Vasis, Christos; Arabatzi, Eleana; Sinatkas, Ioannis et al.

Article

Design and implementation of a multiple equivalent forms print, readout and results processing system

Provided in Cooperation with:
Technological Educational Institute (TEI), Thessaly

This Version is available at:
http://hdl.handle.net/11159/2857

Design and Implementation of a Multiple Equivalent Forms Print, Readout and Results Processing System

Vasis Christos, Arabatzi Eleana, Sinatkas Ioannis, Nikolaou Angelos, Mpatos Panagiotis
Informatics Engineering Department
Technological Educational Institute of Western Macedonia
christosvasis@gmail.com,eleana1992@hotmail.gr,sinatkas@kastoria.teiwm.gr
angelonikolaou@gmail.com,p.mpatos@kastoria.teiwm.gr

Abstract

Examinations with a large number of examinees require considerable time from the examiners. An effective way to deal with this is to use a multiple-choice question system. The proposed system was designed to create an automated mechanism for designing and reviewing the exam form.

Such visual recognition systems have been applied to Joker, Lotto, etc. In these cases however, the papers are pre-printed in huge quantities. This is not the case for examinations that differ from course to course and per exam period. Also, the answer form cannot be a separate section of the rest for validation reasons.

Taking this into account, an electronic system was designed and implemented, in which the examiner can input the content of the examination form and the system will correct it automatically.

Thus, an application was created that can manage and process the responses effectively, in a manner which reveals the degree of the students' understanding of the course material. For example, if a large number of examinees give the same wrong answer to a particular question, this may mean that either the question was not written correctly, or its content was not understood. On the other hand, if the question was answered correctly by a very large number of examinees, the question is perhaps too simple.

The system was implemented using a MySQL database that includes courses, questions, possible answers per question, and eligible examinees. The application was developed with Python environments. The equipment required for operating the system is a fast printer and a scanner with an automatic document feeder.

From the scanned form, the application identifies the student, the exam date, and the answers. Finally, the system provides the overall score and useful conclusions.

The application allows for more frequent examinations; resulting in a gradual understanding of the subject material and a better overall assessment of the students.

Keywords: OMR, Exam System, Management with Information Technologies, Research and Development, and Technological Innovation

JEL classifications: M150 IT Management, O320 Management of Technological Innovation and R&D

Introduction

Assessment is an important stage of the educational process. There are three types of assessment: initial, formative, final. The aim of the initial assessment is to identify the perceptions, shortcomings, and difficulties faced by students in the content of teaching. It helps the teacher to organize the subject material and the teaching method. Formative assessment aims to monitor the educational path towards the goals. The final evaluation certifies and checks the final result and attributes the learner's overall assessment of their adequacy and success rate to the objectives of the course.

The Bloom classification (Bloom, 1956) tried to group the objectives of education and is used as a framework for the organization of the teaching material and examinations. It has been observed that teachers make a major effort in preparing and delivering the lectures, but they often do not make a similar effort to prepare the examinations at all levels. They consider evaluation as the final count. They do not use it as a pedagogical tool because in a large population of learners the frequent evaluation process is difficult to apply. The workload and time required are deterrent factors. What can now decisively contribute to addressing the issue is its mechanized response.

Automated form-based mechanics limit the ways of examining closed-ended questions; that is, questions which have specific answers. The type of questions that can be used with the proposed system are multiple choice questions, true-false questions, and fill-in-the-blank questions. Multiple-choice questions are the most popular choice in this context and have the following advantages:

- They allow for the examination of a large number of examinees
- They allow for the examination of a wide range of the course material
- They are objective
- They are graded easily and quickly

However, they also have drawbacks:

- Encourage copying
- May be tricky or misleading
- Unable to assess higher cognitive functions
- Creation is difficult and time-consuming

Automated systems for this purpose have been suggested and developed previously, and they have recommended ways to evaluate examinees (Harris & Buckley-Sharp, 1968; Diament & Goldsmith, 1970). These systems have improved over time and are used in the examination of large amounts of examinees, such as in health-related faculties. The issues of preventing cheating by copying and quickly correcting the forms have been effectively resolved using computers for exams (Rjoub et al., 2009), but their use for a large amount of examinees is limited by the amount of computers and computer rooms that institutions have available. The only efficient alternative is simultaneous examination using pen and paper. Towards that end, systems were created that use computers for generating equivalent digital tests which are printed and afterwards corrected digitally (Villalon, 2012). Today, systems exist, such as ExPro, which create printed tests with multiple-choice questions and allow the reordering of questions and the archiving and management of results,

(Bani-Ahmad & Audeh, 2010). However, the proposed system is capable of creating not only a unique but also an equivalent questionnaire per examinee, including also a method for automated correction, supported additionally by a quality check tool for assessing the quality of the test questions.

The aim of this study was to develop software that creates and marks exam forms in a way that effectively addresses copying, is mechanically corrected, evaluates the quality of its queries, and identifies each and every one of its problems.

So far, to combat copying, a common method used is to print two or three different sets of questions, with a different order of questions in each set. To correct them, they are then grouped by set.

This paper proposes a way to distribute a different questionnaire to each individual. Each produced questionnaire has the exact same set of questions and the same exact options in each question but rearranges both its questions and the individual options of each one. This is achieved at the production stage of the forms and their correction is addressed by using the QR printed on each one. The QR contains, among other things, information on the order of questions and their choices.

Several criteria have been proposed to control the quality of multiple-choice questions. The criteria that have been proposed are the degree of difficulty and the degree of discretion. These measures are calculated taking into account two population groups: one group being composed of the highest-scoring 25% of all examinees, and the second being the lowest-scoring 25% of all examinees. However, the second group includes examinees who were examined without being adequately prepared and answered the questions at random. Additionally, the total population contains examinees who have taken part in exams before and therefore have encountered the questions before. These examinees hold an advantage against those who are facing the questions for the first time and the test evaluation criteria should not be considering them equally to those who are being tested for the first time. In previous evaluations these two subgroups have not been excluded and this has affected the resulting test quality (Abdulghani et al., 2015; Nanas et al., 2008; Abdulghani et al., 2017).

In this paper, a mechanism is proposed and implemented that verifies the answers with appropriate criteria and identifies the forms that were answered in a random fashion. From the evaluation of the questions and in particular from its categorization, it isolates and does not take into account these subgroups.

Also, to date, quality evaluation is limited only to correct or incorrect choices and does not extend to the investigation of whether the question was phrased accurately or not. For example, if multiple students give the same wrong answer to a question, it can be considered that the answer phrased in a misleading manner; or if the correct answer is never selected, it is potentially wrong. The application keeps all the answers and, in the future, will explore the results at this level as well.

Designing the system

The application initially creates a form template, in our case 14 questions with four options each. In the template, the right answers are always the first choice. In each generated question form, the questions change order in a random fashion and so do the order of the answers. As a result, the correct selection of each question will be in a different order.

The changes to the order of the questions and answers of each form, compared to those in the template, are recorded as QR information generated and printed in the document of the questions. In order to avoid any QR decoding problem, the application encrypts this assignment with a PIN encryption key of the examiner's choice.

For example, the QR of the form of the questions in Figure 1 contains the DATA STRUCTURE information, 3/9/2018, SINATKAS I., 2129271C282023251D221F1E2624, 69CFCC54A266936687A248EDAECF. The alphanumerics contained in the QR have been encrypted with the examiner's PIN. The first one has coded the random series of questions in relation to the standard, while the second contains the matching of the options of each question of the form with respect to the template.

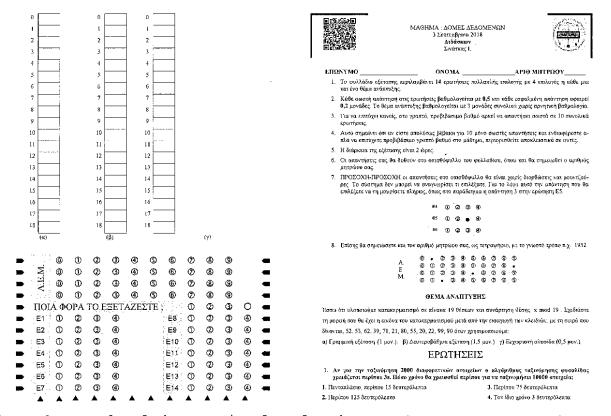Figure 1. Example of the question form for the Data Structures course. QR can be read with any modern device.

Now, to correct the forms, all are scanned and presented for correction. When scanning, proper application functions repair potential problems such as image shifting or minor distortion. When the image is repaired to match the scans of each scanned copy with the pattern, the application proceeds to read the scans.

The application, in each text, decrypts the QR that it carries and locates both the order of the questions that correspond to the particular writing versus that of the template and the corresponding order of the options. Given that it has been agreed that the first ever choice of the template is correct, the application knows the correct answer to each question.

From the same scanned image, the application first locates the exam candidate by reading the candidate PIN and then process the answers. It compares the answers to the correct ones and extracts the result of the examination. But it is not limited to this, it records the selections of the individual in each question. This knowledge is needed to investigate problematic "misleading" answers, and the degree of discrepancy.

To assess the quality of the test, the population of those examined is divided into three categories. The first category is 25% of the students who received the highest total scores, the second group is 25% students who received the lowest scores, and the third group is all other examinees. (Abdulghani et al., 2015; Nanas et al., 2008; Abdulghani et al., 2017)

Index E indicates the success rate of the question. The larger the E, the easier the question. Questions with E = 30-70%, with a recommended difficulty index of 50-60%, are accepted. Questions with E > 70% are considered to be easy and questions with E <30% are considered difficult. E is determined by the relationship

$$E = 100\frac{X+Y}{N}$$

where X is the number of correct answers to this question by the examinees belonging to the first category of high overall score, Y is the corresponding number of the second category of the low scores, and N is the total population of the two categories.

The discrepancy index Δ expresses the effectiveness of the question of distinguishing between the two categories of subjects. The value of index Δ ranges from -1 to 1. The higher the value of this index, the more effective the question is. Accepted values Δ> 0,15 with excellent Δ> 0,35 and rejected with Δ <0,15. D is determined by the relationship

$$\Delta = 2\frac{X-Y}{N}$$

However, an important issue in the calculation of these indicators is the identification of the appropriate population of examinees to be taken into account. It is known that a significant number of examinees responds randomly to the questions. This number has to be identified and the written tests completely excluded from the test evaluation. Also, those taking part in the exam for the second time should not be counted in the population because they already have experience with at least some of their questions.

Applying appropriate algorithms, it locates these forms and removes them from the set. Only the first-time questions are left to evaluate the test and those that did not respond in a random fashion. The purpose of the direct evaluation of the test is the fairer graduation of the test. The analysis provides the possibility of a possible reassessment of the score if problems with certain test questions are found.

Implementation

First of all, it must be noted that the system was designed to deal only with four-choice questions and is limited to questions only in alphanumeric terms; i.e., no shapes or images in the question or the choices. It is also assumed that for every question there is only one correct answer. Such systems have already been developed and marketed but they are limited to the production of a single form. (Anova Consulting, 2016; Gravic, 2018; Arabatzi, 2018)

To achieve this, we need to create an appropriate database that includes the courses and, in each course, the organized questions per chapter of the curriculum. Each question is categorized by the examiner as easy, normal, or difficult. Each question and the four answer options, with the correct one always in the first place, are recorded in the database. The design of the database is shown in Figure 2.

The questions that form the standard test subject are selected by the examiner with certain criteria. That is, if it concerns a final examination, it can choose from each chapter a question in order to cover the entire range of the subject. At the same time, it can determine the total degree of difficulty and with this criterion it selects questions of similar difficulty from each chapter. For example, the examiner can choose all the questions of normal difficulty or choose 25% difficult, 25% easy, and the other normal.
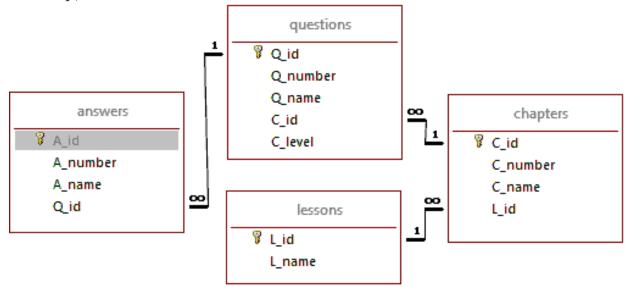

Figure 2. The design of the database. The entities with the links that link them are displayed.

To create the questionnaire form, an appropriate template is created. Its fixed parts are the header, that contains the information of the specific course, the corresponding Institute or Department label and the space where the QR is placed in. Following those are the examiner's instructions, which are common to all forms, and then the questions themselves. The last page of the questionnaire contains the answer sheet. In this example, the questions were printed on A4 sheets using the

Booklet option on a Ricoh Aficio MP 9000 PCL copier of the Technological Education Institute of Western Macedonia in Kastoria.

This sheet includes a section of options in which the examinee inputs their registration number (AEM) by marking the appropriate numbers. following that, the examinee marks answers in the answer sheet. The specific questionnaire includes 14 questions with four options each. There is also input space for the examinee to mark how many times they have been previously taken the exam.

The system was tested in an actual module, the examiner chose to not only use multiple choice questions, but also add an open-ended question which was corrected manually. Figure 1 contains the corresponding space for answering the open-ended question. The examiner can format the template appropriately to provide the proper space so that the examinee has to answer concisely.

Also, for technical reasons, the form has scanning frames printed on it, so that the form can be scanned as efficiently as possible.

The process is as follows:

1 The examiner selects the lesson to be examined.
2 Selects the chapters of the lesson to include.
3 Decides on the degree of difficulty and selects the questions from database.
4 The application creates the standard theme with these questions and always has the correct answer as a first choice in each question.
5 Enters the PIN to use in the application.
6 Asks the system to create a specific number of examination forms.
7 In each of these forms, the QR is created and registered.

The order of questions in each form is derived from a mathematical function that creates random rearrangements. For 14 questions, there are more than 87 billion rearrangements. Also, the same applies to every question, where there are 24 rearrangements of the four possible answers. In total, there are more than 2 trillion combinations with the 14 questions with four answer options each. It is therefore practically impossible for examinees sitting next to one another to exchange any relevant information.

The QR image is generated by a function that accepts both the fixed test data; such as the date, the lesson, and the examiner; and the variable data from the rearrangement of questions and options. These variables are the two records that result from encrypting the PINs of the first order of the queries, and then the sequence of each query options. The form in Figure 1 is 2129271C282023251D221F1E2624 and 69CFCC54A266936687A248EDAECF.

The correction phase includes the following steps:

1 The questionnaires are scanned using a scanner with an automatic feeder tray.
2 The scanned result is saved in an image file.
3 The examiner inputs the PIN used when creating the forms.
4 The file is submitted to the application, which reads each page separately.
5 On each page, it reads and decodes the QR first.

6 The decoding of the QR shows the order of the questions and the order of their answer choices.
7 By reading the marked number options, the application extracts the student's registration number. By comparing this registration number and the student list, the system has all the required information.
8 The application scans the "NUMBER OF PREVIOUS EXAM ATTEMPTS" in order to identify if the answers will be used in assessing the quality of the test.
9 The answer selections of the examinee are scanned and recorded in an appropriate file. For this, the application takes into account the reordering of the specific questionnaire's questions and answers, and the default form's questions and answers.
10 When the application completes the logging process, it outputs the final score for each examinee.

The log file containing each examinee's options is then used to evaluate the test. Initially, with a suitable filter, only those who are being examined for the first time are selected. Then those who did not answer any question are deleted. A suitable algorithm is applied to the rest of the examinees to identify those who responded completely randomly. For example, those who choose to select multiple consecutive marks in the same column or more generally find themselves following a pattern of choices not justified by the correct test answers.

In the final remaining examinees, the relations of determining the degree of difficulty / ease E and the degree of discretionality Δ are applied. The result is more expensive than has been applied so far because the writings and the population taken into account are more reliable. At the same time, the system can indicate the option that poses the problem to the questions that it poses. It will be that wrong choice chosen by most people. Similarly, the "misleading" response that was not chosen by any of the respondents will obviously fail.

Tools

The tools that were used to implement the application are:

1 The MySQL database management system in version 5.7.19
2 PYTHON 3.5.6 was used to implement the applications
3 The pyqrcode library was used to create and read the QR. (Nooner, 2013-16 pythonhosted.org/PyQRCode)
4 The RICOH Aficio MP 9000 digital copier of the TEI of Western Macedonia in Kastoria was used for printing and scanning the forms.

The relational database management system MySQL was selected because it is a widely used open source software that fully covers the needs of the specific application. For its implementation, the Python programming language was selected as it is also open source, offers an extended library of script modules, and is a modern programming language with powerful capabilities.

Future implementations

1 The application can be improved by supporting more types of questions besides multiple choice ones.

2 It can be enriched with multiple choice queries that provide images in the question section or in the answer options.
3 It can be connected to the seat booking management system that has been previously developed to cross-check data and investigate the seat groupings of examinees. (Efthimiopoulos, 2018)
4 It can be combined with a networking system that can email the corrected questionnaire to each student in order to learn from their mistakes, should they request it.

Conclusions

The application is an extremely useful tool because it:

1 Automates to a great extent the creation of multiple-choice questionnaires.
2 Creates a unique questionnaire form for every examinee, making it difficult to copy.
3 Corrects the questionnaires in an automated manner.
4 Enables each teacher to make more frequent evaluations, even with a large number of students.
5 Evaluates the quality of the test and enables the examiner to improve problematic questions and, if needed, to reevaluate the final score of the students accordingly.
6 Benefits the student by scoring them not for one single final exam, but instead taking into account their performance in several intermediate exams.
7 Can be used to attract students to regular lectures in order to participate in a weekly formative assessment.

During the practical implementation of the application, it was found that the 90% of the students who participated in these weekly assessments during the academic year 2017–2018 passed their examinations, when the overall success rate was 40%.

Most of the systems that exist today, such as Moodle, BlackBoard, eClass and others, offer extended capabilities of electronic test creation and correction, but they are systems that require a computer for each examinee, which is practically inefficient for the simultaneous examination of a large number of examinees. On the other hand, systems exist that create questionnaires and offer an OMR (optical mark recognition) process for automatic correction, such as Expo, FormScanner and Remark Office OMR, but don't offer, or offer to a limited extent, the capability of creating multiple questionnaires, which is the core feature of the suggested application. Our application creates unique questionnaires, which makes cheating harder, if not impossible.

Acknowledgements

References

Abdulghani, H.M., Ahmad. F., Irshad, M., Khalil, M.S., Al-Shaikh, G.K., Syed, S., Aldrees, A.A., Alrowais, N., and Haque, S., 2015, "Faculty development programs improve the quality of Multiple Choice Questions items' writing", *Scientific Reports*, 5(9556).

Abdulghani, H.M., Irshad, M., Haque, S., Ahmad, T., Sattar, K., and Khalil, M.S. (2017) "Effectiveness of longitudinal faculty development programs on MCQs items writing skills: A follow-up study", *PLOS ONE*, 12(10).

Agricultural University of Athens, 2009, WebOMR, webomr.aoa.aua.gr, accessed 9/2018.

Anova Consulting 2016, "QuickTesting software®", www.anova.gr/pages/QuickTestingPage_eng.htm, accessed 9/2018.

Arabatzi, E., 2018, "Design and creation of software for scanning multiple-choice questionnaires and printing of results", Unpublished, TEI of Western Macedonia.

Bani-Ahmad, S., and Audeh, A., 2010, "Re-engineering of multiple-choice exam-form production tools: Cost-effective and quality-assurance approach" *In Educational and Information Technology (ICEIT), 2010 International Conference,* 3, V3-260.

Bloom, B.S., 1956, *Taxonomy of educational objectives, Handbook 1: The cognitive domain*, Longman, London.

Diament, M.L., and Goldsmith, R., 1970, "A Simple Automated Method of Marking Multiple-choice Questionnaires Using a Computer", *Medical Education*, 4(1), 53-55.

Efthimiopoulos, K., 2018, «Σχεδίαση και ανάπτυξη δικτυακής εφαρμογής δέσμευσης θέσης από εξεταζόμενο φοιτητή», Unpublished, TEI of Western Macedonia.

Ginger Webs Pvt. Ltd., 2016, "OMR Home", www.omrhome.com, accessed 9/2018

Gravic Inc., 2018, "Remark Test Grading", remarksoftware.com/products/test-grading, accessed 9/2018.

Harris, F. T. C., and Buckley-Sharp, M.,1968, "Automation of multiple-choice examination marking", *Medical Education*, 2(1), 48-54.

Mohini, and Singh, A.J., 2012, "Automated data capturing and recognition technology to minimize human intervention in university examination system", *Asian Journal of Science and Technology*, 4(12), 007-010.

Nanas, S., Gerovasili, V., Poulaki, S., Bouhla, A., Tripodaki, E., Loukas, T., Routsi, C., and Roussos, C., 2008, "Optimization of multiple choice examinations with the use of specifically designed software", *Archives of Hellenic Medicine*, 25(6), 781–785.

Nooner M., 2013-16, pythonhosted.org/PyQRCode, accessed 9/2018.

Python Software Foundation, docs.python.org/3.5/reference/index.html, accessed 9/2018.

Rjoub, A., Eyadat, Y., Sharou, N., and Mardeeni, L., 2009, "A Multi-form Multiple Choice Editor Exam Tool Based on HTML Website and Artificial Intelligence Techniques 1", *Journal of Computer Science*, 5(6).

Villalon, J., 2012, "An eMarking tool for paper based evaluations. In Advanced Learning Technologies (ICALT)", *2012 IEEE 12th International Conference*, 43-45.