

Chapter 8

E-Assessment Systems and Online Learning with Adaptive Testing

Marjan Gusev and Goce Armenski

Abstract In this paper we explain how a systematic approach to design Assessment as a Service on a cloud with SOA architecture can exploit add on functionalities like online learning with interactive assessment as a type of formative and integrative assessment for the learning process. The idea is realized by a system (computer based program) that systematically asks questions and leads the students towards knowledge construction and discovery. The system should be as simple as possible and intelligent enough to enable the realization of this idea. This online learning tool with adaptive testing uses software agents, where several strategies define the agent's behavior. We observed that the strategy "3 correct answers in a row" performs the best.

Keywords E-testing · E-learning · Computer based testing

8.1 Introduction

There are a lot of e-Learning software products, mainly realized as Learning Management Systems (LMS) that do not support e-Assessment and e-Testing with full coverage of features [14, 43]. Some significant examples include commercial products Blackboard [6] and WebCT [50], or open source software products, like Moodle [32], Fle3 Future Learning Environment [15], The Manhattan Virtual Classroom [10], LRN [30] etc.

M. Gusev (✉) · G. Armenski
Faculty of Information Sciences and Computer Engineering, Ss. Cyril and Methodius
University, Rugjer Boshkovikj 16, 1000 Skopje, Macedonia
e-mail: marjan.gushev@finki.ukim.mk

G. Armenski
e-mail: goce.armenski@finki.ukim.mk

On the other hand, there are stand-alone products and designs of e-Assessment and e-Testing that have a sophisticated set of functionalities, but lack integration and interoperability with existing products. Examples include sophisticated testing centers for professional certificates, like Prometric [38] or Pearson Vue [37] and plenty of solutions that offer different types of electronic testing, questionnaires, and surveys on Internet, such as eSurveyPro [17], eSurvey Creator [16], Free Online Surveys [21], etc.

The existing LMS systems and specialized e-Testing systems are not designed to be interoperable or to act as cloud solutions and deliver Assessment as a Service, i.e. be capable to exchange tests and related data elements with other LMS systems, and therefore they can not support each other or even create new forms of efficient learning. For example, Gierlowski and Nowicki claim that vast majority of currently available electronic knowledge assessment tools are not only extremely similar but also offer strictly limited functionality [22].

In this paper we address the research problem of how e-Testing can be used efficiently in e-Learning, particularly in the construction of online adaptive learning environment and the realization of similar pedagogical methods.

Clark and Meyer define 3 essential e-Learning types as information acquisition, response strengthening and knowledge construction, using inform goals, perform-procedure goals and perform-principle goals correspondingly [9]. Aristotle and Socrates used the best method to educate the students by continuously asking them questions to stimulate critical thinking, and to illuminate ideas [5, 11]. We map these methods into knowledge construction by knowledge discovery. Our goal is to define a learning system, where the students provide answers and come to understand that they have learned the relevant knowledge by analyzing and constructing answers. The philosophy behind this idea is to select a strategy how to choose (and ask) questions and lead the students towards knowledge discovery. The final aim is to realize a computer based program which will lead the students towards knowledge discovery and construction. The system should be as simple as possible and intelligent enough to enable the realization of this idea. We call this system **online learning with adaptive testing**, while the standard e-Testing system is mainly used to assess the student knowledge as **click test**.

The online learning with adaptive testing is a system that evaluates a different test, each student every time he/she applies for. This is realized by defining a strategy for the exam and building a database with questions, which is allowed for self testing, learning, and conducting exams. The realized electronic system is based on sophisticated web technology with appropriate service oriented architecture capable to be hosted on cloud.

To realize the overall idea we also address autonomous software agents. They act as intelligent agents in the role of instructors capable to modify the way the students learn. The goal of the online learning system with adaptive testing is to set the system to act in the role of an instructor and ask questions which lead the students from one knowledge item to another in the process of knowledge construction. There are several strategies to define the agent's behavior and our research proved that the **3 in a row** strategy performs the best. The experiments

realized at the end of the course showed that by using this system the students learn the relevant knowledge items faster and more easily.

The paper is organized as follows. [Section 8.2](#) presents the state-of-the-art on e-Assessment system, its architecture and organization as cloud solution, the interoperability aspects and the knowledge database organization; then algorithms and procedures covering test question types, test delivery models, test creation algorithm and grading. [Section 8.3](#) presents the online learning system with adaptive testing by analyzing the interactive response learning system, navigation algorithm and decision making strategy. This section also contains description of related work and coverage of software agents. The final [Sect. 8.4](#) is devoted to conclusion and future work.

8.2 State of the Art

In this section we give overview of state-of-the-art on e-Assessment systems with correlation to the online learning with adaptive testing.

8.2.1 Computer Based Testing

Luecht and Sireci present a brief history of Computer Based Testing (CBT) [31], concluding that there is no single CBT model ideal for all educational tests, rather, all models have their strengths and weaknesses, and some are better suited to the characteristics of a particular testing program than others.

Each electronic test consists of questions that represent appropriate knowledge item (or learning objective). Seven dimensions (structure, response action, media inclusion, interactivity, complexity, fidelity, scoring method) are identified by Parshall et al. as important for e-Assessments, and a corresponding taxonomy is created [35]. Scalise and Gifford define a taxonomy of 28 innovative item types that may be useful in computer-based assessment [41]. These taxonomies address question types in e-Assessment systems.

Crisp presents a taxonomy of e-Assessment systems based on the level of constraint in the item/task response format, while analyzing the interactive assessment [12]. He observed the interaction among the teacher (instructor) and the student, and defined its goal to be “assessment **for learning**” or “assessment **of learning**”. Sclater and Howie identify credit bearing tests, self assessment and diagnostic tests according to their purpose and possible use in interactive assessment [42].

Table 8.1 presents the correlation of our definitions to these categorizations.

There has been intensive research and development in the field of e-Learning systems. For example, Davies and Davis in [14] present the results of EU funded projects in using grid infrastructures for e-Learning and technologies for online

Table 8.1 Correlation of different taxonomies to online learning with adaptive testing and click tests

Source	Online learning with adaptive testing	Click test
Purpose in our definition	Adaptive learning	Assessment
Category by Sclater and Howie [42]	Self assessment and diagnostic testing	Credit bearing testing
Category by Crisp [12]	Formative assessment	Diagnostic and summative tests

interoperable assessment. Dagger et al. define various characteristics of modern LMS using adaptive hypermedia and semantic web technologies based on service oriented architecture [13].

8.2.2 Architecture and Design

In this section we give an overview of a modern e-Assessment system, its architecture consisting of a set of services used in an e-Learning context and collectively realize required business objective. We also address interoperability as a very important feature of this system.

8.2.2.1 Modern E-Assessment System Architecture

There are a lot of papers and projects describing modular e-Assessment architecture. For example, Gierlowsky designed a highly scalable and modular architecture with several layers and modules [22]. Armenski and Gusev designed a three-layered architecture to capture most of the demands of a modern e-Assessment system [2] as presented in Fig. 8.1. This architecture is intended to be used by any

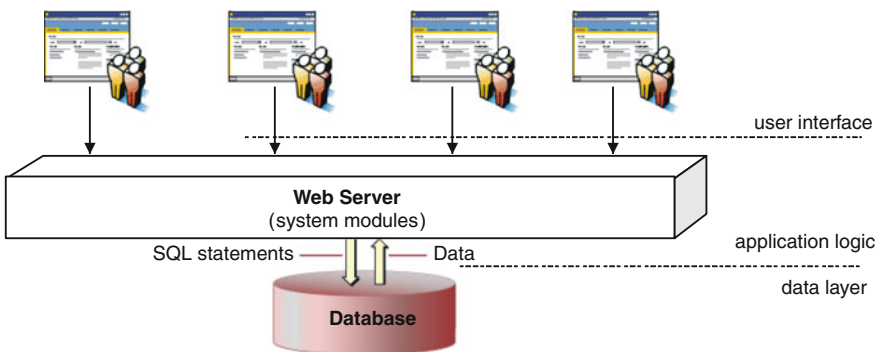


Fig. 8.1 Three layered architecture of the system [2]

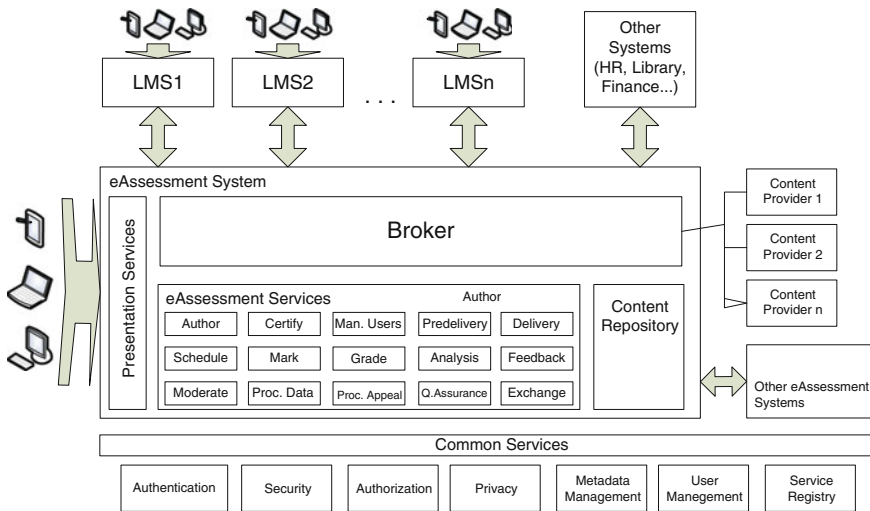


Fig. 8.2 Architecture of an ultimate e-assessment system [3]

computer on Internet through common web browser user interface, and separates the database layer from the application layer for basic system modules and also from the user interface layer.

Furthermore, they have presented a novel architecture based on SOA to support all relevant functions in computer based assessment [1] and defined an ultimate e-Assessment system with more advanced approach trying to define the ultimate assessment engine, the architectural style behind, and its overall architecture [3], as presented in Fig. 8.2. Although this model allows construction of a stand-alone system, it offers a possibility to construct a Software as a Service cloud solution, by defining a broker module that establishes communication to different LMS systems using various standards and acts the role of system service orchestrator.

Recently, Ristov et al. defined a sophisticated cloud solution on top of these ideas [39], by defining three subsystems, i.e. the Management, Assessment and Reporting subsystems, as presented in Fig. 8.3 and an organization-scheduling algorithm for different Virtual Machines (VMs), instead of one broker. The Admin, Student and Reporting agents serve to enable communication among different virtual machines, and communication between agent and administrator, teacher and student correspondingly. A special agent module, called Infrastructure agent is responsible for resource provision, i.e. activates or cuts provision of a VM by analyzing the workload.

8.2.2.2 Interoperability

Regarding the interoperability of e-Learning systems, Dagger et al. differentiate three generations [13]. However, interoperability has not been addressed thoroughly, although several detailed frameworks support standardization efforts in the

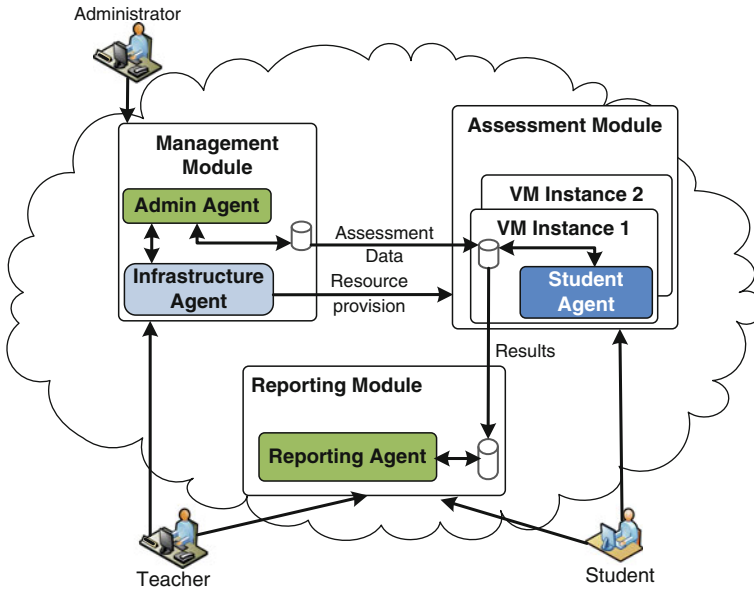


Fig. 8.3 E-assessment system organization [39]

e-Learning domain, like JISC e-Learning Technical Framework (ELF) [19], IMS Abstract Framework (IAF) [24], Open Knowledge Initiative (OKI) [33], LeAPP Learning Architecture Project [29] etc. Another example is the project [45] with goal to establish appropriate infrastructure to develop competences, addressing an assessment model, without supporting services [44].

Although service oriented architecture is mostly implemented trend in the last years, still there are no commercial e-Assessment systems on the market following the real system architecture based on SOA, which is ready for interoperability challenges for the growing demands in the world. Mainly this is due to the lack of standards and a lot of industry pushed solutions which in essence do not like to be interoperable.

In the beginning, the organizations formed consortiums to define interoperable standards, and while all industry players were implementing them, there was an on-going debate about cloud computing and a possibility for interoperable services on top of these solutions. Therefore still interoperability of e-Education systems is a hot research topic with a lot of research on standards and work on the development of new solutions is underway. The results of all these efforts will definitely have impact and influence on all new software solutions.

The capability of a given software system to use the same formats to store and retrieve information means it uses **interoperable** formats. In addition **interoperability** of services means that two different software systems realized on various hardware and platforms can interchange information or services [24]. Dagger et al. [13] discusses two levels of interoperability between LMS and its

tools: interoperability of content and interoperability of tools. Vossen and Westercamp [49] identified one more level of interoperability in exchanging user data. Several published standards like SCORM [28], IMS Content Packaging [25] and IMS Learning Design [26] are evidence of intensive research about content interoperability recently.

The presented model of e-Assessment system [3] follows the trend to separate the content from tools, making possible a seamless and dynamic exchange of tools, functionalities, semantics and control. The specified system is built with Service Oriented Architecture, based on encapsulating existing business functions as loosely coupled, reusable, platform-independent services which collectively realize the required business objective. The final goal is to enable a system that uses widely adopted standards, increases system flexibility and supports a lot of pedagogical diversities.

Another innovation that the model in [3] puts an accent on is the concept of pluggability. In a real system, this means that the interoperability is already established and the system architecture can be pluggable to other systems both in educational and business environments. An e-Assessment system can be pluggable if it has ability to be easily attached to any existing LMS and its functionalities to be used as if that e-Assessment system was part of the LMS from its basic installation.

Cloud computing offered a completely new perspective on the development of solutions. For example, if an e-Assessment system is built such that it is pluggable to any LMS, it should not only offer a complete service, but it should also offer subsets and various sub services, like realization of questionnaires, inquiries, public opinion gathering service etc. To make this possible, the system should use a highly interoperable service oriented architecture, building modular services that can exchange interoperable information. For example, a company that uses a structured set of customers and would need to gather customers' opinion through a questionnaire. The company probably would like to exchange the set of customers within the accounting software service, send e-mails via e-mail marketing service and use e-Testing software service for questionnaires. All these services should be interoperable and exchange required information.

A modern LMS should support the exchange of learning resources and profiles with other systems including the legacy systems. It will also support an extensive usage of e-Testing for learning process, rather than just for assessment. The exchange includes knowledge items, hypermedia content and personalized learning environments. Finally, it will benefit with increased efficiency and effectiveness. For example, Thurlow et al. give benefit overview [48].

8.2.3 Algorithms and Procedures

This section gives an overview of used algorithms and procedures for the e-Testing process in an e-Assessment system. It includes organization of knowledge items and questions, test delivery methods and test creation algorithms.

The knowledge items are stored in a knowledge database, organized usually as a tree, where each lecture consists of parts, each part, of sets, each set of learning objectives, etc. The final leafs are questions that correspond to a given knowledge item. In addition to basic information provided by the question, an extensive information can be stored for each question in the knowledge database including statistics of realized testings and answers given.

Tests are created according to test creation algorithms, which form tests by selecting knowledge items from the knowledge database and then the e-Testing software continues to realize the testing by presentation of questions and collecting the student answers via test delivery models. Patelis gives an overview of various test delivery models [36] identifying linear tests, dynamic linear, testlets, mastery models, and adaptive tests. Thompson and Wiess identify three primary variable-form approaches: computerized adaptive testing (CAT) and linear-on-the-fly testing (LOFT), and multistage testing [47]. Luecht and Sirreci differ eight CBT models primarily with respect to their use of adaptive algorithms, the size of the test administration units, and the nature and extent to which automated test assembly is used [31].

8.2.3.1 Test Creation Algorithm

Test creation algorithm is closely connected to the chosen method for test delivery. The idea for creating different tests for every student, forced us to apply the model for dynamic test creation. With that idea every student will get different test, with same weight like all other students. These dynamically created tests will have a fixed number of questions because this was first time system for automated assessment to be applied at our University. In order to provide a less painful change in the way of taking the assessment and to lower the difficulties in its adaptation we have decided that fixed number of question is a better solution than dynamic one. The same reason forced us to use dynamic test creation model instead of model for adaptive testing because of the easiness and the transparency that non adaptable test have. The applied model gives an opportunity for students to list the questions one by one, and answer only those whose answer they know.

The strategy for test generation is defined by the course administrator when he schedules the assessment. The course tree structure is used to set a strategy. The administrator is marking the learning objects from which questions will be selected, specifying the number of questions taken from each learning object. So, the course administrator will have control over the curricula. Since each learning object has questions with same weight, the tests which will be generated will have same weight too, but the students will get different tests from those learning objects selected by the course administrator. The system has a feature with which already made strategies are saved and can be used in the future.

Test creation algorithm may be rather complex for adaptive testing and variable-form testing with algorithmic approach, where the test is designed to be administered with a dynamic, interactive algorithm, in contrast to multistage

testing, where there are fixed routes between the testlets [47]. For example, AS method uses an adaptive algorithm that maximizes the test information function for each examinee. This approach leads to overexposure of a relatively small portion of the entire test bank since the most informative items are continually in high demand. AT algorithm is based on the idea that after completing the testlet, the computer scores the items within it and then chooses the next testlet to be administered. Thus, this type of test is adaptive at the testlet level rather than at the item level.

8.2.3.2 Grading

Automated scores are consistent with the scores from expert human graders, they are fair and have been validated against external measures in the same way as is done with human scoring, according to [51].

Grading as a process starts when the testing is finished, i.e. when the student submits the final answers to the system or when the time limit is exceeded. It is a process of evaluation of the submitted answers by matching the answers against correct answers stored in the database.

The evaluation process in case of fixed response questions is realized as a straight forward procedure of checking if the answer is correct. There is only one constraint since the test allows rotation of possible answers, so the check is performed against real correct answers, instead of a given schema.

A special procedure starts to evaluate the answer in the case of answers where additional computation is required, or in case of essay questions. Often these procedures are realized with human interventions and decisions.

In our system, final results are stored in a database and sent to the display system. The system displays the final results in two ways, either using points or percents. There is an option to see the right answers compared to those entered by the user. In such a way the student realizes where the mistake is and finds the correct answer. This is a method of assisted learning, usually used by teachers with corrections whenever a mistake is noticed.

A lot of efforts were made to enable security of the system. One problem that arises often is the method of guessing. The students do not try to answer a given question, instead, they are trying to guess the answer by random clicking on a possible answer. In a long term education process, as a kind of repetition method, the student will, hopefully, learn the corresponding knowledge item. However, in order to eliminate guessing, we have implemented negative marking. This is a technique where each mistake is punished by achieving negative grade. Applying this procedure the students now avoid guessing and answer the questions only if they are certain about the correct answer.

8.3 Description of the Online Learning with Adaptive Testing

Here we will explain the details of the online learning system realized with adaptive testing. In this system, there is a high interaction between the teacher and students, integrating the e-Testing solution into an efficient learning process. The basic idea is explained in [Sect. 8.3.1](#) as interactive response learning system, while the navigation through the lecture and knowledge items in [Sect. 8.3.2](#) and the decision strategy algorithm in [Sect. 8.3.3](#).

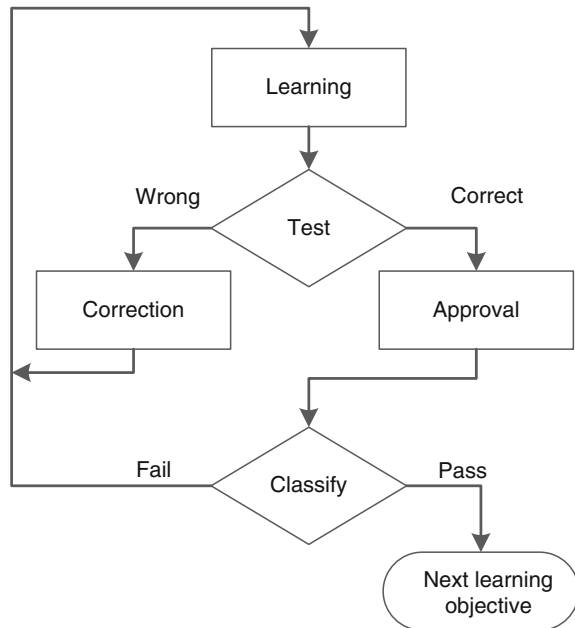
8.3.1 Interactive Response Learning System

E-education refers to activities connected to the process of increasing the students's knowledge and skills. E-assessment is not just referring to the process of evaluation of students knowledge, but it can be efficiently used in e-Education. New services to be developed in a ultimate modern e-Assessment system should also have ability to integrate and embed in the learning context or in the process when the student increases the knowledge. It usually requires a higher level of student involvement in the assessment and learning process. It leads to an idea about a complex learning process where teaching, learning and assessment interact frequently to increase the efficiency and overall impact. To realize this new learning approach, the e-Assessment services should be highly integrated in e-Education processes and be capable to address the complex challenges and characteristics. The system proposed in [23] is identified as interactive response learning system or just online learning tool.

The basic idea is based on a scenario presented in [Fig. 8.4](#). A student chooses to attend a course lectures online, makes a subscription and creates a record for his activities. Then, the student can navigate the course tree structure and starts with the online learning tool. Learning materials are presented for each lecture as part of the LMS and than test generation starts to evaluate the students's knowledge about particular learning objectives, knowledge items or skills.

This scenario is typical for each student. In this case, the impact of the lecturer is built in the system. The lecturer usually provides a corrective measure, i.e. corrects the students if there is a mistake or approves correct answers. This is exactly what the online learning tool with adaptive testing does. However, there is a strategy when the teacher will continue to the next learning objective or explanation of a new knowledge item. Usual behavior of a good teacher is to move to the next level only if the student has learned the previous learning objectives and knows all relevant knowledge items in order to be capable to follow the next item. This has to be done in a way that the student does not loose concentration or gets no frustration if not all questions are answered correctly, but has sufficient understanding about the corresponding learning objective. This is why we have to

Fig. 8.4 Online learning scenario with adaptive testing



implement adaptive testing which adapts to the current knowledge of the student, a situation that will simulate the teachers way of thinking.

The realization of this scenario uses several algorithms for adaptive testing and usage of software agents. The navigation through the learning objectives should follow a predefined path, but also be capable to enable different scenarios to reach the final goal, for example, better students with good background would like to choose a shorter path and avoid variations or similar questions, and the students that are not motivated and with no background should probably follow the longer and exhaustive path. All these algorithms will be discussed in the next sections.

8.3.2 Navigation Algorithm

The navigation through the learning objectives is possible for the already explained knowledge database with tree-like structure. Lectures are branches and knowledge items are leafs in the tree. Each lecture consists of smaller parts and each part consists of different sets and finally of learning objectives. The constraints used in this tree-like knowledge database structure are presented in Table 8.2.

The navigation can start in the leftmost nodes in the tree. The questions are selected out of possible candidates in the given set/learning objective. The testing process is realized by asking questions one by one in sequential order. There is a strategy which decides whether the provided answers are sufficient to show that the

Table 8.2 Constraints in the knowledge database

Item	Constraint
Course material	Has at least 3 parts
Part	Has at least 3 knowledge items
Knowledge item	Has at least 4 question sets
Question set	Has at least 5 questions
Question	At least one is hidden for final exam

student has learned the corresponding learning objective. This strategy is explained in the next section and here we will present how the navigation can continue.

Successful finish of a given learning objective allows the student to move to the next learning objective. This is done by preorder traversal in the tree i.e. L-R-O, meaning that the student finishes the leafs from left to right and then moves to upper level or to another branch in the tree-like structure. Moving to the next level is once more accompanied by a summary test that selects at least one question of all nodes on the corresponding level. It means that all the sets from a given part have to be passed before the final test is generated for a given part (knowledge item). The next part is traversed until a complete fulfillment of lecture. Once more, a test is generated by selecting at least one question from each previous node.

The effect of this navigation through the system is interesting for the students. Since the test generation algorithm randomly chooses a question from a given knowledge item and randomly presents possible answers, the students have a feeling that the system is different for all of them. In a sense, this also happens with a real teacher. Another issue is the navigation path, since there are more than 3 nodes on same level, the movement to next node is also random, so each student, in effect, has different navigation path, i.e. learning path.

Once the student passes all knowledge items on the same level, the system sets a test for the given part, now randomly selecting a question from each node. Here the system might ask the very same question, which is allowed following the strategy that repeating enables better learning. Although the first impression of the students is that the system asks the same questions and they appear randomly, in the background, the system is navigating through the knowledge database tree and acts as a teacher who is always asking questions, corrects the wrong ones and awards the correct answers by letting the students to upper levels.

All navigation paths are stored in the system and also all statistics about given answers. The lecturer knows which students are subscribed and can view their records and analyze their achievements.

8.3.3 Decision Strategy

A decision making strategy to evaluate if the student has learned a certain learning objective or knowledge item is very important. There are several issues that have to be analyzed before developing such a strategy.

If one question is set from a given knowledge item (we assume there are at least 3 questions for a given learning objective), we may expect guessing as a method of answering. In this case we are not sure if the student really learned the corresponding knowledge item or not.

If all questions are asked, then we will be sure that the student has answered all of them, but this leads to exhaustive navigation through all questions. In this case, especially the better students will be bored. So the only alternative is to find an appropriate strategy that will decide when the student has learned the corresponding knowledge item.

Another issue is raised for a situation when a wrong answer is given by the student. The system gives the result after each answer. This works as a corrective measure providing valuable feedback, so the student will now know the correct answer to a given question. However, this does not mean that the student has learned the corresponding knowledge item, just that the correct answer was shown to the student and the method of recognition might be used instead of presentation of real knowledge.

The only alternative to realize this adaptive strategy is based on correctly answered questions and their order. For this purpose we have set an experiment and tested which strategy the students think is the best in the learning process. The experiment was to test the decision strategies presented in Table 8.3.

The decision strategy enables the online learning system with adaptive testing to decide when to jump from one to another learning objective. The strategies AC and 1C are easily understood, since they refer to correctly answering of all or one question. The strategies 3C and 3R assume that the student should answer three questions correctly. The difference is in how these correct answers are obtained. 3C just counts correctly answered questions in a given learning objective and makes positive decision without checking if the student has already answered wrongly some questions, while 3R counts in-a-row, so if three questions are answered correctly consequently (in-a-row) then the decision is made. For example, let the learning objective have 10 questions and the obtained answers are wrong, two correct, wrong and three correct or WCCWCCC, where W means wrong and C correct answer, then the 1C strategy makes decision after the second question, 3C after the fifth question and 3R after the seventh question. Our experiments showed that 1C and 3C strategies are vulnerable to guessing, while 3R is less.

Table 8.3 Decision strategies to move into another learning objective

ID	Strategy	Description
AC	All correct answers	The student should answer correctly all questions for a given learning objective
NC	N correct answers	The student should answer correctly N questions for a learning objective, although some questions might be not answered correctly
NR	N correct in-a-row	The student should answer correctly N questions for a learning objective consequently (one by one in-a-row)

The best alternative 3R strategy is realized by the following algorithm. If the student answer is not correct, the counter for correct answers in a row is reset. Then the questions are repeatedly selected randomly within a given knowledge item. The decision is made only after three consecutive correct answers.

This adaptive strategy concerns not only correct answers in a row, but also timing constraints and level of difficulty. After easy questions are set, the difficult ones follow in an adaptive manner according to series of correct answers.

In our system we implemented the adaptive strategy, which records the student achievement. If the student is knowledgeable and answers correctly in three consecutive parts, then the strategy is adapted to “2 correct answers in a row” and in that sense it moves to “one correct answer in a row” meaning one question per learning objective. We believe that this strategy is suited to better students and will eliminate the syndrome that the testing process is boring. It will still be motivating for the students.

There is a record of the efficiency of the tests the student takes and of the navigation paths. This data can be presented to the teacher and also to the student.

The goal of e-Assessment should be specified explicitly. If the final aim is to assess student knowledge and skills then we should use a strategy that classifies the grades according to predefined criteria, as it is done by ETS [18]. However, the goal of the online learning system with adaptive testing is to provide a proof that the student has passed over all learning objectives showing correct answers for relevant knowledge items. It does not grade or classify the grades, nor evaluates the student efficiency. It is intended as a support tool for the students to learn the corresponding learning objectives.

8.3.4 Software Agents

There are a lot of definitions about software agents as computer programs that act for a user with authority to decide. Russel and Norvig define intelligent agents as an abstract functional system similar to a computer program. According to their definition [40], intelligent agent is an autonomous entity which observes through sensors and acts upon an environment using actuators and directs its activity towards achieving goals (i.e. it is rational), as presented in Fig. 8.5. The online learning tool with adaptive testing can be classified as intelligent agent since the agent (online learning tool with adaptive testing) takes action of the environment (student’s knowledge) by perception (assessment) of current student’s knowledge and skills. The sensors (e-Testing system) send information to the core where the actions and decisions are made (whether the student has learned the corresponding learning objective) and actuators start actions (the student can move to next level—learning objective).

Another definition in [20] defines that an autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future.

Fig. 8.5 Simple reflex intelligent agent [40]

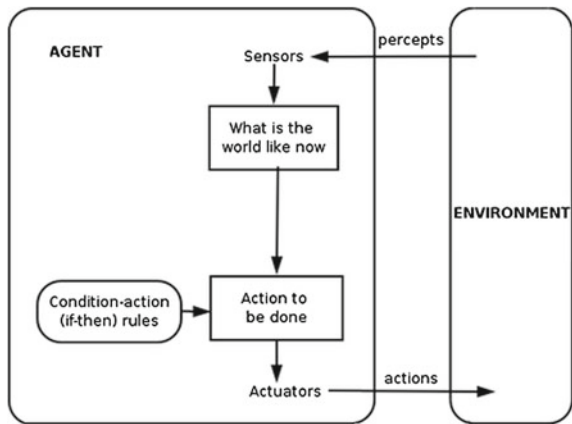


Table 8.4 Autonomous software agent properties that our system has

Property	Description
Reactive	Responds to learned knowledge items
Autonomous	Controls the process of learning (moving from one learning objective to another)
Continuous	Runs as continuous process over time
Communicative	Interacts with students
Adaptive	Changes its behavior based on current state

According to the definition given in [20], our system acts as an autonomous software agent, since the environment is the students knowledge, sensing correlates to assessment, and action that produces state in environment is learning. Drives are built-in preferences and act as primitive motivators. The predefined goal in the online learning tool with adaptive testing is to have approval that the student has visited all learning objectives and has proved to have initial knowledge about the knowledge items. The goal is not to have a precise assessment and classify the student’s knowledge and skills according to a given schema, as it is in the regular assessment software packages. The agent’s action-selection mechanism is based on the described decision strategy and navigation algorithm, allowing the student to move to the next level of learning objectives whenever the predefined criteria is satisfied.

The described online learning system with adaptive testing has several agent’s properties as presented in Table 8.4.

A good overview of definitions about software agents, their classification and architectures is given by Bădică et al. [4]. The authors give a comprehensive list of agent features, although in Table 8.4 we refer only to specific properties analyzed by our system.

8.3.5 Related Work About Adaptive Testing

Clark and Meyer define asynchronous e-Learning with feature to dynamically tailor instruction to the changing needs of learners. This feature is realized by an adaptive control with a program that dynamically adjusts lesson difficulty and support based on the evaluation of learner's response [9]. They provide an extensive evidence and references for dynamic adaptive control.

Adaptive learning using web has been analyzed a lot in the literature. Brusilovsky gives an overview of adaptive and intelligent technologies for web-based education and identify five major technologies with immediate roots in Adaptive Hypermedia and Intelligent Tutoring Systems [7, 8]. According to their classification our definition of the online learning system with adaptive testing belongs to adaptive navigation support (assisting in hyperspace orientation), curriculum sequencing (finding optimal path through the learning material) and intelligent solution analysis (using intelligent classifiers with extensive error feedback).

Paramythis and Loidl-Reisinger give overview of Adaptive Learning Environments (ALE) and e-Learning standards [34]. They define 4 categories of adaptation in learning environments and our definition of online learning with adaptive testing belongs to content discovery and assembly, where the adaptive component lies in monitoring of student's knowledge.

In this paper we address adaptive testing, where a specific algorithm will adapt the test to the knowledge and skills of each student, not just with respect to selected items by the tester (teacher). According to [46] the basic CAT method is an iterative algorithm that searches for the optimal question, based on the current estimate of the student's score, and after answering correctly or incorrectly, the score is updated.

So, the method is to choose the optimal question based on current score, and dynamically to update the score. There are different strategies to choose the optimal question, but in our model of online learning with adaptive testing we refer to two step procedure, first we would like to be sure that the student has learned the corresponding learning objective by not asking all questions, and navigating the knowledge tree as method where to move next.

The final goal in adaptive testing is also very important. Most of the existing strategies for adaptive testing aim to classify the student's answers and make precise evaluation of his/her knowledge. Adaptive testing in this case should always adapt to the knowledge level the student shows and the system should ask questions from upper and lower levels in order to make a final decision about the level of knowledge. For example, up-down methods are used in most of the existing testing software products [18] and there are a lot of variants, such as, 1 up 2 down, 1 up 3 down, etc. Kaernbach introduces simple adaptive testing with the weighted up-down method, where each correct answer leads to the higher level and each wrong answer leads to lower level [27]. There are a lot of examples of implementation of this strategy and the final score is obtained by weighted sum of number of questions answered in appropriate level.

However, our online learning tool with adaptive testing aims to a different goal. The previous strategy corresponds to upgrade of IC strategy which now will return backwards one knowledge item if wrong answer is obtained. Since the goal is not the evaluation of score and grading the student, but verifying if corresponding learning objective is learned by a student, then this strategy will not give better results than those we have experimentally shown.

Online learning with adaptive testing acts as software agent that represents the teacher who sets the lecture and approves all leaves and branches in the tree structure for the given lecture. This schedules the possible traversal path and lists all learning objectives that the student is expected to learn within the given lecture.

CAT provides an accurate point estimation of individual ability or achievement. Another approach is computerized classification testing (CCT) usually used to classify students according to their knowledge. Typical examples include pass/fail or basic/proficient/advanced levels. In online learning model with adaptive testing we use only pass/fail assessment by applying the decision strategy.

A properly designed and implemented CAT can affect the motivation of students [47]. Because lower-ability students will receive easier items, they will become less discouraged and stressed. Conversely, high-ability students will not be wasting time on items that are far too easy; they will receive items that appropriately challenge their high ability. This is also built in our online learning with adaptive testing which gracefully improves the strategy for better students.

8.4 Conclusion and Future Work

Initially, the design of the e-Testing system started in 1999 to help the realization of frequent assessments (each month) where more than 500 students take part. The original idea to create a system that can help the realization of exams for large number of students was expanded to realize an independent system for electronic testing. New features added values to realize not just an assessment tool, but also to support the learning process and overall education. Later on, the development of software as a service was a great challenge, especially in solving the interoperability issues.

The presented e-Testing system is implemented and in use from 2001 at the University Ss Cyril and Methodius. Added-values, such as, online learning tool with adaptive testing and self testing were installed in 2002, and a new version was released in 2003. A more sophisticated newer version was installed in 2006, completely changed using the service oriented architecture and in 2009 the system was upgraded to the cloud computing model. In the process of developing the assessment system and introducing the online learning system with adaptive testing we have faced and successfully solved a lot of problems, especially those that arise due to cheating. We have tested several strategies arriving at the best. The design of an interoperable cloud solution will enable this tool to be used as add-on in the existing LMS.

During the last 10 years 9,132 students and 110 teachers—professors from the University were registered to use the system. The database consists of 74 courses with 27,027 questions and 107,116 offered answers, i.e. average 4 answers per question. 63,255 tests were processed and 1.585.126 questions were set to students, i.e. each exam consists of average 25 questions. 51 courses were using the online learning tool with adaptive testing. On average each of these online courses has 81 parts and 900 learning objectives—knowledge items. Each student has answered on average 4, 5 questions per learning objective (knowledge item) using the 3 in-a-row strategy. Once the students passed this type of online learning with adaptive testing they successfully passed the exam without a lot of effort. It really helped them to fulfill their knowledge.

The system is interactive with course material since if a wrong answer is obtained the student goes back to the material to learn the concepts and comes back to the system. The process where students realized the self-testing led to more motivation, since they tested their knowledge more often and this accomplished the learning process more successfully.

In this paper we presented state of the art about e-Assessment systems. We have analyzed details of organization of e-Testing, and compared our definition to other e-Assessment systems addressing the interactive assessment. We also address the architecture and organization of a cloud solution, and coverage of interoperability standards and recommendations, aiming that the online learning system with interactive testing can be used as a tool or upgrade to the existing LMS systems.

The online learning system with adaptive testing was presented with specification of main requirements organization of the knowledge database, algorithms and procedures for testing process, including test delivery models, test creation algorithm and grading strategy. A special section is dedicated to the navigation algorithm and the decision making strategy. The solution uses software agent technology and classification algorithm for navigation through learning objectives and realizes appropriate decision making strategy.

The experiments were analyzed to conclude when the student has learned corresponding learning objective. The comparisons showed that 3R (**3 correct answers in a row**) is the best both for the students and the teacher. The conclusions are brought by analyzing the interviews with students and teachers about their motivation and impact. Several conclusions also reflect their attitude that the online learning system with adaptive testing actually implements:

- Students like fun and entertainment, i.e. the system is a kind of a computer game in the learning process.
- Students like challenge, freedom, unexpected elements, they like competition and this system offers it.
- Any strategy trying to repeat the same question until a correct answer is given makes the system a boring place.

This paper summarizes how e-Assessment systems are built, and how e-Testing can be implemented for e-Education, putting the accent on new trends in technology and implementation of AI related methods to establish better learning

system. The final goal is to realize how e-Testing supports the e-Education making the “e” in e-Education to stand for **efficient** instead of just electronic.

In near future we plan to make advanced research on the discussed strategies, including more statistical tools and AI related methods. We have started to develop a new cloud based solution in 2012 and the process is ongoing, trying to be consistent with existing data and system. The application of the online learning system with adaptive testing does not depend on whether the system is cloud based or not, although the cloud based solution will enable to use it as a tool in different environments and LMS, by exchanging appropriate information.

References

1. Armenski, G., Gusev, M.: E-testing based on service oriented architecture. In: Proceedings of the 10th CAA International Computer Assisted Assessment Conference, pp. 17–23. © Loughborough University (2006)
2. Armenski, G., Gušev, M.: Infrastructure for e-testing. *Facta Univ. Ser. Electron. Energ.* **18**(2), 181–204 (2005). <http://factae.elfak.ni.ac.yu/fu2k52/contents.html>
3. Armenski, G., Gusev, M.: The architecture of an ultimate’ e-assessment system. In: ICT Innovations 2012, Web Proceedings. Association for Information and Communication Technologies ICT-ACT (2009)
4. Bădică, C., Budimac, Z., Burkhard, H.D., Ivanović, M.: Software agents: Languages, tools, platforms. *Comput. Sci. Inf. Syst./ComSIS* **8**(2), 255–298 (2011)
5. Beck, J., Stern, M., Haugsjaa, E.: Applications of ai in education. *Crossroads* **3**(1), 11–15 (1996)
6. Blackboard: (2013). <http://www.blackboard.com>
7. Brusilovsky, P.: Developing adaptive educational hypermedia systems: From design models to authoring tools. *Authoring tools for advanced technology learning environment*, pp. 377–409. Kluwer Academic Publishers, Dordrecht (2003)
8. Brusilovsky, P., Peylo, C.: Adaptive and intelligent web-based educational systems. *Int. J. Artif. Intell. Educ.* **13**(2), 159–172 (2003)
9. Clark, R., Mayer, R.: *E-learning and the science of instruction: proven guidelines for consumers and designers of multimedia learning*. Pfeiffer (2011)
10. Classroom, T.M.V.: (2012). <http://manhattan.sourceforge.net>
11. Collins, A.M., Stevens, A.L.: *A Cognitive Theory of Interactive Teaching*. Bolt Beranek and Newman Inc, Cambridge (1981)
12. Crisp, G.: *Interactive e-assessment: Moving beyond multiple-choice questions*. Centre for Learning and Professional Development. Adelaide (Australien): University of Adelaide (2009). http://ipac.kacst.edu.sa/eDoc/2009/173798_1.pdf [10.11.2010]
13. Dagger, D., O’Connor, A., Lawless, S., Walsh, E., Wade, V.: Service-oriented e-learning platforms: from monolithic systems to flexible services. *IEEE Internet Comput.* **11**(3), 28–35 (2007)
14. Davies, W.M., Davis, H.C.: Designing assessment tools in a service oriented architecture. In: 1st International ELeGI Conference on Advanced Technology for Enhanced Learning, Vico Equense-Naples, Italy, 15–16 Mar 2005, pp. 1–7 (2005)
15. Environment, F.L.: (2012). <http://fle3.uiah.fi>
16. eSurvey Creator: (2013). <http://www.esurveycreator.com>
17. eSurvey Pro: (2013). <http://www.esurveypro.com>
18. ETS: (2012). <https://www.ets.org/>
19. EL Framework: (2012). <http://www.elframework.org/>

20. Franklin, S., Graesser, A.: Is it an agent, or just a program? A taxonomy for autonomous agents. *Intelligent Agents III Agent Theories, Architectures, and Languages*, pp. 21–35 (1997)
21. Free Online Surveys: (2013). <http://www.freeonlinesurveys.com>
22. Gierłowski, K., Nowicki, K.: A highly scalable, modular architecture for computer aided assessment e-learning systems. *Distance Education Environments and Emerging Software Systems: New Technologies*, IGI Global (2011)
23. Gusev, M., Armenski, G.: On-line learning and e-testing. In: *Information Proceedings of the 24th International Conference on Technology Interfaces ITI 2002*, pp. 147–152. IEEE (2002)
24. IMS Global Learning Consortium: Abstract Framework: (2012). <http://www.imsglobal.org/af/index.html>
25. IMS Content Packing Specification: (2012). <http://www.imsglobal.org/content/packaging/>
26. IMS Learning Design Specification: (2012). <http://www.imsglobal.org/learningdesign/>
27. Kaernbach, C.: Simple adaptive testing with the weighted up-down method. *Atten. Percept. Psychophys.* **49**(3), 227–229 (1991)
28. Koper, R., van Es, R.: Modelling units of learning from a pedagogical perspective. *Online Educ. Using Learn. Objects.* **40**, 43–58 (2004)
29. LeAP Project Case Study: Implementing Web Services in an Education Environment: (2012). <http://www.education.tas.gov.au/admin/ict/projects/imsoecasestudy/LeAPProjectCaseSummary.pdf>
30. LRN: (2012). <http://dotlrn.org>
31. Luecht, R., Sireci, S.: A review of models for computer-based testing (2012). <http://research.collegeboard.org/publications/content/2012/05/review-models-computer-based-testing>
32. Moodle: (2013). <http://moodle.org>
33. OKI Project: (2012). <http://www.okiproject.org>
34. Paramythis, A., Loidl-Reisinger, S.: Adaptive learning environments and e-learning standards. In: *Second European Conference on E-Learning*, pp. 369–379 (2003)
35. Parshall, C., Davey, T., Pashley, P.: Innovative items for computerized testing. In: van der Linden, W.J., Glas, C.A.W. (eds.) *Elements of Adaptive Testing, Statistics, for Social and Behavioral Sciences*, pp. 215–230. Springer, Berlin (2002)
36. Patelis, T.: An overview of computer-based testing. The College Board, RN-09, Office of Research and Development (2000). <http://www.collegeboard.com/research/html/rn09.pdf>
37. Pearson Vue: (2013). <http://www.pearsonvue.com>
38. Prometric: (2013). <http://www.prometric.com>
39. Ristov, S., Gusev, M., Armenski, G., Bozinoski, K., Velkoski, G.: Architecture and organization of e-assessment cloud solution. In: *2013 IEEE Global Engineering Education Conference (EDUCON)*, pp. 736–743 (2013)
40. Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*, 3rd edn. Prentice Hall, Englewood Cliffs (2002)
41. Scalise, K., Gifford, B.: Computer-based assessment in e-learning: A framework for constructing “intermediate constraint” questions and tasks for technology platforms. *J. Technol. Learn. Assess.* **4**(6), 5–44 (2006)
42. Sclater, N., Howie, K.: User requirements of the ultimate online assessment engine. *Comput. Educ.* **40**(3), 285–306 (2003)
43. Sclater, N., Low, B., Barr, N.: Interoperability with CAA: does it work in practice? In: *Proceedings of the Sixth International Computer Assisted Assessment Conference*, Loughborough University, 317–326 (2002)
44. Tattersall, C., Hermans, H.: Ounls assessment model. In: *January 2006 TENCompetence WP6 Meeting* (2006). <http://dSPACE.OU.NL/handle/1820/558>
45. TENCompetence—Building the European Network for Lifelong Competence Development: (2012). <http://www.tencompetence.org/>
46. Thissen, D., Mislevy, R.: Testing algorithms. In: Wainer, H., Dorans, N., Green, B., Steinberg, L., Flaugher, R., Mislevy, R., Thissen, D. (eds.) *Computerized Adaptive Testing: A Primer*. Lawrence Erlbaum Associates Inc, London, 101–133 (2000)

47. Thompson, N., Wiess, D.: Computerised and adaptive testing in educational assessment. *Trans. Comput. Based Assess. New Approach. Skills Assess Implications Large-Scale Test.* 127–133 (2009)
48. Thurlow, M., Lazarus, S., Albus, D., Hodgson, J.: Computer-based testing: Practices and considerations. *Synth Rep.* **78** (2010)
49. Vossen, G., Westerkamp, P.: E-learning as a web service. In: *Proceedings of the Seventh International Database Engineering and Applications Symposium*, pp. 242–249. IEEE (2003)
50. WebCT: (2013). <http://www.webct.com>
51. Williamson, D.M., Bennett, R.E., Lazer, S., Bernstein, J., Foltz, P.W., Landauer, T.K., Walter, D., Way, D., Sweeney, K.: Automated Scoring for the Assessment of Common Core Standards (2010). <https://www.ets.org/s/commonassessments/pdf/AutomatedScoringAssessCommonCoreStandards.pdf>