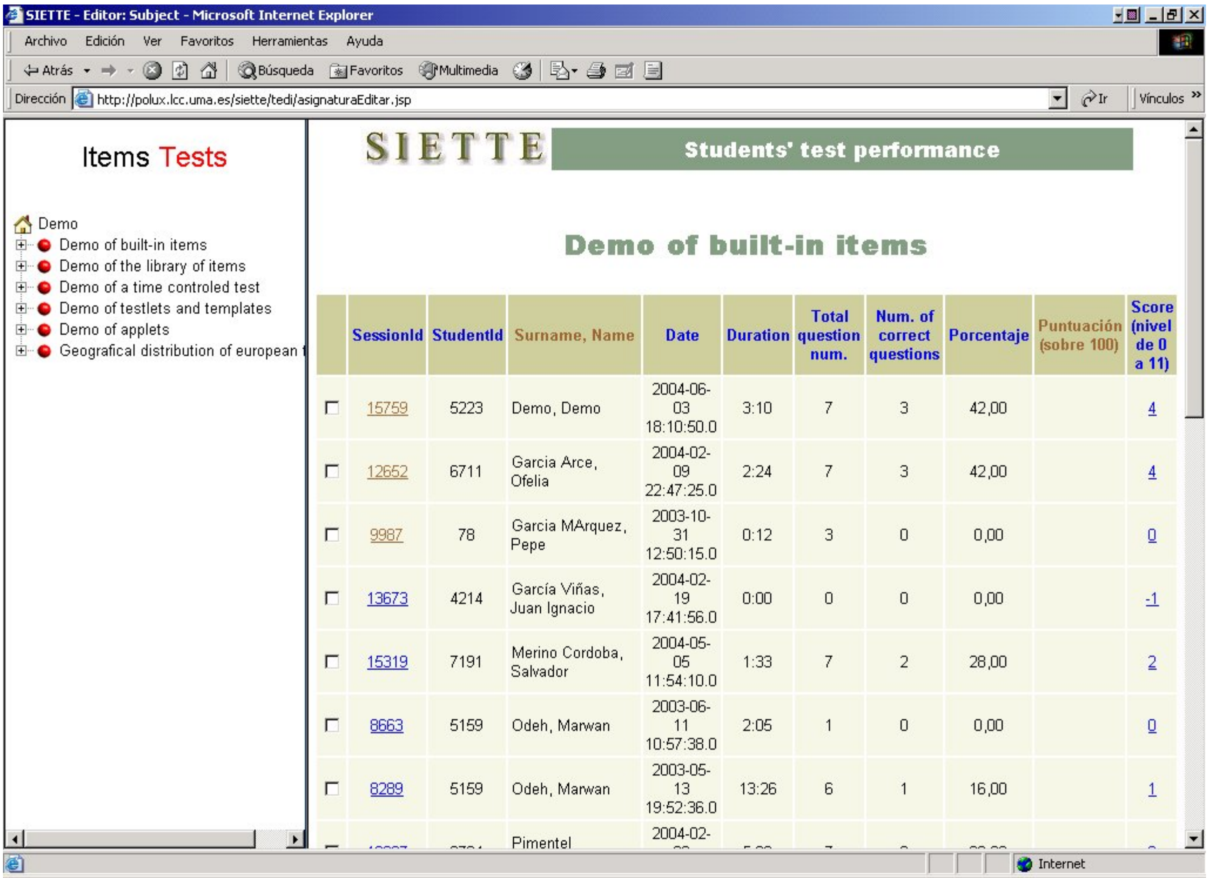
These types of items can be combined in the same test. Former items have the classical format of a stem and a set of answers. SIETTE offers other kinds of item construction scaffoldings and a library of exercise templates. It collects most of the exercises that usually appear in textbooks. They can easily be added to a test by instating the desired template. Additionally, SIETTE includes an item generation mechanism. This mechanism has been implemented through item templates written in a web language (e.g. JSP, PHP, etc.). These templates generate questions of the sort described previously after being pre-processed. For more information about the types of items and item generation see (Guzmán & Conejo, 2004a).

The result analyzer

The student model repository stores information about student test sessions. SIETTE's result analyzer allows teachers to study these data. It contains the following two utilities:

A student performance facility: For each test it contains the list of students that have taken the test. For each student it shows the identifier of the test session, his/her identifier and name, the date of the beginning of the test session, the total number of items posed, the number of items correctly answered and his/her final grade. It allows watching the complete test session, that is, the items that were given to the student in the same order posed, with the student's response and the correct response. This tool gives detailed statistics of the final student knowledge level estimation. For each topic, the estimation, the number of items posed and the number of correctly answered topic items, as well as a graphical representation of the estimated knowledge distribution, is provided. Additionally, it offers the possibility of deleting student test information from the student model repository.

An item statistics facility: It supplies statistical information about student responses to the item in all test sessions in which the item has been posed. These data can be studied for each topic to which the item is directly associated, and for each one of its preceding topics, including the subject. Once the topic to be studied has been selected, a table is shown. It contains a column for each item answer. Each row represents a knowledge level in which the subject can be assessed. Each cell c_{ij} of the table represents the number of students with final estimated knowledge level *i* that have selected the answer *j*. In addition, cumulative statistics are shown: that is, taking all the data of the student model repository as a sample, the likelihood that a student will select an answer given his final estimated knowledge level. This information is very useful for calibration purposes.


SIETTE - Editor: Subject - Microsoft Internet Explorer
Archivo
Edición
Ver
Favoritos
Herramientas
Ayuda
Atrás
Búsqueda
Favoritos
Multimedia
Dirección
http://polux.lcc.uma.es/siette/tedi/asignaturaEditar.jsp
Ir
Vínculos
Items
Tests
Demo
Demo of built-in items
Demo of the library of items
Demo of a time controled test
Demo of testlets and templates
Demo of applets
Geografical distribution of european
SIETTE
Internet

Students' test performance

Demo of built-in items

	SessionId	StudentId	Surname, Name	Date	Duration	Total question num.	Num. of correct questions	Porcentaje	Puntuación (sobre 100)	Score (nivel de 0 a 11)
□	15759	5223	Demo, Demo	2004-06-03 18:10:50.0	3:10	7	3	42,00		4
□	12652	6711	Garcia Arce, Ofelia	2004-02-09 22:47:25.0	2:24	7	3	42,00		4
□	9987	78	Garcia MArquez, Pepe	2003-10-31 12:50:15.0	0:12	3	0	0,00		0
□	13673	4214	García Viñas, Juan Ignacio	2004-02-19 17:41:56.0	0:00	0	0	0,00		-1
□	15319	7191	Merino Cordoba, Salvador	2004-05-05 11:54:10.0	1:33	7	2	28,00		2
□	8663	5159	Odeh, Marwan	2003-06-11 10:57:38.0	2:05	1	0	0,00		0
□	8289	5159	Odeh, Marwan	2003-05-13 19:52:36.0	13:26	6	1	16,00		1
			Pimentel	2004-02-						

Figure 3. The student performance facility of the result analyzer

Related work

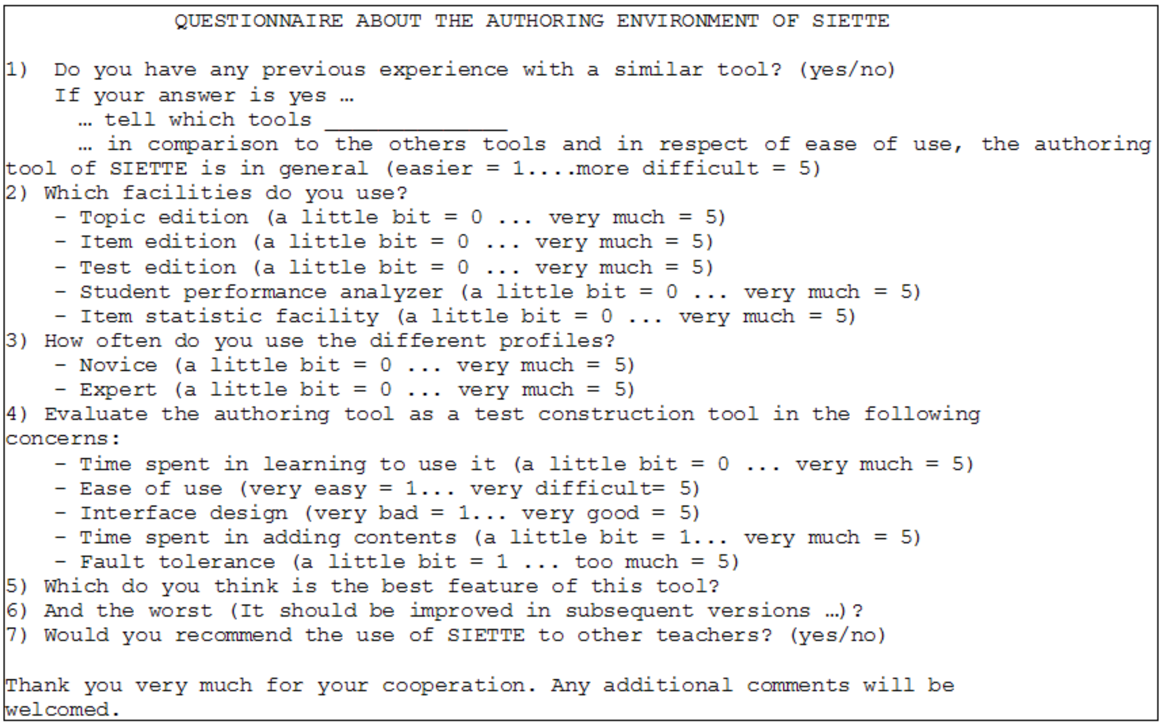
In adaptive testing there are no standards, nor are there any de facto standards. Most adaptive testing-based software tools make use of an eclectic approach to knowledge estimation (van der Linden & Pashley, 2001). Maybe this is the reason why there are not many adaptive testing-based tools. In the literature some systems can be found, like Hezinet (López-Cuadrado et al., 2002), a web-based learning system that includes an application for adaptive testing administration. It has an item pool of 600 items and uses the classical dichotomous 3PL model. INSPIRE (Papanikolaou et al., 2003) is also a web-based learning system that uses adaptive tests to assess the progress acquired during instruction. This system provides feedback to students while taking a test. It uses the 3PL model but keeps the discrimination factor constant for all items with a value of 2. More recently, other adaptive testing systems have appeared, such as CALEAP-Web (Gonçalves et al., 2004) and the work of Lilley et al. (2004).

In general, the former systems do not make rigorous use of adaptive testing theory. They implement an algorithm that combines adaptive testing with heuristics. These heuristics are used to make item calibration easy. They are based on the CBAT-2 algorithm (Collins et al., 1996). Through CBAT-2 the item parameters of the 3PL model can be easily computed. The discrimination and the guessing factors always take the same value, and the value of the difficulty is updated through a heuristic after any student takes the test. Certainly, this algorithm facilitates the use of adaptive testing, but does not use it properly. There are other works that apply adaptive tests, but using paradigms located further away from classical adaptive testing theory, like SKATE (Chua-Abdullah, 2003) or the work of Rudner (2002).

Finally, regarding the authoring tools used by these systems, references of these works do not, unfortunately, mention how items are added to the systems.

Evaluation

A study was conducted in order to evaluate the current authoring environment of SIETTE. For this study a questionnaire was prepared. Figure 4 shows the whole questionnaire. The final recipients were teachers with certain experience in the use of this tool. They teach undergraduate and postgraduate university courses, and they have used SIETTE as an assessment tool for their students. In general, the questions asked how often they used the different tools of the authoring tool, and to evaluate their satisfaction in terms of its features. The responses were defined in a Likert-type scale, i.e. in a scale from 0 to 5. The questionnaire was submitted to the users through an email.



QUESTIONNAIRE ABOUT THE AUTHORING ENVIRONMENT OF SIETTE

1) Do you have any previous experience with a similar tool? (yes/no)
If your answer is yes …
… tell which tools _______________
… in comparison to the others tools and in respect of ease of use, the authoring tool of SIETTE is in general (easier = 1....more difficult = 5)
2) Which facilities do you use?
- Topic edition (a little bit = 0 ... very much = 5)
- Item edition (a little bit = 0 ... very much = 5)
- Test edition (a little bit = 0 ... very much = 5)
- Student performance analyzer (a little bit = 0 ... very much = 5)
- Item statistic facility (a little bit = 0 ... very much = 5)
3) How often do you use the different profiles?
- Novice (a little bit = 0 ... very much = 5)
- Expert (a little bit = 0 ... very much = 5)
4) Evaluate the authoring tool as a test construction tool in the following concerns:
- Time spent in learning to use it (a little bit = 0 ... very much = 5)
- Ease of use (very easy = 1... very difficult= 5)
- Interface design (very bad = 1... very good = 5)
- Time spent in adding contents (a little bit = 1... very much = 5)
- Fault tolerance (a little bit = 1 ... too much = 5)
5) Which do you think is the best feature of this tool?
6) And the worst (It should be improved in subsequent versions …)?
7) Would you recommend the use of SIETTE to other teachers? (yes/no)

Thank you very much for your cooperation. Any additional comments will be welcomed.

Figure 4. Questionnaire submitted to the teachers

All users indicated they did not have any previous experience with a tool similar to this one. The part of this tool most frequently used is the item edition capability. 75% of the users pointed out that they use it very frequently (value 5 in the questionnaire). The other functionalities most commonly used are the test and topic capabilities respectively. Regarding the result analyzer tool, its two capabilities were used less than the previous ones. In any case, there was a significant difference between the percentage of teachers that use the student performance facility in comparison with the item statistic facility. The first facility is used with average frequency, whereas the second one is hardly ever used. The profile used most is the novice one. Most teachers stated that they used it very often. In contrast, there is a limited number of teachers that have never used the expert profile, and the rest use it with average frequency.

Concerning author satisfaction and tool effectiveness, the questionnaire included several issues that the teachers had to evaluate. Accordingly, teachers pointed out that the time spent in learning how to use the tool is satisfactory (values are located between 2 or 3). Regarding the tool's ease of use, the results are also satisfactory, but some teachers have suggested that the inclusion of contextual help within the fields of the component forms would be very useful. In addition, teachers indicated that the design of the interfaces is good. Some of them pointed out that it does not contain heavy components, as a result of which the load of the web pages is considerably rapid. On the other hand, one teacher suggested that the structure of the interface is sometimes confusing and that it should be improved by reorganizing the configuration parameters of items, tests and topics. The time spent in the construction of a subject was evaluated as average to good. Some teachers indicated that the greater the experience in using the tool, the lesser the time spent in the construction of a subject. The final issue evaluated was fault tolerance. All teachers fully agree that this feature is well managed by the authoring tool. A teacher pointed out that during the construction of a test he/she had several problems with his/her connection to the Internet, but these problems did not affect the content added, i.e. there was not a significant loss of information.

Finally, teachers were asked about the best and the worst feature of the tool. About the best feature, some of opinions were: *it provides web-based interfaces that do not require installing additional software, it permits preparing a test quickly, tests are immediately available in the virtual student classroom, it supplies a wide range of options to construct items and tests (e.g. different types of items, different test correction, item selection and finalization criteria, ...),* etc. Concerning the worst features the comments were: *the absence of a help manual makes it difficult to use certain options, the procedure to alternate between the two profiles is not very intuitive, it does not operate properly when accessing though Linux operating systems*, etc. The last question was if they would recommend this tool to other teachers. Most teachers answered affirmatively. Just one teacher pointed out that he/she uses the system for a limited set of students (in a masters course), but he/she does not dare recommend it for subjects with a large number of students.

Conclusions

SIETTE is a well-founded testing system that generates adaptive tests for grading or self-assessment. These tests have many advantages: tests are suited to students, the number of items required for assessment is lesser than in conventional testing procedures, estimations have high accuracy, etc. In SIETTE contents are structured on the basis of subjects. Each subject is composed of a set of topics, structured hierarchically using aggregation relations. Each topic has a set of associated items that can be used to assess it and all its preceding topics. Furthermore, SIETTE provides teachers with an authoring environment consisting of a set of tools that allow teachers to elicit knowledge, i.e. item, topic and test construction. It is a multi-user environment in which teachers can collaborate in the test creation process, although this collaboration can be restricted by applying different permissions to the elements of each subject.

SIETTE has, on the one hand, adaptive features: item selection, student assessment and test finalization criteria. These criteria are based on the performance of the students while taking the tests. On the other hand, through adaptable characteristics the test editor is personalized for each teacher's profile and permissions. For instance, the novice profile is very useful for teachers with no skills regarding adaptive test configuration. In addition, different item construction scaffoldings give teachers the possibility of easily adapting the test presentation to the group that is going to take a test.

The evaluation carried out shows that in general the authoring environment is well considered. In fact, some teachers of the School of Computer Science Engineering frequently use it as a complement to students' final qualifications. Currently, we are working on improving the features of the authoring environment in the direction pointed out by the teachers in the questionnaire. For instance, we are making a help manual, introducing changes in the interfaces to solve some former problems. Some filtering operations are being included in the student analyzer tool in order to allow a better study of data, and in addition its graphics are being improved.

Both the student classroom and the test editor are available for any user through the following URL: http://www.lcc.uma.es/SIETTE. In the student classroom, a subject "Demo", which includes several demo tests, has been defined. These tests have been created to show some of SIETTE's characteristics. Moreover, a "demo" teacher account has been created to freely access the test editor and the result analyzer, in order to show their operability and adaptable features.

Acknowledgement and Disclaimer

SIETTE is going to be used as part of the *LEActiveMath* project, funded under FP6 (Contr. Nº 507826). The content of any publication of any kind produced by the *LEActiveMath* consortium is the sole responsibility of its authors and in no way represents the views of the European Community and its services, who cannot be held responsible for any use that might be made of data appearing therein.

In addition, authors would like to thank the reviewers for their useful comments and suggestions about the former versions of this manuscript.

References

Brusilovsky, P., & Miller, P. (1999). Web-based Testing for Distance Education. In De Bra, P. & Leggett, J. (Eds.), *Proceedings of World Conference of the WWW and Internet*. Norfolk, USA: AACE, 149-154.

Chua-Abdullah, S. (2003). Student Modelling by Adaptive Testing - A Knowledge-based Approach. *Doctoral dissertation*, Canterbury, UK: University of Kent.

Collins, J. A., Greer, J. E., & Huang, S. X. (1996). Adaptive assessment using granularity hierarchies and bayesian nets. In Frasson, C., Gauthier, G. & Lesgold, A. (Eds.), *Proceedings of the 3rd International Conference on Intelligent Tutoring Systems,* New York: Springer Verlag, 569-577.

Conejo, R., Guzmán, E., Millán, E., Trella, M., Pérez-de-la-Cruz, J. L., & Ríos, A. (2004). SIETTE: A Web-Based Tool for Adaptive Teaching. *Internation Journal of Artificial Intelligence in Education, 14*, 29-61.

Glas, C. A. W. (2000). Item calibration and parameter drift. In Van der Linden, W. J. & Glas, C. A. W. (Eds.), *Computerized Adaptive Testing: Theory and Practice*. Norwell, MA: Kluwer Academic Publisher, 183-199.

Gonçalves, J. P., Aluísio, S. M., de Oliveira, L. H. M., & Oliveira, O. N. (2004). A Learning Environment for English for Academic Purposes Based on Adaptive Tests and Task-Based Systems. *Lecture Notes in Computer Science, 3220*, 1-11.

Guzmán, E., & Conejo, R. (2002a). An adaptive assessment tool integrable into Internet-based learning systems. *Paper presented at the International Conference on Information and Communication Technologies in Education*, November 13-16, 2002, Badajoz, Spain.

Guzmán, E., & Conejo, R. (2002b). Simultaneous evaluation of multiple topics in SIETTE. *Lecture Notes in Computer Science, 2363*, 739-748.

Guzmán, E., & Conejo, R. (2004a). A library of templates for exercise construction in an adaptive assessment system. *Technology, Instruction, Cognition and Learning, 1*, 303-321.

Guzmán, E., & Conejo, R. (2004b). A Model for Student Knowledge Diagnosis Through Adaptive Testing. *Lecture Notes in Computer Science, 3220*, 12-21.

Hambleton, R. K., Swaminathan, J., & Rogers, H. J. (1991). *Fundamentals of Item Response Theory*, California, USA: Sage.

Kay, J. (2000). Stereotypes, Student Models and Scrutability. *Lecture Notes in Computer Science, 1839*, 19-30.

Lilley, M., Barker T., & Britton, C. (2004). The development and evaluation of a software prototype for computer-adaptive testing. *Computers & Education, 43*, 109-123.

López-Cuadrado, J., Pérez, T. A., Arrubarrena, R. M., Vadillo, J. A., & Gutiérrez, J. (2002). Generation of computerized adaptive tests in an adaptive hypermedia system. *Paper presented at the International Conference on Information and Communication Technologies in Education*, November 13-16, 2002, Badajoz, Spain.

McCalla, G. I., & Greer, J. E. (1994). Granularity-Based Reasoning and Belief Revision in Student Models. In Greer, J. E. & McCalla, G. (Eds.), *Student Modeling: The Key to Individualized Knowledge-Based Instruction*, Berlin Heidelberg: Springer Verlag, 39-62.

Oppermann, R., Rashev, R., & Kinshuk (1997). Adaptability and Adaptivity in Learning Systems. In Behrooz, A. (Ed.), *Knowledge Transfer*, London, UK: pAce.

Owen, R. J. (1975). A Bayesian Sequential Procedure for Quantal Response in the Context of Adaptive Mental Testing. *Journal of the American Statistical Association, 70*, 351-371.

Papanikolaou, K. A., Grigoriadou, M., Kornikolakis, H., & Magoulas, G. D. (2003). Personalizing the interaction in a web-based educational hypermedia system: the case of inspire. *User Modeling and User-Adapted Interaction, 13*, 213-267.

Rudner, L. M. (2002). An examination of decision-theory adaptive testing procedures. Paper presented at the *Annual meeting of the American Educational Research Association*, April 1-5, 2002,, New Orleans, LA, USA.

van der Linden, W. J., & Glas, C. A. W. (2000). *Computerized Adaptive Testing: Theory and Practice*, Netherlands: Kluwer Academic Publishers.

van der Linden, W. J., & Pashley, P. J. (2001). Item selection and ability estimation in adaptive testing. In van der Linden, W. J. & Glas, C. A. W. (Eds.), *Computerized Adaptive Testing: Theory and Practice*, Norwell, MA: Kluwer Academic Publisher, 1-26.

Vassileva, J. (1997). Dynamic Course Generation on the WWW. In du Bolay, B. & Mizoguchi, R. (Eds.), *Proceedings of the 8th World Conference on Artificial Intelligence in Education*, Amsterdam: IOS Press, 498-505.

Weber, G., & Brusilovsky, P. (2001). ELM-ART: An Adaptive Versatile System for Web-based Instruction. *International Journal of Artificial Intelligence in Education, 12*, 351-383.