CREATIVE MATH. & INF.
15 (2006), 156 - 166

Docimological perspectives on the computer assessment method

RĂZVAN RĂDUCANU

ABSTRACT. This paper discusses the advantages and disadvantages of the computer assessment method from a docimological point of view. The paper also proposes an useful application of database programming in Visual C++ regarding the computer based assessment process of students. More exactly, it is presented a mechanism is implementing a timer which can be used to set the time allowed for each question in the test. In the present model we have considered only three questions but the model can be adapted to support any number of questions.

1. HISTORICAL PERSPECTIVE

The use of computers in student evaluations is not a new idea, but nowadays this method has became more and more popular. Presently there are certain faculties such as medicine which use the computer assessment techniques on a large scale. The method has a number of advantages and a number of disadvantages relatively to the object of the evaluation. In this paper we will discuss these advantages and disadvantages in different cases. This method of computer assisted assessment is a truly big step forward for the docimology science.

The computer assessment method has been used since 1960 in various evaluation situations. The first tentative of computer assisted evaluation contained only text (i.e. no pictures and sound) and have confronted the candidate with a boolean choice. The main advantage for the candidate consisted in the instant feedback which provided him instantly the result of the test.

In the first years of using the computer assisted evaluation , this method was used mainly on formative evaluations, whose main goal was to provide instant feedback to the candidate during the classes, so the candidate would be able to have an idea about his knowledge during the semester. Presently the trend is to make the computer assessment method a summative value, so it can be used as a final evaluation for the students at the end of a semester.

The computer assisted assessment was mainly used in the '60 because of its temporal efficiency relatively to the traditional paper tests. As the computer assisted evaluation began to be used as a summative evaluation technique there were some debates regarding its performance relatively to the classic evaluation method. The conclusion was that this new kind of evaluation is preferred by the students and has a better performance compared to the traditional testing method.

Received: 06.02.2006. In revised form: 17.09.2006.
2000 *Mathematics Subject Classification.* 97-00, 97-04, 97U70.
Key words and phrases. *Docimology, computer assessment, tests, evaluation methods.*

2. Several ways to implement computer assessment method

The computer assessment method is presently used to evaluate students at a large scale on certain faculties. The computer assisted evaluation techniques can be classified in two main types. The first type is the most popular and the students mark theirs answers on paper. These papers are then scanned and the results are analyzed and interpreted by the computer. Finally the computer presents to the evaluator the final results. The second type of computer assessment method - which will be exemplified in this paper - is based on a computer interface in which the student inputs directly the answers. At the end of the test the computer can present the result of the test to the student, the feedback being instantly provided by the application.

From an educational point of view the main advantages of the computer assessment method are:

- the time saving for the evaluators;
- the instant feedback for the students;
- professors can track performance of individual students;
- testing can be delivered simultaneously on multiple sites;
- the professors can obtain real time feedback on their course and can adapt their course consequently.

The disadvantages from the same point of view are:

- this kind of testing favors the objective tests (with no open answer questions);
- the question banks require expertise in designing test items and appropriate data storage;
- invigilation may be difficult in centers where students sit side by side to work on the same test simultaneously;
- plagiarism is possible if a test is done on line with student access to the internet and email;
- requires very reliable and secure computer test delivery systems.

From an administrative point of view the main advantages are:

- allows a 100% objective evaluation;
- answer marking is automated and rapid
- tests can be assembled quickly from computer stored question banks

The disadvantages from the same point of view are:

- the testing program is expensive;
- the possibility of network failures leads to the preparation of backup evaluation techniques;
- the personnel who invigilate and designs the test should be trained accordingly.

The computer assessment technique can be used at several points in a course depending on the purpose of the assessment. A first type of testing regards the evaluation of the students' prior knowledge at the beginning of a course. This kind of testing is called "diagnostic assessment". The second type of computer assessment is called "self-assessment" and it is based on the students' needs to asses themselves during a course in order to identify their own weak points. The third type, the "formative assessment", was mentioned at the beginning of this paper. In this case students receive instantly computer generated feedback on their performance and the evaluators can make an idea about the students' knowledge up to the testing point. This is usually a mid-term kind of testing. The last type of computer assisted evaluation is the "summative assessment" which is the final evaluation– generally it is used as a final test to pass an exam.

The first computer based tests were strictly text based and the candidates were confronted only with boolean options. Presently there are different soft products which implement pictures, sounds and even movies. These kind of tests represent a big step forward for the computer assisted techniques relatively to the classical paper based tests.

Nowadays computer aided tests use the following controls:

- check boxes - for multiple choice questions;
- radio buttons - for questions that requests boolean answers;
- edit boxes or text controls - for free short answers;
- pictures or drawings - for image identification questions, etc.

All these controls are described at www.caacentre.ac.uk.

An important advantage of computer assisted evaluation is the fact that it can reproduce real conditions, which is a very important fact - especially for the medical schools evaluations. A great issue of this kind of tests is the security of the evaluation. At the standard computer assessment the possibility of fraud is very high because there is a very big possibility that the students can see each others screens. If the test is online there is the possibility that the students email themselves the answers or try to find the right answers on the internet. There are ways to avoid these problems by randomizing the questions and by using dedicated networks for such tests.

Another advantage of the computer testing consists in the fact that the questions can be stored in a electronic question bank. The professors can easily make new tests and these tests can be programmed to randomize the questions. In this case each student can take a different test based on the same set of questions. This aspect increases the test security. Another advantage of this aspect consists in the fact that the student can assess himself to verify his knowledge.

The computer assisted tests can be implemented on different levels of difficulty. If a student has passed a certain level of difficulty, he can go on to the next level and if the a student has failed to pass a certain level he has to take the test corresponding to that level once again.

One open issue of computer assessment is whether it can assess higher intellectual skills or not. This arises from the fact that it is difficult to programme computers to reliably assesa free text answers. This explains the predominance of objective tests in computer based testing. Is is not the question type that dictates what level of intellectual skill is tested rather it is the content of the question that determines what type of competence is assessed.

Before designing a computer based test, a professor should ask himself the following questions:

- Will computer based testing add to the existing assessment regime?
- Is the purpose of the proposed test formative or summative?
- Is there sufficient time to learn about computer based testing, write questions, design the test, learn how to use the testing system, and deliver the test?
- Is there local technical support available to deliver computer based testing?
- Is the local network and Internet access sufficiently robust to manage a computer based test?
- Can students' answers in a computer based test be stored securely?
- Are there sufficient computers to test large numbers of candidates simultaneously?
- Do all of the students have sufficient computer skills to perform adequately in a computer based test?
- Will students have an opportunity to practice with computer based testing before facing a summative computer based test?
- How will the test be scored and what kind of test analysis is required?
- How will the test be invigilated?

3. A COMPUTER ASSESSMENT APPLICATION

This section proposes a way of implementing a computer based test, which implements a timer. The timer can be used to set the time allowed for each question (in this case 10 sec). If the student cannot answer the question in 10 sec, he gets 0 points and the test moves further to the next question. First we create a database called TEST which contains two tables. The first table is called Întrebări and it contains the following four fields: întrebari, răspuns1, raspuns2 şi răspuns3 and the second table is called Înregistrare and it contains the following three fields: Nume, Prenume, and Grupă. As a suggestion, we propose to create the data base in Access. But the programmer can use any database generator to create it. Once we have created the database, we have to configure an ODBC data source to point to the database we have created.

In order to create the TEST we create an SDI application with ODBC database support. When we chose the support for database option we have to make sure to chose the table Înregistrare from the TEST database. Once we have created the application frame with AppWizard, we insert the following controls which are used to record the students' data.

We will implement functions on the EN_CHANGE messages of the edit controls because we cannot permit the enabling of the "Înregistrare" button until all the edit control fields-are filled. In the moment all these fields are filled, the "Înregistrare" button will be enabled. Pushing this button will determine the appearance of another button with the caption text: "Start". By pushing this button the student will enter the proper test.

Finally, when the student have gone through all the questions of the test, a message box containing the final score will appear. At the same time, this score will be written in the "Înregistrare" table and into a text file - which will be created by the application - together with the name, surname and group of the student.

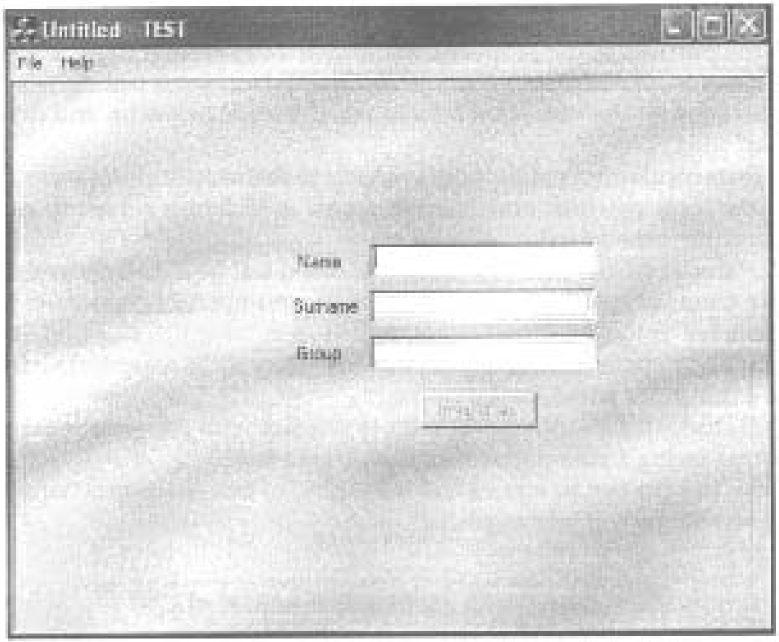
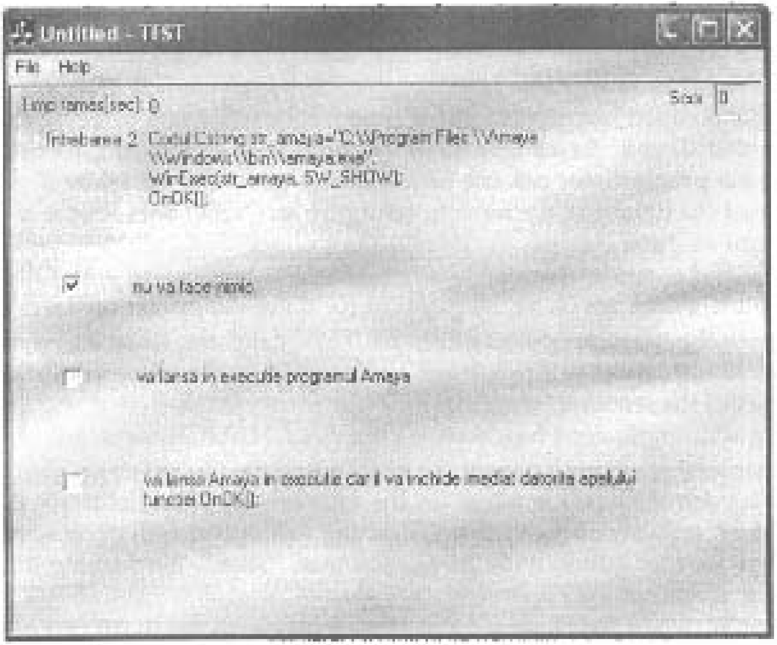


FIGURE 1. The introductory form.



Timp ramas[sec]: 0

Scor 0

Intrebarea 2: Codul CString str_amaya="C:\\Program Files\\Amaya
\\Windows\\bin\\amaya.exe";
WinExec(str_amaya, SW_SHOW);
OnOK();

☑ nu va face nimic

□ va lansa in executie programul Amaya

□ va lansa Amaya in executie dar il va inchide imediat datorita apelului functiei OnOK();

FIGURE 2. The main form.

In order to implement such a program, we will write the following code in the class derived from CView :

```
  void CTESTView::OnButton1()
  {
      // TODO: Add your control notification handler code
here
      CEdit* pe1=(CEdit*)GetDlgItem(IDC_EDIT_NUME);
      CEdit* pe2=(CEdit*)GetDlgItem(IDC_EDIT_PRENUME);
      CEdit* pe3=(CEdit*)GetDlgItem(IDC_EDIT_GRUPA);
      CString str1,str2,str3;
      pe1->GetWindowText(str1);
      pe2->GetWindowText(str2);
      pe3->GetWindowText(str3);
  if((str1!="")&&(str2!="")&&(str3!="")){
  GetDlgItem(IDC_INREGISTRARE)->ShowWindow(FALSE);
  GetDlgItem(IDC_START)->ShowWindow(TRUE);
  }
  }
  void CTESTView::OnStart()
  {
      // TODO: Add your control notification handler code
here
  ///////////////////////
  //CString str1, str2, str3;
  CEdit* pe1=(CEdit*)GetDlgItem(IDC_EDIT_NUME);
  CEdit* pe2=(CEdit*)GetDlgItem(IDC_EDIT_PRENUME);
  CEdit* pe3=(CEdit*)GetDlgItem(IDC_EDIT_GRUPA);
  pe1->GetWindowText(str1);
  pe2->GetWindowText(str2);
  pe3->GetWindowText(str3);
  if((str1!="")&&(str2!="")&&(str3!="")){
  FILE* fp=fopen("studenti.txt","a");
          fprintf(fp,"\n%s %s %s",str1, str2, str3);
          fclose(fp);
          ///////////////////////
  //UpdateData();
  //m_pSet->Update();
  //m_pSet->Requery();
  UpdateData(FALSE);
      GetDlgItem(IDC_EDIT_NUME)->ShowWindow(FALSE);
      GetDlgItem(IDC_EDIT_PRENUME)->ShowWindow(FALSE);
      GetDlgItem(IDC_EDIT_GRUPA)->ShowWindow(FALSE);
      GetDlgItem(IDC_STATIC_NUME)->ShowWindow(FALSE);
      GetDlgItem(IDC_STATIC_PRENUME)->ShowWindow(FALSE);
      GetDlgItem(IDC_STATIC_GRUPA)->ShowWindow(FALSE);
      GetDlgItem(IDC_START)->ShowWindow(FALSE);
```

```
        ///////////////
        GetDlgItem(IDC_STATIC_TEXT)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_INTREBARE)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_R1)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_R2)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_R3)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_NR)->ShowWindow(TRUE);
        GetDlgItem(IDC_CHECK1)->ShowWindow(TRUE);
        GetDlgItem(IDC_CHECK2)->ShowWindow(TRUE);
        GetDlgItem(IDC_CHECK3)->ShowWindow(TRUE);
  //      GetDlgItem(IDC_NEXT)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_TRAMAS)->ShowWindow(TRUE);
        GetDlgItem(IDC_EDIT_SCOR)->ShowWindow(TRUE);
        GetDlgItem(IDC_STATIC_SCOR)->ShowWindow(TRUE);
  /////////////////////////////////////////////
  //UpdateData(TRUE);
  m_iCount=10;
  m_sTime.Format("%d",m_iCount);
  SetTimer(ID_COUNT_TIMER, 1000, NULL);

  }
  }
  void CTESTView::OnNext()
  {
      // TODO: Add your control notification handler code
here
  //OnTimer(ID_COUNT_TIMER);
     CTESTDoc* pDoc=GetDocument();
     if(pDoc->m_strIntrebari.IsOpen()){
     if(!pDoc->m_strIntrebari.IsEOF()){
         pDoc->m_strIntrebari.MoveNext();
         m_Intrebare=pDoc->m_strIntrebari.m_Intrebari;
         m_R1=pDoc->m_strIntrebari.m_Raspuns1;
         m_R2=pDoc->m_strIntrebari.m_Raspuns2;
         m_R3=pDoc->m_strIntrebari.m_Raspuns3;
         m_Numar=pDoc->m_strIntrebari.m_Nr;
         m_pSet->Requery();
    }}

 int   stare1=m_Check1.GetCheck();
 int   stare2=m_Check2.GetCheck();
 int   stare3=m_Check3.GetCheck();
 switch(contor)
    {
        case 1:
                m_Check1.SetCheck(0);
                m_Check2.SetCheck(0);
                m_Check3.SetCheck(0);

                if((stare1==1)&&(stare2==0)&&(stare3==0))
                {
                        scor++;
                        UpdateData(FALSE);
                }
                break;
            case 2:

                m_Check1.SetCheck(0);
                m_Check2.SetCheck(0);
                m_Check3.SetCheck(0);

                if((stare1==1)&&(stare2==0)&&(stare3==1))
                {
                        scor++;
                        UpdateData(FALSE);
                }
                break;
            case 3:

                m_Check1.SetCheck(0);
                m_Check2.SetCheck(0);
                m_Check3.SetCheck(0);

                if((stare1==0)&&(stare2==0)&&(stare3==1))
                {
                        scor++;
                        UpdateData(FALSE);
                }
                break;
}
UpdateData(FALSE);
    contor++;
/////////////////////////////////////
    CEdit* pe4=(CEdit*)GetDlgItem(IDC_EDIT_SCOR);
    char sc[10];
    itoa(scor, sc, 10);
    pe4->SetWindowText(sc);
    UpdateData();
/////////////////////////////////////
    char to[19], total[100];
    strcpy(to,"Scorul final este:");
    wsprintf(total,"%s %d",to,scor);
    if(pDoc->m_strIntrebari.IsEOF()){
         KillTimer(ID_COUNT_TIMER);
         MessageBox(total,"Scor final",MB_OK|MB_ICONINFORMATION);

         FILE* fp=fopen("studenti.txt","a");
         fprintf(fp," %d",scor);
         fclose(fp);
    m_pSet->AddNew();
    UpdateData(FALSE);
    pDoc->m_TESTSet.m_Nume=str1;
    pDoc->m_TESTSet.m_Prenume=str2;
    pDoc->m_TESTSet.m_Grupa=str3;
    pDoc->m_TESTSet.m_Scor=long(scor);
    m_pSet->Update();
    m_pSet->Requery();
    UpdateData(FALSE);
       }
  }
  void CTESTView::OnChangeEditPrenume()
  {
    // TODO: If this is a RICHEDIT control, the control
will not
    // send this notification unless you override the
CRecordView::OnInitDialog()
    // function and call CRichEditCtrl().SetEventMask()
    // with the ENM_CHANGE flag ORed into the mask.

    // TODO: Add your control notification handler code
here
    CEdit* pe1=(CEdit*)GetDlgItem(IDC_EDIT_NUME);
    CEdit* pe2=(CEdit*)GetDlgItem(IDC_EDIT_PRENUME);
    CEdit* pe3=(CEdit*)GetDlgItem(IDC_EDIT_GRUPA);
    CString str1,str2,str3;
    pe1->GetWindowText(str1);
    pe2->GetWindowText(str2);
    pe3->GetWindowText(str3);
    if((str1!="")&&(str2!="")&&(str3!="")){
    GetDlgItem(IDC_INREGISTRARE)->EnableWindow(TRUE);
    }
  }
  void CTESTView::OnChangeEditGrupa()
  {
    // TODO: If this is a RICHEDIT control, the control
will not
```

```
        // send this notification unless you override the
  CRecordView::OnInitDialog()
        // function and call CRichEditCtrl().SetEventMask()
        // with the ENM_CHANGE flag ORed into the mask.
        CEdit* pe1=(CEdit*)GetDlgItem(IDC_EDIT_NUME);
        CEdit* pe2=(CEdit*)GetDlgItem(IDC_EDIT_PRENUME);
        CEdit* pe3=(CEdit*)GetDlgItem(IDC_EDIT_GRUPA);
        CString str1,str2,str3;
        pe1->GetWindowText(str1);
        pe2->GetWindowText(str2);
        pe3->GetWindowText(str3);
        if((str1!="")&&(str2!="")&&(str3!="")){
        GetDlgItem(IDC_INREGISTRARE)->EnableWindow(TRUE);
        }
    }
    void CTESTView::OnChangeEditNume()
    {
        // TODO: If this is a RICHEDIT control, the control
  will not
        // send this notification unless you override the
  CRecordView::OnInitDialog()
        // function and call CRichEditCtrl().SetEventMask()
        // with the ENM_CHANGE flag ORed into the mask.
        CEdit* pe1=(CEdit*)GetDlgItem(IDC_EDIT_NUME);
        CEdit* pe2=(CEdit*)GetDlgItem(IDC_EDIT_PRENUME);
        CEdit* pe3=(CEdit*)GetDlgItem(IDC_EDIT_GRUPA);
        CString str1,str2,str3;
        pe1->GetWindowText(str1);
        pe2->GetWindowText(str2);
        pe3->GetWindowText(str3);
        if((str1!="")&&(str2!="")&&(str3!="")){
        GetDlgItem(IDC_INREGISTRARE)->EnableWindow(TRUE);
        }
    }
    void CTESTView::OnTimer(UINT nIDEvent)
    {
        // TODO: Add your message handler code here and/or
  call default
                if(m_iCount==0){
                    m_iCount=10;
                    OnNext();
                }
                UpdateData(FALSE);
                m_sTime.Format("%d",m_iCount);
                m_iCount--;
                UpdateData(FALSE);
    CRecordView::OnTimer(nIDEvent);
}
```

In order to use this program in everyday tests, each professor has to construct his own database and also has to modify the code accordingly.

The advantage of this program consists in the fact that it eliminates any possibility of embezzlement (the answers should be given in a time interval) and finally it will write the student's dates and score in a database and in a text file. In this way the results can be posted immediately. More, the results can be indexed on name, surname or group or score because these have been written in a database.

4. Final remarks

Computer based testing offers many advantages over traditional paper based tests including automated marking and student feedback, multimedia question types and efficient test assembly. Computer based tests can be delivered anywhere via a secure computer network and are increasingly invigilated in dedicated assessment centers at some distance from the test source. Computer based test are not suitable for any course. For example this kind of tests is very useful in medical schools and in the faculties of informatics profile. On the contrary, for the mathematical courses there in no reasonable use of such a test, because the mathematical tests should test the way of thinking of every student and the dialog between professor and student is very important from the docimological point of view.

References

[1] Bull P., McKenna C., *Blueprint for computer-assisted assessment*, London, 2003

[2] Norman G. R., *Factors underlaying performance on written test knowledge*, Med. Educ, 1987

[3] Russel M., Hanney W., *Testing writing on computers: an experiment comparing student performance on tests conducted via computer and via paper and pencil*, Education Policy Analysis Archives, 1997

[4] Swets J.D., *Computer aided instruction*, Science, 1965

[5] Thelwall M., *Computer-based assessment: a versatile educational tool*, Computers and Education, 2000

[6] Ward W., *A comparison of free response and multiple choice forms of verbal aptitude tests*, Applied Psychological Mesurement, 1982

[7] Whittington D., *Technical issues and security*, *Computer assisted assessment in higher education*, London, 1999

Department of Applied Mathematics
"Al. I. Cuza" University Iaşi
Bd. Carol I Nr. 11
700506 Iaşi Romania
E-mail address: rrazvan@uaic.ro