UNIVERSITY OF PERADENIYA
SRI LANKA

PROCEEDINGS
OF THE
ANNUAL RESEARCH SESSIONS
2000

E.O.E. Pereira Theatre
Faculty of Engineering

DECEMBER 16, 2000

DEVELOPMENT OF "EdRex" – A COMPUTER PACKAGE FOR THE AUTOMATION OF EXAMINATION PROCEDURES IN THE FACULTY OF AGRICULTURE

H.V.A. WICKRAMASURIYA, M. BASNAYAKE*, and A. ANGAMMANA
Dept. of Agricultural Extension, Faculty of Agriculture, University of Peradeniya, Peradeniya
Computer Centre, Faculty of Agriculture, University of Peradeniya, Peradeniya

ABSTRACT

The efficient functioning of the Examination Unit is essential for the smooth administration of a university faculty. Therefore, the objective was to develop a software package for the Examination Unit. It was an in-house development through End User Computing at negligible financial cost. The package was developed as an application of Access[1] with Visual Basic[1]. Product development took two years during which time a Beta version and two other versions were rigorously tested. At present the product enables instantaneous registration of a full batch, of compulsory courses, and easy registration of optional courses, the generation of class lists, class attendance sheets, admission cards, examination attendance sheets, mark sheets, results sheets, and transcripts. Response time for each process is less than 2 seconds, enabling a very significant increase in the efficiency of the Unit. Some security features were provided. User support and training were provided. This package is currently being used in the Faculty and the response received has been extremely positive.

INTRODUCTION

For the smooth administration of a university faculty, efficient functioning of the Examination Unit is essential. In the Faculty of Agriculture a heavy burden is placed on this Unit due to the course unit system, which effectively trebled the amount of work in comparison to the former system of examinations held once an year. It is useful to automate procedures, which would greatly enhance the efficiency of personnel and also provide a quicker service in computing results, release of results, and issue of transcripts. The existing programme in dBASE[1] had only limited facilities and thus it had to be re-engineered or a fresh programme had to be developed. This project was an in-house development of a completely new computer package through End User Computing (i.e. development of software by users who are non-computer science personnel).

The Open University of Sri Lanka has developed a computerized system for its activities, including registration and examinations, under a foreign funded project with expatriate expertise (Johnson & Johnson, 1997). Such a project necessitates high levels of funding and expertise for development and maintenance. Since this is not feasible for all software development it was decided to develop a package at a low financial cost. Accordingly, the objective of this project was to develop, with the resources available within the Faculty, a software package that would automate all the routine procedures in the

Examinations Unit of the Faculty of Agriculture, and thereby increase the efficiency of the Faculty and improve the services provided to students.

"EdRex" the name given to the package is the abbreviation for Educational Registration and Examinations.

METHOD

This product was developed as an application of Access*, which is a Relational Data Base Management System, developed by Microsoft. Access* has the inbuilt ability to use Visual Basic* and is a 4GL (4th Generation Language) language enabling a higher level of End User Computing (EUC) even by non-computer science personnel. Access* was chosen due to the higher potential for use on other computers too since MS-Office (of which Access* is a component) is installed in almost all PC's. This is not the case with dBASE*, Fox-Pro*, or Oracle*. The ability to write code in Visual Basic* within Access*, congruity with other Office applications, the certainty of continued development of Access* by Microsoft, the availability of the royalty free Development Tool Kit and the relatively low cost were other reasons for the choice of Access*.

The following activities were undertaken in the development of the product.

1. Analysis of current process: The different activities undertaken in the Exam Unit and their sequence were initially analyzed.

2. Programme specification: The specific features and scope were determined. The initial focus was on routine procedures taking the most amount of time, and those needing quick processing for the examination board meeting.

3. Determination of expected outputs: The required views and printed reports were decided based on the documents previously used. Prior to printing each document, screen views were required.

4. Determination of required inputs: Data elements needed to provide the outputs were determined.

5. Development of the database structure: Attributes of the data entities were specified in data definition, which was done during designing of the tables. Tables were indexed on key identifying fields such as 'StudentID' (student ID No.). 'StudentID' was also assigned as the primary key in appropriate tables. Tables were related on keys such as 'StudentID' and Subject Codes. Individual students have multiple records in 'Class', 'Exam', and 'ExamGrades' tables for each subject. In the 'BioData' and 'CurrentRegistration' tables each student has a single record. Temporary tables were designed to contain dynamic and temporary lists and data. At the end of an examination, grades are transferred to an archive table. On completion of the degree the grades will be transferred to another archive table, which will be used for the generation of transcripts.

6. Designing of interfaces: Graphical mode devices such as switchboards and dialogue boxes were mainly used to provide a Graphical User Interface (GUI). Menus with drop down menus were also used in a few instances.

Designing and testing the functionality of forms: Flow control is governed by event driven procedures, mainly through 'OnClick' of command buttons on switchboards and dialogue boxes. These events trigger Visual Basic* code procedures, which are mostly confined to that object. Public procedures called up by queries such as in the computation of grades, were used as procedures in Modules. A sample of a part of the Visual Basic* code used to determine the letter grade is given in Fig. 1.

```
Public Function Grade_Letter(TotMark. TM As Double. PM As Single. TH. PH As Byte. Rs As
String) As String

If (TotMark < 0 And TotMark > -0.5) Then
   Grade_Letter = "....."
Elself (TotMark < 0) Then
   Grade_Letter = "ab"
Elself (TH <> 0 And PH <> 0 And (TM < 30) And (PM < 30)) Then
   Grade_Letter = "F"
Elself PH = 0 And (TM < 30) Then
   Grade_Letter = "F"
Elself PH <> 0 And (PM < 30) Then
   Grade_Letter = "FP"
Elself TH <> 0 And (TM < 30) Then
   Grade_Letter = "FT"
Else
If Rs = "N" Or Rs = "OB" Then
Select Case TotMark
  Case 84.5 To 100
     Grade_Letter = "A"


  Case 0.0 To 29.4
     Grade_Letter = "F"
End Select
Else
Select Case TotMark
  Case 54.5 To 100
     Grade_Letter = "C"
```

Fig. 1. Sample of Visual Basic* code (used in determining letter grade)

8. Development, unit testing and debugging of Queries and Modules: Data manipulation in queries is done through Standard Query Language (SQL). The SQL code was generated automatically by Access* upon completion of the Query By Example (QBE) grid in query design.

9. Development, unit testing and debugging of Reports: The content of documents was determined mainly on the content of the corresponding documents used by the Faculty previously. However, additions (such as total number of students in the examination attendance sheet) were incorporated to increase the utility of the documents. Document layout was done to minimize paper usage and also to improve its appearance. Information common to all students (such as Notes on transcripts) were pre-printed on the risograph as forms to minimize printing time and cost.

10. Integration testing and debugging: The different components were integrated through the switchboards and tested.

11. Data Conversion: Data processed in dBASE* was available for some batches. Data conversion was done to convert it to Access* tables and also to fulfil the table specifications used by "EdRex".

12. Incorporation of security features: The security features are described in a following section.

User input: A strength of the product was the close interaction between the developer and the end users, (i.e. Examination Unit staff), during many stages of development. Apart from the basic testing, much of the testing was done by the end users, using actual data. Whenever the users needed modifications, these were indicated and the modifications were incorporated into the programme and hence continual addition of new features, as required, has been done. For example, the ability to change 'repeat status' two months after sitting for the examination, on receipt of approval of Medical Certificate by the Chief Medical Officer and Faculty Board, was enabled on their request.

Development profile: Development of the product was started in September 1998, and the initial dry runs (Beta Version) conducted in mid-1999. It was tested using actual data in December 1999, (Version 1.0), to compute the results of one batch of students. This test run was successful. It was further developed and tested on two batches in July 2000, (Version 1.1), and this test too was successful. The present (December, 2000) version is 1.2.

Product components: In Access a database comprises of different objects such as tables, queries, forms, reports, macros, and modules. This package has 6 databases and a number of objects as given in Table 1.

Table 1. Product components

Component	Number
Databases	6
Tables	19
Queries	
Select Queries	25
Crosstab Queries	3
Make-Table Queries	10
Update Queries	7
Append Queries	6
Forms	42
Reports	11
Macros	8
Modules	7

RESULTS – THE PRODUCT

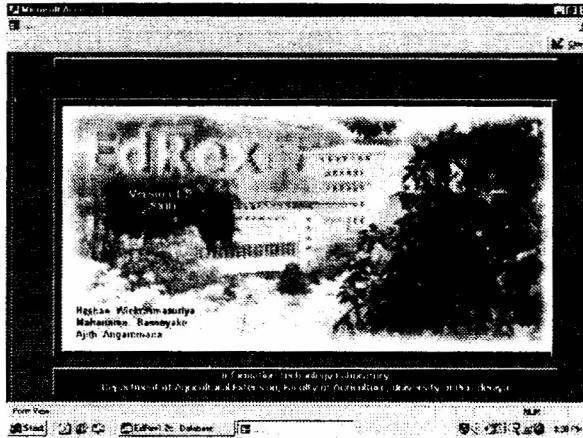The initial Splash Screen that appears on starting the package is as given in Fig. 2.

Fig. 2. Splash Screen of "EdRex"

The programme, which is user friendly, is operated through a series of switchboards and dialogue boxes, and can be used by any person with a basic level of computer literacy. Knowledge of Access[*] is not needed.

Main product processes and outputs: At present the product enables the following:
a) Instantaneous registration of a full batch for the ensuing year within the core programme.
b) Automatic registration for all the compulsory courses for a given majoring module.
c) Easy registration of optional courses.
d) Generation of class lists.
e) Generation of class attendance sheets
f) Automatic generation of admission cards.
g) Easy generation of examination attendance sheets for each subject.
h) Generation of grades.
i) Computation and printing of mark sheets.
j) Computation and printing of results sheets.
k) Viewing and printing of transcripts.
Sections of a few of the product outputs are given in Fig. 3 to indicate the nature of the hard copies obtained.

Use of product: In this system a student's name has to be entered only once for the entire student period of 4 years. Student ID No. is automatically generated in places such as the mark entry form for a particular subject. Entry of Subject Codes is not needed as they could be selected from the drop-down list of a combo-box, which is a type of control in Access.

Navigation of the system is through a series of switchboards which are forms meant to enable the user to move to the desired option in the package. The main switchboard (Fig. 4) enables the choice of the main components of the programme, which are Registration, examination attendance sheets, and admission cards (Admission), examination mark entry

and results generation (Results), and Transcripts. A tooltip can be seen when the cursor is placed on the command button for a short time.

1 Results Sheet   **2** Transcript   **3** Exam Attendance Sheet

Fig.3. Sample sections of Results sheet, Transcript, & Examination attendance sheet generated by "EdRex"
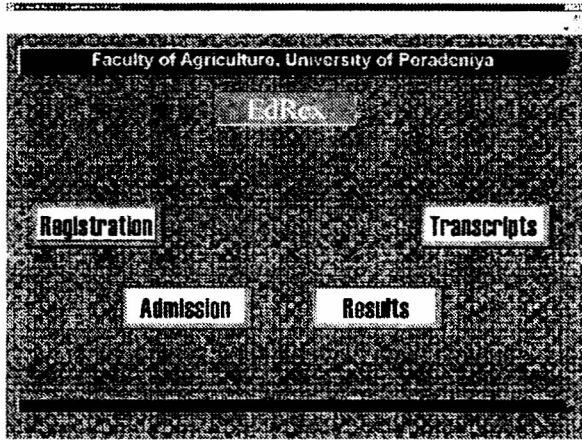
Fig. 4. The main switchboard.

Searches are possible based on Student ID No. and student name. Proximity searches could also be carried out by inputting just part of the name.

User input is mainly through forms. An example of a input form used by clerical staff in the Examination Unit is shown in Fig. 5. In this form, which is for marks entry, once the subject code has been selected it will automatically show the subject title, credits, theory hours, and practical hours, and lists the ID numbers of the students registered for that subject. On entry of the theory and practical marks it instantaneously displays the grade in a virtual field. Grade computation takes into account the ratio between theory and practical hours of the subject and the normal/repeat status of the student, as they are criteria in determining the grade. Accidental modification of the Student ID is prevented by disabling focus of that field and also by locking it to prevent editing. In many instances selection of values (e.g. student ID No., Subject Code, Semester), is through combo-boxes in order to facilitate use and minimize data entry errors.
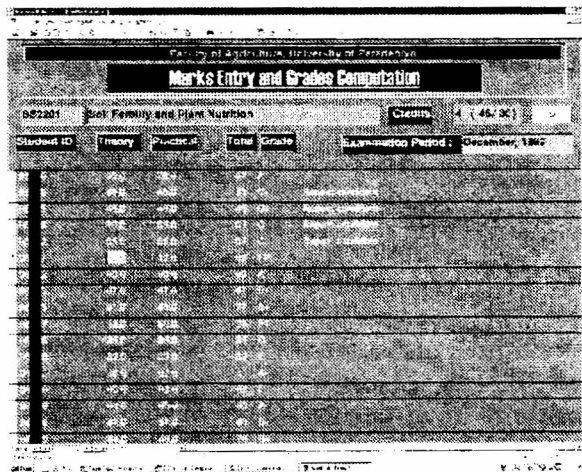
Note: Bogus marks to show computation of different grades.

Fig. 5. Data input form

At present marks received at the Examination Unit as paper based records are entered through the keyboard. Data verification is done by the data entry operator who checks the marks, and also by another person who rechecks the marks after which 'Checked' command button is clicked. If the 'Checked' button has not been clicked for all the subjects, the hard copy of the mark sheet and result sheet will bear the statement 'Test Copy' in bold letters, and hence cannot be submitted for the signatures of the appropriate authorities.

Prior to the print option, an on screen view of the document to be printed is shown.

Internal features: Data integrity checks to enable data validation are incorporated. For example, the inadvertent typing of a value <0 or >100 as a mark will automatically trigger a warning and request for correction. Warnings are also given when an important table is to be overwritten (Fig. 6)
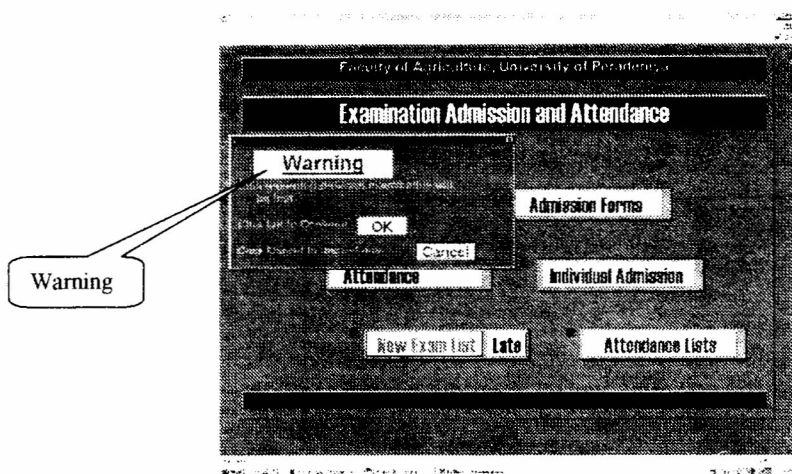

Fig. 6. Warning generated prior to overwriting of a table.

Constraints appearing in forms, such as the maximum allowable characters for the addressee of a transcript, ensure that the reports include all data.

Filtering of data is done based on fields such as Semester and Subject.

Scope: The product is limited to a single user at a time and does not support multiple concurrent users.

PRODUCT QUALITY

Performance: Processing is rapid, with all documents being processed in less than 2 seconds (Table 2). The response times given are based on performance on actual data for a batch of around 200 students, when the marks for all the subjects had been included, on an Intel Pentium III 650 MHz CPU, with 128 MB RAM. When data for all batches in the Faculty are contained in the tables the response time would probably be slightly slower for some of the processes. However, practically, this difference will not have any impact since the response times are so quick. Time taken for printing is additional and is dependent on the speed of the printer.

Table 2. Response time for main processes and documents

Process/ Document	Response Time Seconds (Approx.)
Process	
Register Batch	0.3
Register Compulsory Courses	0.8
Compute Grades	1.6
Document	
Class List	0.6
Class Attendance Sheet	0.6
Admission Forms	0.7
Exam Attendance Sheets	0.7
Mark Sheet	1.7
Result Sheet	1.8
Transcript	0.9

Accuracy: The outputs of the programme were checked rigorously to ensure that they were accurate. Errors possible in manual calculations, under the previous system, have been eliminated.

Reliability: Numerous test runs were done at the Beta stage and fault correction done. The package was also used twice on actual data (initially on one batch and secondly on two batches simultaneously) and it generated the actual documents used for those examinations. Minor shortcomings were rectified. The final documents were thoroughly checked and found accurate.

Response of users and administrators: A formal evaluation of the product was not conducted. However, the response of all the users in the Examination Unit (a total of three persons) and the Dean of the Faculty to the product has been positive indicating its appropriateness and utility.

PRODUCT USE

Product usage: The product has been used to process the admission forms and attendance sheets of three sets of examinations, and has been used to process the marks of two sets of examinations.

Usefulness: This programme increased efficiency remarkably. For example, the entering and computation of mark sheets which earlier took a minimum of two weeks for a single exam, could now easily be done in two days, one day for entering marks plus another day for rechecking each individual entry. Similarly, the time taken for the other features, such as preparation of admission forms and examination attendance sheets, has also been reduced notably.

DATA SECURITY

At present the security of the system is limited. Physical access to the computer is restricted, logging on to the computer is through a password, only authorized users with their individual passwords can use the programme, and accessing the EdRex programme is through another password. To obtain the final grade sheet the Assistant Registrar, or his nominee, must check and click on "Checked". A CD Writer has been installed for record

maintenance. It is proposed to use non-rewritable CD's for storing the data of passed out students, which will be used for the processing of transcripts. This measure would make it difficult to tamper with the data.

User groups have been defined. However, at present differential access to the different options, based on the category of the user such as clerk or AR, has not been given to the different groups. This should be done in the future.

At present there is no system to log attempted security violations. This feature should also be incorporated in the future.

To safeguard against damage to data and enable data recovery in cases of data loss, backup copies are retained on floppy diskettes and a hard disk on another computer.

PRODUCT SUPPORT

User support: The response time to difficulties of the users was relatively quick with attention provided in about half a day (range of 30 mts. to 2 days). Software support was provided by the team leader and hardware support by another team member.

Training: On site training, using actual data for actual examinations has been provided to the staff responsible for examination matters.

System maintenance: Maintenance of the system has been mainly by the team leader since continual enhancements are being done. It will be handed over to the other team members in the future.

Documentation: Documentation of the product needs to be improved. At present the relationships between different objects within a database have been documented. An example is seen in Fig. 7, which shows the relationship between objects in the 'Results' database. This flowchart may not be conventional. However, when troubleshooting, this flowchart is useful in determining the relationship between different objects in an Access* database.

The steps to be followed for the different procedures have also been documented for the users.

COST

The cost incurred for development was only in time. Although no exact records were maintained, it is estimated that at least 1500 computer hours would have been expended on this project.
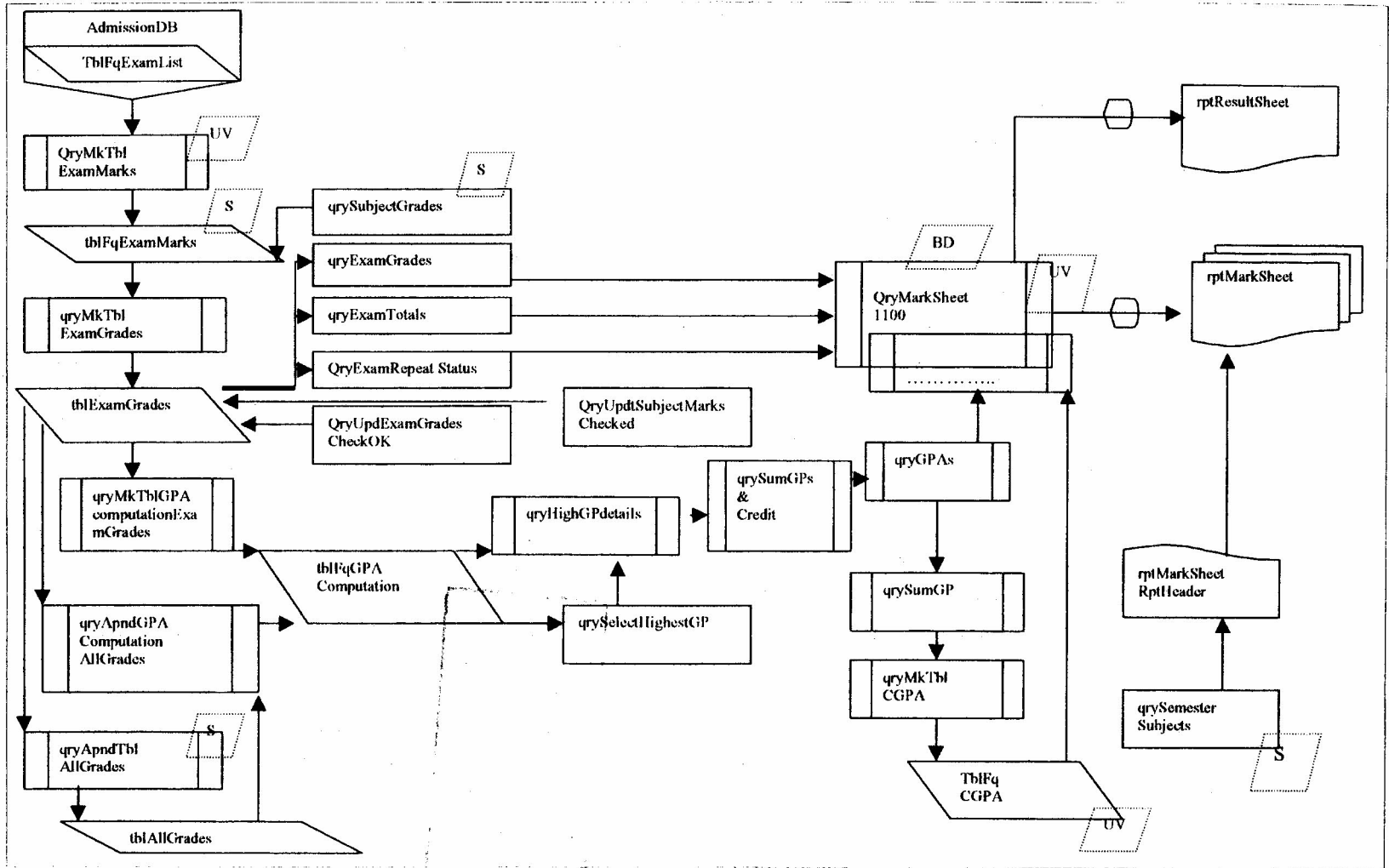
Fig. 7. Relationship between objects in database "Results"

RECOMMENDATIONS

Since any information system should evolve with time, "EdRex" too will be continually refined and developed. Recommendations for the future are as follows:

a) Additional features such as the determination of scholarship winners should be incorporated into the system.

b) Automating marks entry either through the provision of a diskette by the Departments, or through Optical Character Recognition, should be enabled to reduce entry time, and, more importantly, to eliminate data entry errors.

c) Automating many procedures in the Registration Unit should also be done.

d) Better documentation of the system is needed to assist trouble shooting and upgrading in the future, especially by another developer.

e) Files in the programme should be hidden and user access should be limited to prevent accidental modification of the programme and data.

f) Security features should be enhanced as indicated previously.

g) Periodic housekeeping runs should be incorporated to maintain the system.

Possibilities for the modification and further development of the product to make it applicable to other educational entities should be determined.

CONCLUSION

The product that has been developed through this study is currently being used to enhance the efficiency of the Examinations Unit of the Faculty. It has been rigorously tested, shortcomings rectified, improvements made, and security features incorporated. Recommendations have been given to improve this product further. The successful development of this product clearly indicates the potential for End User Computing, and hence the possibilities for non-computer science personnel to tap into the vast potential that the Information Technology (IT) sector has enabled.

ACKNOWLEDGEMENT

REFERENCES

1. Johnson R. and S. Johnson 1997. *An Information System for the Open University of Sri Lanka*. Infostat Consulting. UK. (Unpublished.)

2. Litwin, P.K. and M.G. Getz 1997. *Access 97 Developer's Handbook*, 3rd Ed. BPB Publications, New Delhi

3. Microsoft Corporation. 1996. Microsoft Access 97 (Software). Help Files. USA

Prague, C.N., W.C. Amo, and J.D. Foxall 1996. *Access for Windows 95 Secrets*. Comdex Computer Publishing, New Delhi.

Simpson, A., and E. Olson 1997. *Mastering™ Access® 97*. 4th Ed. BPB Publications, New Delhi.