UNIVERSITY OF ZIMBABWE
DEPARTMENT OF COMPUTING SCIENCE

A STUDY INTO THE IMPLEMENTATION OF A MANAGEMENT INFORMATION SYSTEM FOR ZIMBABWEAN HIGH SCHOOLS

BY

TINOVIMBA GERALD MOTSI

UNDER THE SUPERVISION OF

DR G. T. HAPANYENGWI

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS OF THE BACHELOR OF BUSINESS STUDIES AND COMPUTING SCIENCE DEGREE PROGRAM UNDER COURSE CODE CT360

JUNE 2004

Terms of Reference

This project has been duly authorised for submission in partial fulfilment of the requirements of the Bachelor of Business Studies and Computing Science (Honours) degree program under course code CT360. The Permanent Secretary of the Ministry of Education, Sports and Culture granted the ministry's head office research clearance on Wednesday the 22nd of October 2003. The Regional Director (Harare Province) also cleared the research on Thursday the 23rd of October 2003. Please refer to the Appendix B for the letter of clearance.

Preliminary Vision

Most Zimbabwean schools have invested quite extensively in information technology hardware of the last decade. The schools however continue to use either manual or computerised systems adopted from other industries. Preliminary research into the fees processing system of high schools has indicated that there exists a potential to unlock value within these schools through the development of an integrated and open source system template that maybe customised for each school.

The need for integration seeks to free up the resources that are currently being reserved for the manual transfer of data from one system to another. It is envisioned that this freeing up of resources would result in the system users being left with more time for innovative and qualitative tasks.

An open system is being suggested because of the recognition that most schools do not have the financial resources to dedicate towards more computerisation because of the harsh economic climate that they are facing at the moment. An open system would also facilitate rapid system improvement through the identification and resolution of system inadequacies since it would be open to scrutiny from academic and industry based personnel. The system would also provide an opportunity for the academic community to contribute towards the development of Zimbabwe.

A customisable, as opposed to a generic or bespoke, model is being suggested because of the need to accommodate the subtle differences that would make it difficult to implement a single generic model. In addition, a bespoke system for each school would result in an uneconomic reengineering of common components because of the large span of common operations among the schools.

Preface

The documentation for this project has been organised into five major sections. The first of these, entitled Requirements Analysis, documents the process that led to the formulation of the functional and non-functional requirements of the system. It starts by stating the rationale of the study and then moves on to describe the methodology used in studying the current system. A major subsection is then devoted to a description of the current system with the section ending with the listing of the requirements.

The Requirements Analysis section is followed by a section on design. The design starts with a brief overview of the design philosophy that guided the design process and the decision-making processes that determined the choice of the development environment. Being a data heavy application, the overview is followed by a subsection on the modelling of the system database. In line with the method proposed by Somerville (1995), this is followed by Architectural design with the section concluding with a section on the design of all the identified components.

A section on the evaluation and review of the system follows the Design and serves as a conclusion to the developer's section of the documentation. The second-last section contains a user manual that contains detailed instructions on using the system. This section concludes the main body of the documentation. The last section is made of appendices that provide background material on the project.

Acknowledgements

Firstly, I would like to thank my family for their support over the course of this project. I am particularly grateful for their continued tolerance of the anti-social behaviour that projects such as this one tend to encourage. I would also like to extend my gratitude to Dr Hapanyengwi, my project supervisor, for the guidance that he has patiently given me over the last two years.

My gratitude also goes to Mrs Musiyanduka the Chairperson of the Department of Computing Science for her assistance in arranging the documentation for the Ministry of Education applications. I would also like to thank Mr Sidimeli, Mr Chinyama, Mrs Kabasa, Ms Chokumanja and Mr Mukwandara of the same department for their assistance over the course of the project. I would also like to extend my gratitude to the late Mr Chinamasa formerly of the Ministry of Education, Sport and Culture. I would not have gotten very far without his advice.

I would also like to thank the staff of Mt Pleasant High School, Prince Edward High School and St Georges for the time they took out of their busy schedules to assist me with my research. They went far much further than they had to.

References

The reader may find it useful to consult the following documents that played an integral role in shaping the course of this project.

Online Material
1. Clinick A; *VBScript or JScript*; Microsoft Developer Network; http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnclinic/html/vbsvjs.asp; 1998
2. Hapanyengwi GT, Mazhindu-Shumaba RK; *Project Planning and Implementation For the Computing Science Student*; **University of Zimbabwe Course Pages**; http://www.uz.ac.zw/science/cs/ct260/Project%20Planning%20and%20Development%20Book.DOC; 2003
3. Ivy State College; *The Dewey Decimal Classification System*; **Ivy State College**; http://www.in-map.net/vines/basicInfo/dewey/dewey.html; 1999
4. Johnston P; *The CHAP Login System*; **PAJ Home**; http://pajhome.org.uk/crypt/md5/chaplogin.html; 2003
5. Microsoft; *Issues Migrating from DAO/Jet to ADO/Jet*; **Microsoft Support**; http://support.microsoft.com/default.aspx?scid=http://support.microsoft.com:80/support/kb/articles/q225/0/48.asp&NoWebContent=1; 2003
6. Microsoft; *Microsoft Developer's Network*; **Microsoft**; www.msdn.microsoft.com; 2001
7. Microsoft; *Technical FAQ;* **Microsoft**; http://msdn.microsoft.com/vbasic/previous/vb6/techfaq.aspx; 2004
8. Motsi TG; *Librarius 2000 Plus*; **Geocities Homepages**; www.geocities.com/tgmotsi; 2001
9. Motsi TG; *Levy System*; **Geocities Homepages**; www.geocities.com/tgmotsi; 2001
10. MySQL AB, *About MySQL AB*, **MySQL**, http://www.mysql.com/company/, 2004
11. Pierre-Emmanuel Gross; *RTF Printing Routines*; **Code Guru Website**; http://www.codeguru.com/vb/openfaq/comments/159.shtml; 2003
12. State Library of Queensland; *Browne System – Country Lending Service Manual*; **State Library of Queensland**; http://www.slq.qld.gov.au/pub/clsman/daytoday.htm; 2004
13. Sun Microsystems; *Comparison of ASP and JSP*; **Sun Microsystems**; http://java.sun.com/products/jsp/jsp-asp.html; 2004
14. Sun Microsystems; Sun One Active sever Pages; **Sun Microsystems**; http://wwws.sun.com/software/download/products/3f4e6fe6.html; 2004

Printed Material
1. Somerville Ian; *Software Engineering*; **Addison Wesley**; 1995
2. Sceppa David; *Programming ADO*; **Microsoft Press**; 2000
3. Balena Francesco; *Programming Microsoft Visual Basic 6.0*; **Microsoft Press**; 1999
4. Hethman PS; *Illustrated Guide to HTTP*; **Manning**; 1997
5. Orfali R; *Client Server Programming with Java and Corba*; **Wiley**; 1997
6. Gates Bill; *Business @ The Speed of Thought*; **Penguin**; 1999
7. Raggett D; *Raggett on HTML 4*; **Addison Wesley**; 1999
8. Ladd Eric, O'Donnell Jim, et al; *Using HTML 4.0, Java 1.1 and JavaScript 1.2*; **Que**; 1998

Website References
1. The University of Zimbabwe Open Public Access Catalogue; www.uz.ac.zw

"Traveller, if there is no path to your goal;
Make your own path step by step"

The Riddle of Spartakus

Contents

Table of Figures

REQUIREMENTS ANALYSIS

Methodology

The research will be by means of interviews with personnel from Prince Edward High School and St George's College. These two schools represent the two predominant types of schools in Zimbabwe that is, a government and a privately run school. The following is the proposed list of interview questions:

1. **Enrolment of Students**
   a. May you please describe the enrolment process with particular focus on
      i. Information collected.
      ii. How this information is processed and stored.
      iii. The additions and modifications made to the information stored in 1.a.ii over the course of the student's enrolment at the school.
      iv. Documentation and/or computer facilities used in facilitating 1.a.i & 1.a.ii.
   b. What problems have you experienced in managing the process detailed in 1.a
   c. If you were to specify the requirements for a computer system to aid the solving of the problems identified in 1.b, what would you include?
   d. Of the information stored by your schools with respect to each enrolled student, what would you want to be directly available to the student's guardian through a web interface?
   e. Of the information referred to in 1.d, what would you wish a parent to be able to modify directly through a web interface.

2. **Academic Registry**
   a. What courses do you offer?
   b. What constraints do you place on the courses that any given student can enrol for?
   c. What information do you store with regards to the courses on offer and the students enrolled for such courses?
   d. What is the assessment criterion for each group of courses?
   e. What aspects of the results of the assessment referred to in 2.d are made available to guardians? What documentation is used in facilitating this?
   f. What problems have you experienced in
      i. Timetable preparation
      ii. Assessment
      iii. Communication of assessment results
      iv. Storage of information
      v. Any other aspects of the registry
   g. How would you employ a computer system to aid the problems identified in 2.f

2

3. Fees Management
    a. Types of fees charged
          i. What fees do you charge?
          ii. To whom do these fees apply?
          iii. Who is responsible for which fees?
          iv. Who gives the directive to charge any given student a given fee?
          v. What documentation if any is used in (3.a.iii) above?
          vi. What documentation used in communicating the fees charged against any given student to that student?

    b. Storage of information
          i. What specific transaction records do you keep?
          ii. What documentation is used in this storage?
          iii. For how long are these records kept?
          iv. What happens to the records when their period of use has expired?

    c. Uses of information
          i. To what specific uses do you put the above-specified information?
          ii. Who are the consumers of this information and what specific aspects of it do they require?
          iii. What problems, if any, have you experienced in storing or accessing this information?
          iv. What opportunities for further utilization of the information would you want explored?

4. Libraries
    a. What library facilities does your institution provide?
    b. What classification for books have you adopted?
    c. What classification have you adopted for borrowers?
    d. May you please describe the circulation process?
    e. What restrictions are placed upon the operations that can be carried out by the personnel in the library?
    f. What links exist between the library and the other subsystems within the school?
    g. What problems have you experienced in managing your libraries?

5. Event Scheduling
    a. How do you schedule extra-curriculum activities?
    b. Who has access to the schedules referred to 5.a?
    c. What problems have you experienced in the scheduling of events?

6. **Boarding Facilities**
    a. What boarding facilities do you provide?
    b. Describe the process used in allocating students to the different halls of residence.
    c. What information is generated and consumed by the process detailed in 6.b?
    d. What documentation is used in 6.b?
    e. What problems have you experienced in managing the information generated by the accommodation system?

Case Studies

Prince Edward High School

Prince Edward High School is a government school, situated in Harare, catering for about one thousand two hundred boys. It provides both boarding and day schooling facilities.

Enrolment Process

Enrolment for Form One is continuous throughout the year. Each week, the enrolment secretary receives application letters that she forwards to the Headmaster for consideration. The school receives in excess of two thousand applications each year for the two hundred Form One places on offer. Siblings and descendents of former students receive automatic entry into the school. The other applications are accepted at the discretion of the headmaster subject to the applicant residing within the acceptance zone for Prince Edward High School as defined by the Ministry of Education, Sport and Culture.

All the application letters received are filed by the enrolment secretary after they have been considered by the headmaster regardless of whether they will have been successful or not. Each of the applicants receives notification of the state of their application at this stage. The school then sends out starter packs to all the successful applicants in the December preceding the start of the academic year. Each starter pack contains general information about the school along with a questionnaire that is used to create an enrolment record for the student. The enrolment record is maintained in a customised Microsoft Access database with a Visual Basic front end. At the beginning of each term, administrative personnel each receive a hard copy of the enrolment record, which is termed a Nominal Roll. The Nominal Roll is the primary reference point for Guardian contact details. Access to the nominal role is tightly controlled due to the private nature of the contact details it contains.

The Academic Registry

Like most other Zimbabwean schools, Prince Edward High School employs a streaming system. The range of subjects offered to any given student is primarily depended on the stream the student will be enrolled in. The top stream registers for 10 subjects including two practical subjects targeted at a particular application area such as accounting or woodwork. The class' form teacher and the relevant subject teachers maintain these details. Students are re-streamed in forms one and two.

The school prepares and sends out four assessment reports per student per term. A copy of each report is also filed with the academic registry. The first three are basic progress reports and they detail each subject teacher's opinion on the progress the student will be making and the steps that may be taken by the guardian to improve or consolidate this. The fourth report is a major assessment report. It is based on performance in an examination and general performance over the course of the term. A copy of the major report is shown as Figure 1 on the next page.

PRINCE EDWARD SCHOOL HARARE ZIMBABWE

Form .... Name .... Term .... 200......
Age ....(yrs) ....(mths) (at 31st Dec.)   Form Average Age ....(yrs) ....(mths) Attendance .... / ....
Position .... / ....   Average % .... Form Average % ....   Position Last Exam .... /........

SUBJECT	MARKS (%)	CLASS (AVE)	POS'N IN CLASS	EFFORT SYMBOL	REMARKS	TEACHER'S INITIALS
ENGLISH LANGUAGE						
MATHEMATICS						
GEOGRAPHY						
SCIENCE						
HISTORY						
AGRICULTURE SHONA BIOLOGY						
COMPUTING RELIGIOUS STUDIES						
FRENCH MUSIC						
ART LATIN TECHNICAL / GRAPHICS						
WOODWORK COMMERCE METALWORK						

Physical Education: ....
Sports: ....
....
Societies: ....
Clubs: ....
....
Form Teacher's Remarks: ....
....
....
House Master's Remarks: (   )..
Headmaster's Remarks ....
....
....

EFFORT KEY (A) Very Good   (B) Good   (C) Average   (D) Poor   (E) No Effort   (F) Definitely Failure at this rate
Boarders should be in residence not later than 5.30p.m. on the previous day. A HEALTH CERTIFICATE must b e provided by a doctor if pupil has been in contact with any of the prescribed communicable diseases.

Next Term begins at 7.30a.m on Tuesday ....200......

Figure 1: PE Major Assessment Report

6

Examinations at Prince Edward High School generally fall into two broad categories i.e. internal and external examinations. The external examinations are sat for at the end of forms four and six as is typical in Zimbabwe. These exams are scheduled by an external examining body with the school only providing invigilators. Internal examinations on the other hand are fully administered by the school. Each head of department is required to provide a list of papers he/she will be wishing to have the students examined in. Each scheduling request includes the following details with regards to each paper

1. A unique identification code for the paper.
2. Title of the subject.
3. The academic year of the students who sit for the paper.
4. Duration.
5. Preferred Date, time and venue.
6. Number of students enrolled for the subject.

The examination scheduling requests are then forwarded to the Deputy Head who then prepares the final examination timetable after resolving any clashes and adding a list of invigilators for each paper. The timetable is then circulated to students, teachers and administrative personnel. Please refer to Figure 2 to Figure 4 on the following three pages for extracts from the November 2003 schedule.

Financial Management
The accounting system at Prince Edward High School is not integrated with the rest of the administrative system. Students' details are generally communicated from the registry to the accounting department by means of the Nominal Roll.

The accounting function is broadly split into two i.e. the Bursar and School Development Association (SDA) sections. The Bursar's office handles the government gazetted fees and runs a manual system. The SDA system on the other hand runs three accounting systems i.e.

1. QuickBooks Nominal Ledger
2. QuickBooks Debtors
3. Payplus Payroll

Invoices are generated by the QuickBooks Debtors system with receipts being written out manually and being posted into the same system. Creditors and assets are handled through the QuickBooks Nominal Ledger.

Guardians are regarded as ordinary debtors within the accounting function. Fees due are debited to each account through categorised billing. The categories basically split students into boarders and day scholars, as this is the only determinant of the fees payable by any given guardian. Fees invoices are then generated by the system and forwarded to the administrative staff for inclusion with the rest of the end of term circulars sent to parents. Payments are due before the beginning of each term and are posted into both the nominal ledger and debtors' systems.

PRINCE EDWARD SCHOOL

EXAMINATIONS NOVEMBER 2003

DAY	TIME	LEV	CODE	SUBJECT	DUR	NO	VENU	INVIGILATORS
Thurs	8.00-11.00	O	7112/1	Principles of Acc	3h	50	Hall	PXB PAC Rel AAC AAM
6-Nov	8.00-10.30	A	9189/5	Chemistry 5 (Pract)	2h 30	20	Dal	HAM TJC MBB TAR
Clash	12.15-1.45	A	9197/2	Accounting 2	1hr 30	11	Board	MAC Rel TAD
	2.00-4.00	O	3159/2	Shona 2	2h	38	Hall	MPR KMS CAS GTS
	2.00-3.30	A	9197/2	Accounting 2	1hr 30	60	Hall	Rel TET BGY
Fri								
7-Nov	8.00-10.00	O	5034/2	Agriculture 2	2h	77	Hall	CLH BAK PSG NSI DAJ
	8.00-9.00	O	5071/1	Chemistry 1	1h	21	Hall	Rel SAJ SAM
	9.00-10.30	O	5071/2	Chemistry 2	1hr 30	21	Hall	
	8.00-11.00	A	9153/3	Literature in English 3	3h	43	Hall	
Clash	10.00-12.00	O	5034/2	Agriculture 2	2h	2	Board	MBC Rel TJC
Clash	10.00-12.30	O	5071/1	Chemistry 1	1h	2	Board	
Clash		O	5071/2	Chemistry 2	1hr 30	3	Board	
Clash	11.45-1.45	O	5006/2	Integrated Science 2	2h	1	Board	TAD Rel RAG
	2.00-4.00	O	5006/2	Integrated Science 2	2h	176	Hall	ATZ RAZ CAZ RCA ETA
	2.00-4.30	A	9163/3	French	2h 30m	2	22	PXB MBB
								Rel TAB GPC PAC SAC
Mon	8.00-10.30	O	2042/1	Religious Studies 1 (	2h 30	27	Hall	PSG MCB AAC SBM
10-Nov	8.00-9.30	O	2043/2	Religious Studies 2 (	1h 30	1	Hall	Rel MBC RAG
	8.00-11.00	A	9164/2	Mathematics 2	3r	51	Hall	
	8.00-10.30	A	9145/2	Shona 2	2h 30	1	Hall	
	8.00-11.00	A	9191/1	Art 1	3h		Art	MTG GAM
Clash	9.30-12.30	A	9164/2	Mathematics 2	3r	2	Board	GPC Rel NAC
Clash	10.45-1.45	A	9154/1	Divinity 1	3h	1	Board	DAJ Rel MAK
	2.00-5.00	A	9154/1	Divinity 1	3h	30	Hall	SNG AAC
	2.00-4.00	A	9198/1	Chemistry 1	2h	20	Hall	Rel MAC
Tues	8.00-11.00	O	6035/1	Woodwork 1	3hr	34	TG, 20	SNC TAB RelMBC
11-Nov	8.00-10.30	A	9190/2	Biology 2	2hr 30	15	Hall	SAC Rel RCA
Clash	12.30-1.45	O	7103/1	Commerce 1	1hr 15	1	Board	PAC Rel SAC
	2.00-3.15	O	7103/1	Commerce 1	1hr 15	122	Hall	TJC MBC NAC MCC TAR
	2.00-4.00	O	7108/2	Commercial Studies 2	2h	92	Hall 2	CSD TAD
	2.00-4.00	AS	8007/2	Eng Lang + Comm 2	2h	179	1-6.	1 XXD 2 MBD 3 SNG 4 PRG 5 RAG 6 PSG
	2.00-3.15	O	60__/2	Music 2	1h15	1	Hall	IAJ
								REL TAG JAG CLH JRI
Wed	8.00-9.00	O	5006/3	Integrated Science 3	1h	176	Hall	BAK DAJ MAK SAM RFM 24 KDM
12-Nov	8.00-11.00	A	9156/2	Geography 2	3h	74	22	EBM KFM
								Rel KDM JAM PBM
	10.30-1.00	3		Latin	2h30			23 DEJ
	2.00-4.30	O	2013/1	Literature in English	2h30	72	Hall	NSI DAJ SAJ
	2.00-4.30	A	9163	French 4	2h 30	2		REL DEJ
	2.00-5.00	6I		Art	3h			ART
	2.00-3.00	3		Chemistry	1h			22 ATZ CAZ

Figure 2: PE Examinations' Time Table

SYM	INVIGILATOR	INVIGILATIONS EXTERNAL	INVIGILATIONS INTERNAL
ETA	AYISA E T O	27ACR 7P 13A AC	
RCA	ATKINSON RC Mrs	3A 7P 11AR 13A	19 A
PXB	BASOPO P	23A 6A 7P 13A	18 AP
MBB	BENZON M Mrs	23A 29P 3A 6A 7P 13A 14AC	19 AA
MCB	BONYONGWE M	10A 13A	17 AA 18 AP 19 AA PR
TAB	BWANYA T	29A 31A 3A 7PR 11A 13A 17A	
GPC	CHAKWENYA GP	23A 27AC 29P 5A 7PR 10AC 13A	19 A AR
PAC	CHAPMAN P	23AR 6A 7PR 11AC 13 AR	14 PRO 17PRO 18 A
SAC	CHIBAYA S	23AR 29P 7PR 11A ACR 13AR 17P	18 P
PNC	CHIBVURI P	27A 29AR 29P 5A 13AR 17A 19P	18 P
AAC	CHIKOBVU A	27A 29PR 3A 5A 6AR 10A P 14A	18 P 19 A AR
SNC	CHIKUNI SN	27A 29AC 3AR 11A 14A 17A 19PR	13 A
MAC	CHIMOMBE M	23A 3A 5AC 6AC 10PR 17A 10P	14 AP
MBC	CHINGODZA M	28A 3A 7AC 10AR 11AR 17AR	18 P 19 A AR
NAC	CHINODYARUSWA N	3ACR 5A 10A ACR 11P 14AR	19 A AR
MCC	CHOGA M MRS	29A PR 30P 5A 11P	18 A AP 19 A AR
TJC	CHOKUDA T	29AC 30P 5AC 6A 7ACR 11P 17AR	18 A
CSD	DU PLESSIS C S Miss	31P 5ACR 11P 14AC 18A 20P	18 P 19 AR A
TAD	DUMA T	31P 3A 6AC 7AC 11P 18A 20P	18 P 19 AR
XXD	DZUKAMANJA MRS	28AR 29A 31P 3A 11P 18A 20P	18 P 19 AR
MBD	DZUMA M	23A 31P 3A 5ACR 11P 18A 20PR	18 P
SNG	GARIKAI S N	29P 31P 11P 18A P	18 P
PRG	GONDONGWE B Mrs	28A 29A 31P 3AC 11P 18A	18 P
RAG	GONESO R	28A 31P 5A 7ACR 10AR 11P 18A	18 PR
PSG	GUMUNYU PS	31P 3AR 5A 7A 10A 11P 18AR	18 PR
TAG	GURURE T	31P 11PR 18AR	17 AA 18 PR
JAG	GWANGWADZAJ Mrs	31PR 11PR 19A	17 AP 18 AA
CLH	HILLS CL	29A 31PR 3AC 7A 11PR 19A	18 AA
JRI	IMBEAH J R	3P 11PR 18A 19A	17 AAP 18AR
NSI	IMBEAH NS MRS	31A 31PR 7A 12A 19A	14 P 18 AA
DAJ	JARDINE D	31AP 7A 10AC 12AP P 19A	18 AA
SAJ	JIRRI S	28AR 31AR 4PC 5AR 7AR 12P	18 AA
IAJ	JONES I Mrs	3P 11P 14ACR P 19AR	19 A
DEJ	JORDAAN D. Mrs.	3P 12PR 18AR	12A 13A 17 AAP
OAK	KACHEPA O MRS	28AR 5AR 14P 19AR	17 AAP 18 AA
DAK	KARANI D	27A 29AR 3AR 18P	17 AAP 18 AA
BAK	KARUMAZONDO B	29A 3P 4AR 7A 12A 14P	18 AA
SAK	KUFAKWEDEKE S		14AA 17AAP 18A 19AA
MAK	KUJINGA M Mrs	3P 10ACR 12A 18P 20A	18 AA
MAM	MACHERA MRS	3P 14A ACR 18P	17 AAP 18 AA
SBM	MACHOMERA S Mrs	3AR 10A 14A 18PR	17 AAP 18 AA
SAM	MAGWEGWE S	29AR 3P 4AR 5AR 7AR 12A 18PR	18A
PBM	MAINGEHAMA P	3A P 12AR 18P	13 A 19 AAP
CXM	MAJONGWE MRS	20A	13 A 14 AA 17 AAP 18 AA
EEM	MAPIYE E	3PR 18PCR	13 AP 14 AA PR 17 A AP 18 AR A
HAM	MARUFU H	3PR 6A 19P 20A	13 P 14 AA PR 17 AAP 18 A
TJM	MATIBIRI TJ	29A 3PR 20A 21P	13 ARP 14 AA PR 17 AAP

Figure 3: PE Examinations Invigilators List Pg1

KFM	MAWIRE KF	29A 4P 11A 18PC 20A	13 AAP 14 AA 18 ARA
PMM	MBIRI P Mrs		14AAP 17AAP 18AA
ECM	MORGAN MRS	29AR 4P 20A 21P	13 AAP 14 AA
EBM	MUBAYIWA E Ms	29A 4PR 11P 20AR	13 AAP 14 AA
RAM	MUCHINARIPI R	29 29AC 30AC 5P	13 AAP 14 AA 18 ARA
EAM	MUCHIWANGA E	30A 5P 20AR	13 AAP 14 AA 18 AA
KDM	MUKWASI K	30AC 4PCR 5P 12A 17AR	13 AAP 14 AA
KAM	MUJOMBI K		13AA 14AAPR 17AAP
JAM	MUNATSI J	30AC 4A 5P 12AR 17PR	13 AAP 14 AA
RFM	MUNYANDA R Ms	5P 12A 20AR	13 AAP 14 AA 17 AAP
NBM	MUPARURI NB	29A 5P	13 AAP 14AA 17 AAP 19 A
CAM	MUSITU C	5P	13 AAP 14 AA 17 AA PR
AAM	MUZAH A	27AR 30A 3A 4A 5PR 6AR	18AR 19 AAP
EAN	NYAMAYARO E	30AR 3A 4A 5PR 12ACR 21A	18AR 19 A
GSP	PHIRI GS Mrs	4A 5PR	13 AAP 14 A 17 AA PR
TAR	REGO T	29P 6A 11P	13 AAP 14 A 17 AA PR
MPR	ROBINSON MP Miss	4A 6P	13 AA PR 14 AA
BLS	SABADA BL	4A 20AR	13 AA PR 14 AA 17 ARA PR
THS	SAICH D S Mrs.	4A	19 A 17 AR
KMS	SAICH K M	6P	13 AA PR 14 ARA 17 AR AR PR
LLS	SIBANDA L Ms	14ACR	13 AA 14 AR 17 AR AR 19 AA
LAS	SHEERS L Mrs	29AC 13AC 14AC	13 PRO 14 AR 17 AR 18 PRO 19 PI
CAS	SITOTOMBE C Mrs	6P	13 AA 14 AR AR 17 AR 19 AAP
GTS	SUNGURO GT	6P	13 AA 14 AR AR 19 AAP
TET	TIRIBABI T Miss	6PR	13 AA 14 AR 17 AP 19 AAP
BGY	YOCKNEY BG	6PR	13 AR AR 14 AR 19 AAP
RAZ	ZIKO R	7P	13 AR AR 18AR 19 AAP
ATZ	ZIVAVE AT	31AC 4A 7P	12P 13 AR AR 18AR 19 AAP
CAZ	ZVAYI C	31ACR 4AR 7P	12P 13 AR AR

KEY   A =MORNING, P = AFTERNOON, R = RELIEF, C = CLASH

e.g.   19A ACR P indicates that the invigilator has on the 19th a morning
invigilation followed by another morning session of relief invigilation to clash
candidates, and then a further invigilation in the afternoon.

PLEASE CHECK BOTH THE TIMETABLE AND THE SUMMARY. IF YOU DO SWOP
INVIGILATIONS PLEASE LET DHM KNOW(IN WRITING) AND ALSO MAKE THE
AMMENDMENT ON THE TIMETABLE IN THE QUIET MARKING ROOM.

EXAMINATION BASES

1A1	1	2A1	8	3A1	17
1A2	2	2A2	9	3A2	18
1B1	3	2B1	10	3B1	19
1B2	4	2B2	11	3B2	34
1B3	5	2B3	14	3B3	35
1B4	6	2B4	15	3B4	36
1B5	7	2B5	16		

Figure 4: PE Examinations Invigilators List Pg2

Library Facilities
The school has four libraries comprising the main library, a sixth form reference library, a computer & music library and an archival library. Of these, only the main library is a lending library. The computer and music library contains multimedia catalogues whilst the archival library contains records of all students and personnel who have enrolled or worked at the school since its founding. The combined catalogue of the libraries is in excess of sixteen thousand books. Each of the libraries also has access to online journals and examination support materials through connections to the Internet.

All the libraries with the exception of the multimedia library use the Dewey Decimal Classification (DDC) system for cataloguing. Catalogues in the music and computing multimedia library are maintained on a computer system called CDI-SIS, which has its own classification system. The main library uses the Browne System for managing its circulation which comprises short-term loan and long-term loan books. Short-term loans are for one week whilst long-term loans are for two weeks. Each subscriber is allowed to borrow a maximum of one book at a time. Details of the DDC and Browne systems are given in the appendices.

Each library book record contains the following details
- A unique accession number.
- Author
- Title
- Edition
- Year of Publication
- ISSN for periodicals and ISBN for books.
- Date acquired
- Condition when acquired [Old | New]

Each subscription record contains the name of the subscriber and contact details, which are primarily the class of the student, department of the employee or address of the alumni member. The library is open to all members of staff, members of the Prince Edward High School alumni, students and authorised researchers.

Event Scheduling
Prince Edward High School is actively involved in extra-curricula activities with each student being required to formally register for at least one sport and one society. Provincial and national activities for each academic year are scheduled at the end of the previous academic year. The scheduling procedure involves patrons of each sport or society meeting with their colleagues from other schools and then deciding on a calendar of events. The schedule is then communicated to the administrative office, which then prepares a consolidated calendar for all members of staff and guardians. A copy of the calendar is also made available on the school's website. Please refer to Figure 5 to Figure 7 for extracts from the 2003 calendar.

PRINCE EDWARD SCHOOL

CALENDAR

3RD TERM 2003

SEPTEMBER

Mon	8	HOD's Meeting in Boardroom at 9.15am
		Full Staff Meeting in Lecture Theatre at 10.15am
		Housemasters' Meeting in Lecture Theatre at 12.00pm
		Boarding Housemasters' Meeting in LT at 12.30pm
		Boarders return by 5.30pm
Tue	9	School Opens with Assembly at 7.40am
		First 4 periods are form periods
		Lessons begin from period 5
		Music & Computer Club Meeting – 10am
		SDA Exco Meeting in Boardroom at 5.30pm
		PEFAC General Meeting in Tennis Pavilion at 5.30pm
Wed	10	Prefect's Meeting in Lecture Theatre at 10.15am
Thu	11	Music & Computer Club Meeting – 10am
		Karate – Gradings at OH 5pm
		PTA Exec Meeting in Boardroom at 4.30pm
		PTA General Meeting in Pavilion at 5.30pm
Fri	12	Badminton vs Marondera High – Away – 12pm
		Waterpolo vs St Johns Juniors – Home
		Basketball vs Peterhouse Away – All teams
		Karate – Gradings at OH at 5pm
		Tennis vs Peterhouse – 1st, U16A – Home
		Tennis vs Peterhoiuse – U15A, u14A – Away
		Maintenance Meeting 8.30am HM's Office
Sat	13	Cricket vs Churchill 1,16A,15A,14A Home
		2nd,16B,15B,14B Away
		Waterpolo vs Peterhouse Snrs – Home
		Karate – Gradings at OH 9am - 6pm
Sun	14	Chapel Service Fr Chifunyise 6.30pm
		Squash Mashonaland League - HSC
		Karate – Gradings at OH 9am – 10am
		Cricket – Northwinds 1 vs HSC 2
		Cricket – Uprising 2 vs Shamva
		Cricket – Universals 3 vs Northwinds 2
Mon	15	Special Report Begins until 10th November
		Environmental Protection Club in LT at 3pm
		UNESCO meeting Rm 17 - 1pm
		Squash PE 'D' vs Arundel 'B' – Away 3.30pm
		SDA Gen Meeting BR 5.30pm
Tue	16	Inter-House Shona Quiz in Beit Hall at 2pm
		Music & Computer Club Meeting – 10am
		Jnr Debate vs Chisipite – Away – 3pm
		Snr Debate vs Chisipite – Away – 6pm
		EPC Lecture with Guest Speaker in LT at 2pm
		Table-Tennis vs AW & ER Home 2pm
		Squash PE 'D' vs Peterhouse Girls 'A' 2pm – Away
		Squash PE 'C' vs Peterhouse Boys 'B' 2pm – Away
		Bridge Jnr vs P'House A & B – Away – 2.30pm
		PEFAC Meeting in Tennis Pavilion at 5.30pm
		Headmasters/Bursars Meeting in H/M Study 9.15am
Wed	17	Senior Prefects Meeting in HM's Study at 10.15am
		Badminton vs Marondera High – Beit Hall – 2pm
		UNESCO Meeting with HQ delegates in LT at 2pm
		Current Affairs vs Q.E. 2pm – Away
		Table-Tennis vs Cranborne – Away 2pm

		Squash PE A & B vs Chisipite A & B (Mixed Dbls) – Away
		Squash PE C & D vs Gateway A & C – Home
		Inter-House Waterpolo – Pool – 2pm
		SRC Meeting in L/T at 1pm '
Thu	18	Records of Work to HOD's
		YFC – Visit to Kutsaga 10am
		Music & Computer Club Meeting – 10am
		Inter-House Waterpolo – Pool – 2pm
		ISLB Meeting LT 2.30pm
Fri	19	Records of Work to HM
		HOD Departmental Minutes to HM
		Waterpolo vs P'House & L'Gundi Juniors – Away
		Chess PE A vs Highfield – Away 2pm
		Chess PE B vs St Johns High – Away 2pm
		Basketball vs Churchill – home - All teams
		Tennis vs Chisipite A – 1st, U16A – Away
		Tennis vs Chisipite A – U15A, U14A – Home
		Maintenance Meeting 8.30am HM's Office
Sat	20	Cricket vs Eaglesvale 1st,16A,15A,14A Away
		2nd,3rd,16B,15B,14B Home
		Waterpolo vs St Johns & P'house Snrs- Away
		Squash – Veterans vs Schools HSC 8am
		Swimming - Individual Gala – PE - 2.30pm
Sun	21	Chapel Service Pastor P. McCoun 6.30pm
		Karate vs Chinhoyi - Away (M'reign)
		National/Zim Schools Rowing Regatta – Vic Falls
		Cricket – Northwinds 2 vs Marlborough
Mon	22	English, Agriculture & Careers Exercise Books to HM
		Special Report to be handed in
		Environmental Protection Club in LT at 3pm
		UNESCO meeting Rm 17 - 1pm
		Inter-House Cricket Seniors Day One
Tue	23	HM to visit Archives at 9.30am
		Full Staff Meeting in Staff Room at 10.15am
		Inter-House Cricket Juniors Day One
		Music & Computer Club Meeting – 10am
		Jnr Debate vs GHS Venue TBA
		Squash PE 'D' vs Chisipite 'D' – Home 2pm
		Bridge Jnr vs Chisipite A & B – Home – 2.00pm
		NASH Meeting in Lecture Theatre at 2.15pm
		SDA Exco Meeting in BR at 5.30pm
Wed	24	Prefect's Meeting in Lecture Theatre at 10.15am
		Snr Debate vs Cranborne – Pavilion at 2.15pm
		PE Public Speaking Semi-Final Round in Auditorium at 7pm
		Zimbabwe Astronomical Society Meeting in LT at 5.30pm
		Badminton vs Ellis Robins – Away – 1pm
		Shona Dramatics vs E.R. 2pm – Away
		Table-Tennis vs Mazowe Away – 2pm
		Squash PE 'B' vs St Georges 'A' – Away 2pm
		Music – Lunchtime concert 1 – 2pm (TBC)
		Volleyball vs Peterhouse – Home - 2pm
Thu	25	Music & Computer Club Meeting – 10am
		HM to visit Agric Plot at 8.45am
		SSF Meeting in HM's Study at 11.00am
		Careers – Traffic Safety VID Eastlea Learners License 2pm
		JCC Meeting in BR at 5.30pm
		ISLB Visit Casino 9am

Figure 5: PE Calendar Pg1

SPORT AND CULTURAL FIXTURES
3RD TERM 2001

BADMINTON

Master in Charge : Mr J. Imbeah

Fixtures

Date	School	Venue	Time
12.09.03	Vs Marondera	Marondera	12pm
17.09.03	Vs Marondera	BH	2pm
24.09.03	Vs Ellis Robins	Ellis Robins	1pm
27.09.03	Vs St Georges	St Georges	8am
01.10.03	Vs Ellis Robins	Ellis Robins	1pm
06.10.03	Vs St Georges	St Georges	1pm
11.10.03	PE Tournament	BH	8am
22.10.03	Vs St Georges	BH	2pm
01.11.03	Mazowe Tournament	Mazowe	8am
15.11.03	Ellis Robins Tournament	Ellis Robins	8am

BASKETBALL

Master in Charge : Mr Basopo
Coach : Mr Garura

Practice Times

Team	Staff	Days	Time	Venue
1st	Mr R. Gurure	Tue/Thur	3-5pm	Gym
2nd	Mr P Basopo	Tue/Thur	3-5pm	Gym
U 16	Mr R. Madzudzo	Tue/Thur	3-5pm	Outside Crt
U15	Mr D. Jardine	Mon/Wed	3-5pm	Centre Crt
U14	Mr T. Mawire	Mon/Wed	3-5pm	Outside Crt

Fixtures

Date	School	Venue	Time
12.09.03	Vs Peterhouse All teams	Peterhouse	
19.09.03	Vs Churchill All teams	Home	
27.09.03	PE U20 Invitational	PE Gym	
03.10.03	Vs St Georges All teams	Home	
10.10.03	Vs Watershed All teams	Watershed	
11.10.03	U14 Tournament Invitational	Eaglesvale	
15.10.03	Vs St Johns Coll All teams	Home	
24.10.03	Vs Eaglesvale All teams	Eaglesvale	

CHESS

Teacher in Charge : Mr E. Maunze

Date	School	Venue	Time
19.09.03	PE A vs Highfield	Highfield	2pm
19.09.03	PE B vs Allan Wiilson	QE	8am – 5pm
27.09.03	PE A vs Mufakose 2	QE	8am-5pm
27.09.03	PE A vs Mabvku	QE	8am-5pm
27.09.03	PE B vs Allan Wilson	QE	8am-5pm
27.09.03	PE B vs Kuwandzana	QE	8am-5pm

CRICKET

Master in Charge : Mr K. Saich

Practice Times

Team	Staff	Days	Time	Venue
1st	Mr Matabiri	Tue/Thur	2.30 – 5.00	Jubilee
2nd	Mr Barker	Tue/Thur	2.30 – 5.00	Jubilee
3rd	Mr Yockney	Tue/Thur	2.30 – 5.00	Jubilee
16A	Mr Saich	Tue/Thur	2.30 – 5.00	Jubilee
16B	Mr Nyamayaro	Tue/Thur	2.30 – 5.00	Jubilee
15A	Mr Yeo	Mon/Wed	2.30 – 5.00	Jubilee
15B	Mr Chibvuri	Mon/Wed	2.30 – 5.00	Jubilee
14A	Mr Mkandiwa	Mon/Wed	2.30 – 5.00	Rhodes
14B	Mr Ndimutseyi	Mon/Wed	2.30 – 5.00	Rhodes
14C	Mr Mawire	Mon/Wed	2.30 – 5.00	Rhodes

Fixtures

Date	School	Venue	Time
13.09.03	Churchill 1.16A.15A.14A	Home	9am
	2nd.16B.15B.14B	Away	8.30am
20.09.03	Eaglesvale 1.16A.15A.14A	Away	9am
	2nd.16B,15B,14B	Home	8.30am
27.09.03	Jameson 1.16A.15A.14A	Away	8am
	2nd.16B.15B.14B	Home	8am
04.10.03	C.B.C. 1st.16A.15A.14A	Home	9am
	2nd.16B.15B.14B	Away	9am
11.10.03	St.Georges 1.16A.15A.14A	Home	9am
	2nd.16B.15B.14B	Away	8.30am
25.10.03	Hillcrest 1.16A.15A.14A	Home	9am
	2nd.16B.15B.14B	Away	9am

GOLF

Master in Charge : S. Virri
Coach :

Practice Times

Team	Staff	Days	Time	Venue
1st	Mr.Jirri	Mon	2pm	Royal Harare

Fixtures

Date	Venue		Time
28.09.03	Ruwa Golf Club	P'House to organise	TBA

KARATE

Master in Charge : Sensei Simba Chikutu

Date	L	Venue	Time
30.08.03	Jnrs & Snrs MKA Comp	TBA	TBA
11.09.03	Gradings	OH	5pm
12.09.03	Gradings	OH	5pm
13.09.03	Gradings	OH	9am-6pm
14.09.03	Gradings	OH	9am
21.09.03	Chinhoyi/Mabelreign	M'Reign	
27.09.03	Cadet & Childrens Comp	TBA	
27.09.03	OH/Mabelreign	TBA	
18.10.03	Byo/OH	TBA	

ROWING

Master in Charge : Mr N. Dhliwayo

Fixtures

Date	School	Venue	Time
21.09.03	National Schools	Victoria Falls	

SQUASH

Master in Charge : Mr M. Chimombe
Coach : Miss O'Shea

Practice Times

Team	Staff	Days	Time	Venue

Fixtures

Date	School	Venue	Time
14.09.03	Mashonaland League	HSC	TBA
15.09.03	PE 'D' vs Arundel 'B'	Arundel	3.30pm
16.09.03	PE 'D' vs P'Hse Girls 'A'	Peterhouse	2.00pm
16.09.03	PE 'C' vs P'Hse Boys 'B'	Peterhouse	2.00pm
17.09.03	PE A/B vs Chisipite A/B	Chisipite	3.00pm
17.09.03	PE C/D vs Gateway A/C	Home	2.00pm
20.09.03	Veterans vs Schools	HSC	8.00am
23.09.03	PE 'D' vs Chisipite 'D'	Home	2.00pm
24.09.03	PE 'B' vs St Georges 'A'	St Georges	2.00pm
26.09.03	Meeting for TIC's Squash	PE BR	2.00pm
27.09.03	Awards Presentation	HSC	8 – 12pm
28.09.03	Awards Presentation	HSC	8 – 12pm
30.09.03	PE 'B' vs St Johns 'A'	St Johns	2.30pm
01.10.03	PE 'A' vs St Georges 'A'	Home	2pm
15.10.03	PE 'C' vs Eaglesvale 'A'	Eaglesvale	2.30pm

Figure 6: PE Calendar Pg2

SRC MEETINGS

Date	School		Venue	Time
17.09.03	PE		LT	1Ppm
01.10.03	PE		LT	1pm
15.10.03	PE		LT	1pm
29.10.03	PE		LT	1pm
12.11.03	PE		LT	1pm
26.11.03	PE		LT	1pm

STAFF ALLOCATION FOR SPORTS CLUBS

Please would staff indicate at Assembly or Notice Boards times for activities and where boys should go for activity.

Accountancy Club	Mr Karumazondo
Aids Awareness	Mr Zivave
Air Cadets	Mr Yockney/Mr Atkinson
Aquarium	Mr Dhliwayo
Archives	Mrs Van Biljon
Art	Mr Gumba/Ms Maxim
Astronomy	Mr Barnes/Astrology Society
Audio Visual/ICT	Mr Zivave
Badminton	Mr Imbeah/Mr Musitu/Mr Jirri
Dance Sport	Mr Mubaiwa
Boys' Brigade	Mr Chimombe
Bridge	Miss Robinson
Calligraphy	Mr DeKune
Careers	Mr Zivave
Careers Lectures	Mr Barnes/Mr Zivave
Catering	Mrs Saich/Mr Barnes/Mr Tirivangani/PTA/Mrs Burle
Chapel/ Confirmation Class	Father Chifunyise/Mr Yockney/Mr & Mrs Morris ChaplainCounselling
Charity	Mrs Mubaiwa
Classics	Mrs Jordaan
Computers Seniors	Mr Basopo
Computers Juniors	Mr Zviko/Ms Sibanda
Chess	Mr Maunze/Mr Mukwasi/Ms DekaS.
Christian Council	Mrs Bertram/Fr Chifunyise
Craft/Design	Mr Chibvuri
Creative Writing	
Culture	Mr Jardine/Mr R. Solomon
Current Affairs	Ms Mawarire/Ms Deka/Mr Zvayi
Darts	Mr Zvayi / Mr Magwegwe
Debate – Junior	Mr Yockney/Ms Mukandidze
Debate – Senior	Ms N'Diaye/Ms Gondongwe
Diving	Mr Chibvuri
Dramatics/Musicals	Mrs Sheers/Mr Jardine/Mr Solomon
Economics Club	Mrs Sibanda
EPC	Mr Musitu
Enterprise Education	Mr Zivave
Examinations O&A	Mr Atkinson
Exams Analysis	Mrs Saich
Exchange Trips	Mrs Imbeah
Extra English	Mrs Sheers/Mrs Musasiwa
Freedom From Hunger	Mr Muchiwanga
First Aid	Ms MawaireMr Kufakwedeke/Mr Bwanya
French	Mrs Imbeah/Mrs Dzvukamanja
French Conversation/ Reunion Exchange	Mrs Imbeah/Mrs Dzvukamanja/Mr Karani/Mrs Tiribabi
French Resource Centre	French Embassy/Mr Escande/Mrs Imbeah
Geography Society	Mrs Gondongwe/Mr Musitu

Global Village	Mrs Gondongwe/Mr Musitu
Golf	Mr Gumunyu/Mr Jirri
Gymnastics	Mr Garakara (Coach) Mrs Morgan
ICDL Club	Mr Basopo/Mr Zviko Mr Sabada/Mr Songoro
History Club	Mr Muchiwanga
Hospitality	Mr Barnes/Mrs Saich//Mr Saich/Mrs Sheers/Mrs Burle
House Points	Mrs Saich
Human Conservation	Mr Zivave
Interact	Mr Satombo
Internet/Website	Mr Basopo
JAZ	Mr Karumuzondo/Mr Rego/Mr Chikobvu
Karate	Mr Zvayi/Mr Magwegwe
Senior Maths Olympiad	Mr Munatsi
Library – Junior	Mrs Tsveta/Miss Ravenscroft/Mr Masunda
Library – Senior	Mr Maunze/Mr Makombe
Magazine	Mrs Ncube/Mr Barnes/Mr Jirri
Maneco	Mrs Sibanda/Mr Chikobvu
Music/Computer	Mr Masonda
Natural History	Mr Atkinson
PEST	Club Members/Mr Solomon
Photography	
Printing	Mr Mutaramutswa/Andrew & George
Presidents' Award Scheme	Mr Maingehama/Mr Marufu
Public Speaking	Mr Barnes/Mr Jadine/Mr Solomon/Mr Peter
Quiz	Mr Chokuda/Mrs Saich i/c Interhouse Quiz
Quiz – French	/Mr Karani/Mrs Tiribabi/Mrs Imbeah
Radio Club (ZARS)	Mr Chibvuri/Mr Lobb
Rowing	Mr Dhliwayo/Mr Satombo – Coach
Science & Technology	Mr Dekune/Mr Chokuda/Mr Duma/Mr Madukani
Scrabble	Mr Magwegwe/Mr Zvayi
Speech & Drama	Mr Jardine/Mr Solomon
Scribe and Printing	Mr Dekune/Mr Solomon
SU/Christian	Mrs Bertram
Shona Drama/Quiz	Mr Muzah/Ms Deka/Ms Sitotombe/Shona Dept
Scouts	Mr Marufu
Shooting	Mr Atkinson
Stationery	Mr Ayisa
Signwriting	Art Dept
Snooker	Mr Dekune/Mr Muzah
Stamp Club	Miss du Plessis
SRC	Ms Munyanda
Sixth Form	Mr Barnes
Supervised Prep	Mr Chakwenya
Special Report	Mr Barnes
Special Needs	Mrs Sheers/Mrs Musasiwa
Testimonials (U6 Only)	Mr Barnes
TG Resource Centre (CADKEY)	Mr Chikuni Wed & Fri 1.30-2.30pm
Table Tennis	Mr Muringani/Mr Muparuri/Mr Kufakwedeke
Triathlon	Mr Gumunyu / PE
Transport Manager	Mr Ayisa
UNESCO	Mr Maingehama/Mr Madukani/Mr Marufu
Woodwork	Mr Kufakwedeke/Mr Chimombe
Wardrobe	Parents/Mrs Sheers
YFC	Mr Dhliwayo/Mrs Machera/Members

Figure 7: PE Calendar Pg3

Boarding Facilities
Boarding facilities at Prince Edward High School are offered to students ordinarily resident outside Harare. Exceptions are however often made at the discretion of the headmaster as to parental requests. For example, a single parent who travels frequently could apply for special consideration. The school has three halls of residence to which students are allocated randomly with the exception that an effort is made to ensure that siblings reside in the same hall. Records of boarding students are maintained as part of the Nominal Roll.

Desired Focus of Computerisation
Mr Barnes, the headmaster of Prince Edward High School, said he would desire a system that eliminated mechanical tasks. He gave as an example the amount of time spent copying marks and comments from a subject teacher's mark sheet to the report card. He however qualified his remarks by stating that he would not welcome any system that would eliminate the personal touch that the school gave to everything it did. For example, he would have problems with a system that would automatically generate comments through artificial intelligence.

In a separate discussion, Mr Postings, the School Development Association accountant expressed concerns about the amount of time spent manually posting items among the nominal ledger, debtors' ledger and nominal roll. He said he would wish to have an integrated accounting system that eliminated the mechanical tasks creating more time for qualitative financial control.

St George's College

St George's College is a private school for boys situated in Harare. Although it is affiliated to the Roman Catholic Church, its enrolment base is multi-denominational.

The Enrolment Process

Applications for enrolment into private schools in Zimbabwe are partially centralised. In the first instance, a prospective Form One student applies to several schools for consideration as a formal applicant. This is usually done between the months of January and May in the year preceding the one in which the student wishes to enrol as a Form One student. In May, the schools send out entry forms to students they wish to consider for enrolment. The student then makes up to three preliminary choices, ranks them in order of preference and then returns the forms to the respective schools.

In July, the student sits for an entrance exam at the school indicated as the primary choice. The examination is set by the Independent Schools Board and written on the same day countrywide. The schools then make their final choices as soon as the results are made available with each school being given the choice to enrol the student as to their ranking on the student's entry form.

Once the students have been chosen, St George's sends each guardian a Declaration of Acceptance, which is a legally binding document. It also sends out a registration questionnaire, a sample of which is shown as Figure 8 and Figure 9 below. The questionnaire, upon its return, is used to construct an enrolment record for the student, which is stored in a Microsoft Access database in the office of the Registrar.

Academic Registry

St George's College, like most schools in Zimbabwe, employs a streaming system. The system determines the subjects that are offered to any given student. The students are re-streamed in forms one and two. Each department maintains its own record of courses offered and the students enrolled for each course.

The school prepares and sends out two assessment reports per student per term. The first report is a mid term effort assessment report that gives an overview of the effort the student will be putting towards academic progress. The second report is a broader end of term assessment report that is based on the student's performance in an examination and the progress he will have made over the course of the term. This report also includes a record of the position the student will have attained in each subject. A summary of the report containing a record of the student's marks in each course is also filed with the registry. This record is used to manually generate academic transcripts for the students on the completion of their studies or earlier withdrawal from the school.

Form B/2004

ST GEORGE'S COLLEGE

P B 7727, CAUSEWAY, HARARE

PUPIL REGISTRATION FORM

FORM:_____

Please submit a copy of the boy's Birth Certificate. If the boy is a Roman Catholic, please enclose a photocopy of his baptismal Certificate.

A PUPIL

SURNAME_____ RELIGION_____

FIRST NAMES_____ DOB_____/_____19_____

BIRTH CERTIFICATE NO_____ ISSUED AT_____

RESIDENTIAL ADDRESS_____

1 POSTAL ADDRESS_____

HOME TELEPHONE NO.:_____

B FATHER

SURNAME_____ RELIGION_____

FIRST NAMES_____

OCCUPATION/POSITION HELD_____

2 EMPLOYER'S NAME AND ADDRESS_____

BUSINESS TELEPHONE NO_____ HOME TELEPHONE NO_____

CELL TELEPHONE NO_____ E-MAIL ADDRESS _____

C MOTHER

SURNAME_____ RELIGION_____

FIRST NAMES_____

OCCUPATION/POSITION HELD_____

3 EMPLOYER'S NAME AND ADDRESS_____

BUSINESS TELEPHONE NO_____ HOME TELEPHONE NO_____

CELL TELEPHONE NO_____ E-MAIL ADDRESS _____

PLEASE TURN OVER

Figure 8: St George's College Enrolment Questionnaire Pg1

Form B/2003 (continued)

D MEDICAL AID INFORMATION

MEDICAL AID SOCIETY_____   MEDICAL AID NUMBER_____

MEMBER'S SURNAME_____   MEMBER'S FIRST NAMES_____

BOY'S SUFFIX NUMBER_____

E BOY'S PRESENT SCHOOL

SCHOOL _____

ADDRESS_____

CLASS/GRADE/SET_____

F DETAILS OF RELATIVES WHO ARE OLD GEORGIANS

NAME	RELATIONSHIP TO BOY	YEARS ATTENDED COLLEGE	HOUSE
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

G DETAILS OF BROTHERS CURRENTLY (2001) AT:

ST GEORGE'S, HARTMANN HOUSE OR ST MICHAEL'S PREPARATORY SCHOOL

NAME	FORM	GRADE	HOUSE	ST G	H H	ST M
_____	____	____	____	____	____	____
_____	____	____	____	____	____	____
_____	____	____	____	____	____	____
_____	____	____	____	____	____	____

H APPLICATION FOR A BOARDING PLACE

N.B. There is a limited number of boarding places, usually only available to residents outside Harare.

	YES	NO
• Do you wish your son to be a boarder	_____	_____

I DECLARATION BY PARENT/GUARDIAN

I confirm that I am willing and able to pay the fees and I accept and understand that inflationary adjustments to the fees can be expected on 2002 and thereafter. I am willing to accept the discipline of St George's College with regard to my son. I understand that the decision of the Headmaster with reference to this application is final.

SIGNED:_____   DATE:_____

NAME, IN CAPITAL LETTERS:_____

[FOR THE COLLEGE OFFICE ONLY]

BIRTH CERTIFICATE	BAPTISMAL CERTIFICATE	H M'S REPORT	RECEIPT NO.
_____	_____	_____	_____

Figure 9: St George's College Enrolment Questionnaire Pg2

The office of the Deputy Headmaster runs examinations at St George's College. The system is fully manual with timetable slots being allocated at the discretion of the Deputy Headmaster. The process typically starts off with the allocation of slots for the external exams followed by the slotting in of the internal examinations. Subject teachers are only required to specify the papers they will be intending to have their students examined in.

The Accounting Function

The accounting function at St George's College runs on Pastel® Accounting version four. The system also caters for Hartman House, which is essentially the primary education branch of the College.

Within the system, guardians are treated as categorised debtors with each debtor account being linked to an enrolment record in the registry database by means of an account number. The link is however not live with the categories in the Pastel system being updated by a customised Microsoft Access to Pastel script which is executed before any billing is carried out. The school has 22 categories for its students with each category being charged a different set of fees. Invoices are generated using the Pastel system and mailed out at the end of each term with the amounts being due before the beginning of the term for which the fees will be applicable. Special categories exist for students awaiting enrolment and those who will have either withdrawn from the school or completed their studies.

Library Facilities

The library at St George's College uses the Dewey Decimal Classification (DDC) system for classifying the material in its catalogue with the exception of works of fiction, which are classified as to their authors. All the material in the library may be borrowed and the library uses the Browne Circulation Management System. In addition to the standard Browne system, the library also employs a reservation system for managing the circulation of materials in high demand. Details of the operation of the two systems are given in the appendices.

A record of the catalogued material is also maintained in a Microsoft Access database. The database contains the Accession Number, Authors, Title and Dewey Number of each book. The record is updated each year by creating a new database table during the stock taking period at the end of the year.

Membership to the library is open to students and members of staff. The library also provides access to the Internet through Microsoft Windows NT workstations.

Boarding facilities

The school has boarding places for about 120 boys in a hall for juniors and flats for seniors. Most of the boarders are weekly boarders with only about 30 full time boarders. The boarding master keeps a record of the students enrolled in boarding in addition to the record maintained by the Registrar.

Functional Requirements

Requirements Definition

The system must provide facilities for

- `A` – Maintaining enrolment records for both registered and deregistered students.
- `B` – Maintaining an academic registry.
- `C` – Examinations scheduling.
- `D` – Collaboration in the preparation of assessment reports.
- `E` – Fees processing.
- `F` – Event scheduling.
- `G` – Managing a library.
- `H` – Running mail merge jobs.
- `I` – Executing database wide queries in Standard Query Language

Requirements Specification

A – The system must provide facilities for maintaining enrolment records for both registered and deregistered students.

    1. The system should maintain a record of the following details for each registered student.

        a. Registration number.
        b. Surname.
        c. First name.
        d. Middle name.
        e. Sex.
        f. Date of birth.
        g. Date enrolled.
        h. Date record was last updated.
        i. Class.
        j. Birth certificate number.
        k. Father's details
        l. Mother's details
        m. Marital status of parents
        n. Primary guardian's details.
        o. Medical aid society details
        p. Doctor's contact details.
        q. Religion
        r. Primary language spoken at home.
        s. House details
        t. Siblings enrolled.
        u. Remark

2. The system should maintain a record of the following details for each deregistered student.
   a. Registration number.
   b. Surname.
   c. First name.
   d. Middle name.
   e. Sex.
   f. Date of birth.
   g. Date enrolled.
   h. Date deregistered.
   i. Reason for deregistration.
   j. Date record was last updated.
   k. Last academic year of study
   l. Birth certificate number.
   m. Father's details
   n. Mother's details
   o. Primary guardian's details.
   p. Religion
   q. Primary language spoken at home.
   r. House details
   s. Remark
3. Details of parents and guardians should comprise the following
   a. Title
   b. Surname
   c. First name
   d. Residential Address
   e. Home Phone Number
   f. Employer
   g. Business address
   h. Occupation
   i. Business contact telephone phone number
   j. Business contact fax number
   k. Cell phone number
   l. Email address
   m. Marital status
4. Medical aid society details should comprise
   a. Membership number
   b. Name
   c. Type
5. Doctor's contact details should comprise
   a. Name.
   b. Contact phone number.
   c. Contact address
   d. Email Address

6. House details should comprise
    a. Name
    b. House master
    c. Boarding status of students in the house.

7. The system should provide facilities for automating the following updates
    a. Transferring all the students in a given class to a new class.

Rationale	
A1 – 6	Extracted from current registration records
A7	Required to ease the transfer of students at the end of each academic year when there is no re-streaming being undertaken.

B – The system must provide facilities for maintaining an academic registry.
1. The system should provide facilities for maintaining the following details
    a. Classes
        i   Class
        ii   Academic year
        iii Form teacher
    b. Subjects offered
        i   Subject Code
        ii   Title
        iii Teacher
        iv   Academic year offered
        v   Supervising Department
    c. Teachers
        i   Title
        ii   Surname
        iii Initials
    d. Departments
        i   Title
        ii   Head of Department
    e. The courses taken by each student

Rationale	
B.1	Extracted from current records.

2. The system should provide facilities for automating the following updates
    a. Specifying that all the students taking a given subject in a given class or academic year should replace that particular subject with another subject.
    b. Specifying that all students in a given class or academic year should have their association with subjects deleted.
    c. Specifying that all students in a given class or academic year should be assigned a given set of subjects.

Rationale	
B.2.a	Consider the following scenario – All students taking Form 3 English for class 3A1 who have graduated into Form 4 and are in 4A1 need to be registered as taking Form 4 English.
B.2.b & c	Quick reassignment after re-streaming.

3. The system should be capable of generating the list of students enrolled for a given subject.
    a. The list should contain the following details
        i   Subject code.
        ii   Subject Title.

        iii   Subject teacher.
        iv   Date and time it was generated.
        v   List of students enrolled for that given subject.
b. It should be possible to generate a list for a given subject selected from a list of subjects.
c. It should be possible to generate a collection of lists for
        i   All subjects offered to students in a given academic year.
        ii   All subjects offered by a given department.
        iii   All subjects
d. The lists should be available for
        i   On screen viewing
        ii   Printing
        iii   Exporting to a spreadsheet
e. Facilities b & c should also be made available through a web interface. The interface however only needs to provide for onscreen viewing.

Rationale	
B.3.a	Details required to facilitate the verification of student lists with subject teachers.
B.3.c	A time saving facility for batch operations.
B.3.e	This would allow teachers to check for errors automatically without having to make a formal request to the academic registry. Also provides an opportunity for saving stationery.

C - The system must provide facilities for examinations scheduling.
  1. The system should provide for the making of a schedule entry given
     a. A unique code for the paper.
     b. Title of the paper
     c. Level e.g. L6, A, O
     d. Date, starting time and ending time.
     e. Venue.
     f. List of invigilators selected from a list of teachers.
     g. A specification of the students entering for the paper. The specification should be through one of the following.
        i   Specifying a list of subjects
        ii Entering students individually

Rationale	
C.1.a	Most papers already have established codes. It would therefore be inappropriate to generate these automatically
C.1.b	Same as C.1.a
C.1.c	It should be technically possible to generate the level from the list of students specified but a problems would result when a given students takes say an O level exam when he/she is still in Form 3.
C.1.g	The list of students is required to facilitate identification and resolution of clashes. Specifying a list of subjects should be viewed as a time saving alternative to having to specifying the students individually.

  2. The system should provide the following maintenance functions
     a. Changing the details of a schedule entry.
     b. Deleting a given entry deleting all the entries over a given time period.

Rationale	
C.2.b	Facilitating cleanup

3. The system should facilitate the identification and resolution of clashes.
   a. The system should be capable of generating a list of students affected by a clash. The list should contain
      i   The name, registration number and class of the student
      ii The subjects clashing i.e. subject code, title and timing
   b. The system should facilitate the resolution of the clash by moving a given student from a given paper to another.
   c. The system should be capable of generating a list of invigilators affected by a clash.

Rationale	
C.3.b	It is envisioned that the clash would be resolved by scheduling a separate paper on the same day for the affected students.

4. The system should be capable of producing the following reports.
   a. An exam timetable with the following details for each entry
      i   Date and time of the exam.
      ii   Level
      iii   Paper code
      iv   Title of the subject
      v   Duration
      vi   Number of students entered for the paper.
      vii   Venue
      viii List of invigilators
   b. Invigilation summary
      i   Name of the teacher
      ii   Date and time of each subject being invigilated
   c. Exam subject lists
      i   Date and time of the exam
      ii   Paper code
      iii   Subject title
      iv   Level
      v   Duration
      vi   Number of students enrolled for the exam
      vii   List of invigilators
      viii List of students i.e. name, registration number and class
   d. Student's examinations
      i   Name of the students
      ii   Registration number
      iii   Class
      iv   List of papers being taken i.e. date, time, paper code and subject title

Rationale	
C.4	Extracted from current reporting requirements

5. The reports specified in C.4 should be available for
   a. Display on the screen
   b. Exporting to a spreadsheet
   c. Viewing via a web based interface

Rationale	
B.5	There is no direct reference to printing as it is anticipated that printing would be best handled by exporting to a spreadsheet package.

D – The system must provide facilities for the preparation and storage of assessment reports.

1. The system shall provide a facility for specifying sessions for both major and minor assessment reports.
    a. The system shall be capable of running more than one session at a given time.
    b. An assessment session specification shall include details of
        i   Year and term covered by the session
        ii   Whether it is a major or minor session
        iii   Number in current term e.g. 3rd minor report
        iv   Start date
        v   End date
        vi   Classes covered by the session

Rationale	
D.1.a	An allowance to facilitate system evolution. Not a direct requirement of any current implementation.

2. A major assessment report will contain the following details
    a. Name of the student
    b. Academic year
    c. Class
    d. Term
    e. Year
    f. Age of the student (years and months)
    g. Average age of the student's classmates
    h. Number of days absent from school
    i. Student's average mark
    j. Class' Average mark
    k. Student's overall position in his/her class based on his/her average mark
    l. Performance in subjects detailing
        i   Subject title
        ii   Marks
        iii   Average mark for subject
        iv   Position in subject
        v   Effort symbol i.e. A – F
        vi   Remarks
        vii Teacher's initials
    m. Comment on performance in extra-curricula activities
    n. Form teacher's remarks
    o. House master's remark
    p. Headmaster's remarks

3. A minor assessment report will contain the following details
   a. Name of the student
   b. Academic year
   c. Class
   d. Term
   e. Year
   f. Report's number in current session
   g. Performance in subjects detailing
      i   Subject title
      ii   Effort symbol i.e. A – F
      iii Remarks
      iv   Teacher's initials
   h. Form teacher's remarks

Rationale	
D.3	Based on PEHS and St George's College's minor assessment reports.

4. The assessment should be done over the following stages
   a. The specification of the session
   b. The entering of data by subject teachers. The interface should give the teachers.
      i   The list of subjects for which they will be required to enter performance details.
      ii   For each subject
           ➢ A list of students enrolled for that subject
           ➢ Entry fields for the performance details of each student i.e. Mark, effort symbol
           ➢ Generate the average marks and positions and then provide for the entering of the remarks
   c. Entering of comments by the form teachers
   d. Entering of comments by the headmaster
   e. Closure of the session and the printing of the reports

Rationale	
D.4.b	The separation into separate stages of the entering of the marks and the comments is aimed at allowing the system to calculate and present ranking and averages before the entering of the comments.

5. The system shall be capable of producing a progress report for monitoring purposes. The report should contain the following
   a. Active sessions and the stage each will be in
   b. Marks not yet submitted for active sessions in the subject teacher phase
   c. Form teachers with pending comments

Rationale	
D.5	For administrative purposes. The deputy head needs to this information in order to enforce deadlines for the preparation of reports.

6. The system should be capable of generating
   a. All the assessment reports for a given student over a given period of time.
   b. All the assessment reports for a given session by class and academic year with multi-select capabilities.

Rationale	
B.6.a	Generation of academic transcripts as is the case at St George's College.
B.6.b	Batch preparation of end of term reports

`E -` Fees processing.

Please Note
The requirements for the fees processing system have been largely adopted from those that were formulated for the Mt Pleasant High School fees processing system. The case study has been included in the appendices but for a more comprehensive understanding of the specification rationale, one should consult the original development report. The report is included in the list of references.

1. The system must maintain a link to the students' records database for its list of students.

2. The system must facilitate the recording of fees debited to each student account.
    a. The following should be recorded for each fee debited.
        i   Student Registration Number
        ii   Date charged
        iii Amount debited
        iv   Description of fee

    b. The system operator should be able to debit a student's account by a combination of the following
        i   Selecting an editable template containing a list of fees along with their associated amounts.
        ii   Specifying a single fee and its associated amount.

    c. The operator should be able to charge all the students in a given class or house
        i   Facilities similar to those specified in 2b should be provided.
        ii   Before debits are made, the system should list the students to be charged sorted first as to sex then as to surname and first names.

Rationale	
E.2.b	Fees are often itemised. A template would avoid unnecessary repetition by treating a collection of fees that is conceptually single fee anyway, as a single fee.
E.2.c	Most fees will be charged on a batch job basis. In such a case, it is likely that students in the same class, or house in the case of boarding fees, would be charged the same fees

3. The system must facilitate the recording of payments. The following details must be kept
   a. Receipt number
   b. Date receipted
   c. Amount paid
   d. Student Registration Number of student affected

4. The system must be capable of producing statements of account for fee accounts
   a. The statement should contain the following details
      i   Name and Address of guardian (Addressee)
      ii   Student Registration Number
      iii Name of student
      iv   Class
      v   All transactions made between two user supplied dates
      vi   Balance owing and due.

   b. A facility for batch printing statements for all students in a given list of classes or houses should be available.

5. The system should be able to extract fees charged under a given fee over a user-defined period. The results should be available for both display and printing

Rationale	
E.5	Required for making entries into the general ledger.

6. The system should be able to provide an analysis of student account balances into debit and credit balances.
    a. The analysis should contain
        i   A list of all students with debit balances.
        ii   A list of all students with credit balances.
        iii   A list of all students with 0 balances.
        iv   Total debit balances
        v   Total credit balances

    b. The output referred to in 6.a should be made available
        i   On the Visual Display Unit (VDU)
        ii   In hard copy
        iii   For export to a spreadsheet

    c. The source SQL used to prepare the balance analysis should be made available for exporting to a text file.

Rationale	
6.a.i	Required by administration when pursuing legal action.
6.a.ii & 6.a.iii	Inserted to cater for system evolution. Have no current use in the system
6.a.iv & 6.a.v	Required by auditors. Opportunity for use as an input by the SDA executive committee when planning.
6.c	For use in running mail-merge jobs. Please refer to the requirements for mail merge jobs.

7. The system should be capable of maintaining deposit accounts for students
   a. Each deposit account type should maintain
      i   A minimum balance, which maybe set to zero.
      ii   A list of the students who will be having accounts of that type

   b. Deposit transaction records should contain the following field details.
      i   Date made
      ii   Amount
      iii Student affected
      iv   Receipt Number

   c. Account debiting records should contain the following details.
      i   Date made
      ii   Reason
      iii Reference codes e.g. the letter of authorisation from the disciplinary committee with regards to a charge against a "caution deposit account"
      iv   Amount debited

   d. A facility for producing a statement of account for any user-defined period should be provided for each account type. The statement should contain
      i   Name and address of the guardian
      ii   Name and registration number of the student
      iii Account type's minimum balance
      iv   Current balance and shortfall/excess if any
      v   Balance at the start of the period
      vi   A record of all deposits and credits over the specified time period
      vii Balance at the end of the period

   e. A facility for batch producing the statements for all students whose balances are
      i   Below the minimum balance.
      ii   At the prescribed level
      iii Above the minimum balance

Rationale	
7	Deposit accounts will facilitate the keeping of things like pocket money and caution fee deposits.

F – Event scheduling.
1. Events shall be split into the following categories
   a. Clubs and societies
   b. Sports
   c. Examinations
   d. General i.e. those not falling under a, b or c

Rationale	
F.1	Refer to the extract of the calendar of Prince Edward High School.

2. The system shall maintain the following details with regards to each club, society and sport
   a. Members of staff in charge of each activity.
   b. Meeting times detailing
      i   Team
      ii   Supervising member of staff
      iii   Meeting days and times
      iv   Venue of meeting
   c. Fixtures' details specifying
      i   Date
      ii   Event
      iii   Venue
      iv   Time
      v   Remark

Rationale	
F.2	Refer to the extract of the calendar of Prince Edward High School.

3. Details of examinations will be extracted from the examination schedules by means of a live database link.

Rationale	
F.3	This a middle of the road approach. On one end, the calendar needs to contain the details of examinations. On the other, the system seeks to avoid redundancy.

4. General activities will comprise the following details
   a. Date
   b. Event

5. The system shall provide facilities entering, editing and deleting the above entries.

6. The system shall provide a web interface for accessing the above information. The interface shall provide the following facilities.
    a. Viewing a combination of all the club, sport, examination and general events scheduled for a given day or period. The user should be able to restrict the returned results to a combination of some or all of the above.
    b. View the schedules for a given club, society or sport.

Rationale	
F.6	Calendar information is considered public information.

G - Managing a library.
1. The system shall provide a facility for all members of staff, registered students and deregistered students to be registered as patrons.
   a. Registration will be by means of live linkages to the rest of the database.
   b. With respect to these patrons, the Patron ID will be derived from the student's or member of staff's registration number.

Rationale	
G.1	Technically, the library could maintain its own list of patron details so as to limit its dependency on student records and the registry but this would cause undue redundancy.

2. The system will facilitate the registration of patrons not falling under the criteria specified in G.1 above. With respect to these, the system will maintain the following details
   a. A unique patron ID
   b. Title
   c. Forenames
   d. Surname
   e. Contact Address
   f. Contact phone number
   g. Membership category
   h. Expiry date of membership

Rationale	
G.2	Seeks to cater for the approved researchers who may be given temporary membership.

3. The system will provide a facility for categorising patrons
   a. For registered students, categories will be as to the academic year of the student.
   b. For deregistered students, categorisation will be as to the deregistration status.
   c. All members of staff will fall under a single category
   d. External patrons will be assigned to categories on an individual basis.

Rationale	
G.3.a	This is typical in most schools with junior students having fewer library privileges than their seniors.
G.3.b	It may be argues that it would be simpler to place all deregistered students into one membership category. This approach however seeks to improve on the system's flexibility

4. Each membership category will contain circulation rules with respect to the number of books that may be borrowed from each book section or a combination of book sections.

Rationale	
G.4	Circulation rules as opposed to hard coded values will allow for flexibility.

5. The library system will maintain the following details with respect to each book
    a. A unique accession number.
    b. Author
    c. Title
    d. Edition
    e. Year of Publication
    f. ISSN for periodicals and ISBN for books.
    g. Date acquired
    h. Condition when acquired [Old | New]
    i. Section
    j. Location e.g. Main Library

Rationale	
G.5	Based on research into the information typically stored by libraries in Zimbabwe with respect to books in their catalogues. Please refer to the *Librarius 2000 Plus* research document listed in the references section.

6. A section specification will contain details of circulation rules with respect to
    a. Loan period
    b. Time the book will be due on the date on which it will be due
    c. Fine per time period i.e. per day or hour

Rationale	
G.6	The time the book will be due on a given day has been included so as to cater for overnight loans of reserved books. Adopted from the University of Zimbabwe's circulation policy and included for system evolution purposes.

7. The system will keep track of fines
    a. The system will record all due fines including details of
        i Date the fine was entered
        ii Patron fined
        iii Book on which the fine was charged
        iv Amount of the fine
    b. The system will keep a record of fines paid but not yet transferred to other books of accounts
    c. The system will facilitate the making of transfers out of the system with details of
        ➢ Date of entry
        ➢ Receipt number
        ➢ Amount receipted
        ➢ Description
        ➢ Transferring librarian

Rationale	
G.7.a	Most libraries allow patrons to settle their fines within a given time period. A record therefore has to be kept until the fine is paid.
G.7.b	Libraries often receipt fines before transferring them to the bursar's office
G.7.c	Required for auditing purposes

8. The system shall facilitate the maintenance of a circulation record.
   a. The following details will be maintained with respect to each book on loan.
      i   Accession number of the book
      ii   Patron Id of the borrower
      iii Date and time due
   b. Returns should be processed by deleting the original loan entry.
   c. A book may only be borrowed from and returned to the library in which it is registered as located.
   d. The system will facilitate the making of requisitions for books over a given period of time. A pending requisition will count as a book on loan in the calculation of borrowing limits.

Rationale	
G.8.a	The time information will facilitate the making of short overnight loans. Where a book may be returned at any time on the due date, the library may set midnight as the due time.
G.8.c	The catalogues will be integrated. As such, it will be possible for a patron to return a book to the wrong library in an institution with more than one library.
G.8.d	Essentially, requisitions are meant to facilitate circulation. It would therefore be inappropriate for an individual to request a book that he would not be able to borrow because of loan limits or to keep a book on request whilst borrowing another.

9. The system shall provide a facility for suspending members. A suspension record should detail
   a. Date made
   b. User name of the suspending librarian
   c. Patron suspended
   d. Free text remark on the reason for suspension.

Rationale	
G.9	It is common practice to suspend the library privileges of a patron who will have broken the library's rules.

10. The system shall facilitate the production of the following reports
   a. Catalogue record of all the books registered in one of or some of the libraries. The report will contain with respect to each book
      i   Accession number of the book
      ii   Author
      iii   Title
      iv   Edition
      v   Year of Publication
      vi   ISSN for periodicals and ISBN for books.
      vii   Date acquired
      viii Condition when acquired [Old | New]
      ix   Section
      x   Location e.g. Main Library
   b. Record of all books on loan
      i   Accession number of the book
      ii   Author
      iii   Title
      iv   Section
      v   Location e.g. Main Library
      vi   Date due
   c. Record of all overdue books
      i   Accession number of the book
      ii   Author
      iii   Title
      iv   Section
      v   Location e.g. Main Library
      vi   Date due
      vii   Days overdue by
   d. Patrons in one, or some, of the registered categories
      i   Patron ID
      ii   Name
      iii   Contact Address
      iv   Membership category
      v   Number of books borrowed
      vi   Number of books overdue
      vii   Fines pending
   e. List of suspended patrons
   f. Record of transactions in the Fines Account between any two dates detailing
      i   Date and time generated
      ii   Period covered
      iii   Current balance on account
      iv   Balance on opening date
      v   Transactions over the given period.
      vi   Balance on the closing date

11. Generated reports should be available for output onto the screen and for export to a spreadsheet.

Rationale	
G.11	No reference is made to direct printing. It is anticipated that printing will be handled through a spreadsheet.

12. A web based interface for searching the catalogues and viewing patron records should be included
   a. It should be possible to query the catalogues based on a specification of a book's
      i   Accession number of the book
      ii   Author
      iii Title
      iv   ISSN for periodicals and ISBN for books.
   b. Query specifications should include options for returning
      i   Exact matches
      ii   Matches containing the specified string
      iii Full pattern matching
   c. Each patron should be able to view his/her catalogue record detailing
      i   Patron's name and ID
      ii   Books on loan
      iii Books requested
      iv   Fines owing
   d. Patrons should be able to request books using the web interface.

Rationale	
G.12	Based on an analysis of the University of Zimbabwe's Open Public Access Catalogue system. Please refer to the references list

`H -` Running mail merge jobs.
1. The user should specify the records to be used in the mail merge through Standard Query Language.
2. The user should be able to specify the body of the message indicating where personalised details will be inserted.
3. The facility should provide the following editor specific services
   a. Document preparation in the Microsoft Rich Text Format (RTF).
   b. Saving and retrieval of documents in RTF.
   c. Printing of RTF documents in their native style.
   d. Search and replace function.
   e. Undo system with a capacity of undoing at least the last 10 changes made.
   f. Clipboard functions (Cut, Copy & Paste)
   g. The following RTF formatting styles should be provided
      i   Bold
      ii   Underline
      iii   Italics
      iv   Multiple font sizes within a document
      v   Multiple fonts within a document
      vi   Alignment
             ➢ Left
             ➢ Right
             ➢ Centre
      vii   Indention
      viii   Image imbedding
4. A toolbar function for accessing all the functions listed in 3
5. A facility for previewing the Mail Merge job should be provided.
6. A facility for inserting the following into the current document
   a. Calendar Dates
   b. A picture file.
   c. A text or RTF file
7. Where the host machine has an installation of Microsoft Word, the system should provide spell-checking facilities based on the dictionary.

Rationale	
H.1	This will allow the capacity of more reports to be made available by writing new SQL files. It will also provide an opportunity for the reuse of a system developed by the current developer in 1999.
H.2	This is a standard mail merge facility.
H.3	Standard word processing facilities.
H.4	Accessibility
H.5	Minimise paper wastage
H.7	Writing a custom dictionary would be a rather complex task. Microsoft Word exposes a relatively popular dictionary.

I – Executing database wide queries in Standard Query Language
1. The user should be able to Enter, Save, Retrieve and Print out the source SQL.
2. The user should be able to export the results as to spreadsheets.

Rationale	
I.1	Facilitates data mining.
I.2	Facilitates the electronic transfer of data.

J – Supporting the following spreadsheet formats.
1. Generic comma delaminated format as implemented in Microsoft® CSV.
2. Microsoft Excel for Windows 1995

Rationale	
J.1	Will allow exporting of the data to any spreadsheet package.
J.2	Microsoft Excel is the most popular spreadsheet package available. The Windows 95 version represents, in the opinion of the developer, the most basic format that should be accessible to most other spreadsheets.

Non-Functional Requirements

1. IDs used within the system should contain checksum data.

Rationale
To minimise data entry errors.

2. The system should restrict access to private information. The following information may however not be treated as confidential.
   a. Calendar lists
   b. Lists of clubs, societies and sports including the names of club patrons
   c. Library catalogues excluding specifics of to whom a book will be on loan

Rationale
This was a major concern of the personnel who were interviewed during the background research phase.

3. The system should endeavour to generate all printouts on A4 sized paper.

Rationale
It is anticipated that some of the reports will not fit onto A4 paper hence this is not a strict requirement. A4 paper is preferred because it is supported by most desktop printers.

4. The implementation should support the Microsoft Windows family of operating systems with backward compatibility to Windows 95 for non-server components.

Rationale
The schools used in the case studies used PCs with Microsoft Windows as their main operating systems. Training costs would be minimised if the implementation taps into the users' current experience.

5. The system should endeavour to use open source components.

Rationale
Budgetary constraints. Also part of the primary vision of the project.

DESIGN PRINCIPLES

Design Philosophy

The system designer intends to adopt a top-down approach as proposed by Somerville (1995). The design philosophy will hinge on Viewpoint oriented analysis so as to produce a model that will be consistent with the diversity of the envisaged users.

Given the data-heavy nature of the proposed system, the fist stage of the design process will focus on data modelling. Underlying the design process will be an intention to produce an implementation independent design for the system database. It is also intended that the design will be based on an intention to limit the redundancy of the data by providing a single logical representation of all the views required by the system. To this end, the design will be required to include an explicit statement of the level of normalisation attained along with justifications for any redundancies that maybe exist in the final design.

The data modelling will be followed by Architectural Design, which will detail how the system will be broken down into the view-based subsystems. It is the intention of the developer to document this stage using structure charts. This stage should give a complete overview of the overall structure of the system and the level of modularisation that will be desired. In line with this intention, each terminal node in the structure should represent a modular component.

The specified architecture will be elaborated through detailed Component Design. In line with Somerville's recommendations, the design of each component will focus on the following
- ❖ The rationale behind the component's inclusion in the system.
- ❖ The position of the component in the hierarchy i.e. parent and direct descendants.
- ❖ Identification of any known or intended dependencies between the component and the rest of the system
- ❖ User and/or programming interface.
- ❖ Algorithms for any specialised tasks.

Underlying the above philosophy is a desire to completely specify how a component will behave to ensure that this behaviour will be in line with system requirements whilst not constraining creativity during implementation by specifying detailed algorithms.

To facilitate rapid application development, an extensive use of existing system components will be made. In such instances, the modified design documentation of the components will be included. Underlying this process will be the desire to ensure that the meeting of the system requirements will not be compromised whilst also ensuring that time is not wasted reinventing the wheel.

Choice of Development Environment

Given the specialised nature of programming languages, it is the opinion of the designer that a decision on the development environment must be made before the design so as to ensure that the produced design will be technically feasible. This will also allow the use of specialised features that are only available in one development environment.

Applications

The primary constraint in selecting the choice of the development environment has been the need to utilise the code library that is currently available to the developer. It is the opinion of the developer that this requirement is crucial in ensuring that the project will be technically feasible within the period allocated to it. Given that the library of code is in Visual Basic 6.0, it follows that the current applications will also have to be developed in Visual Basic 6.0. The developer would however want to raise the following qualities of Visual Basic that would have made it a natural choice even without the legacy code constraints

  i   The availability of high-level user interface components that conform to the Windows look and feel. This would limit training requirements given the current wide use of Windows operating systems within schools. It also aids rapid application development.

  ii   Visual Basic supports integrated database development that is suitable for the data heavy applications.

  iii To increase productivity of the application, it must be in a position to communicate and if possible, integrate with other applications that may provide useful and advanced facilities. Visual Basic has built in facilities for integration with Microsoft Office applications and in particular, Microsoft Excel that may be used as an exporting format.

  iv   The available version of Visual Basic, VB 6.0, represents a levelling off of the development environment. Most of the features have reached their maturing stage and can safely be assumed to be not subject to further correction unless absolutely necessary. This is important since the backward compatible versions of runtime libraries that are often supplied with improvements in a given language tend to be inadequately compatible with the primary versions.

Data Access Method

Having decided on the programming language, the next step is to decide on the database management system. Visual Basic 6.0 has three database development libraries available namely

  i   Data Access Objects (DAO)
  ii   Remote Data Objects (RDO)
  iii ActiveX Data Objects (ADO)

RDO is not being considered since it has been superseded by ADO and is provided primarily for backward compatibility. The guidance on making a decision between ADO and DOA is provided in the Microsoft Knowledge Base Article 225048 - INFO: Issues Migrating from DAO/Jet to ADO/Jet. The designer has decided to use ADO because of the following reasons

i   DAO was primarily designed for the Microsoft Jet environment. It is thus unsuitable in a system in which database independence is desired.

ii   DAO is inefficient in a multi-user environment and may run out of connection resources if several users connect simultaneously. Given the multi-user nature of the deployment environment, DAO would be unsuitable.

iii   The Jet buffer has a five second timeout. Updates using DAO would thus not be visible to other user for five seconds.

iv   Microsoft is promoting ADO as the Data Acess technology of the future. Continued support for DAO may thus be limited.

Report Designer

Microsoft Visual Basic supports tight integration with two report designers namely the Microsoft Data Report Designer and Seagate Crystal Reports. The choice of the report designer to use has been based on the Microsoft Technical FAQ which is located at http://msdn.microsoft.com/vbasic/previous/vb6/techfaq.aspx under the question "When would I use the Data Report Designer in Visual Basic 6.0 vs. Seagate Crystal Reports?" The developer has decided to use Crystal Reports because of the following

i   The Data Report Designer supports exporting to HTML only. Crystal Reports on the other hand supports exporting various formats in addition to HTML. In particular, Crystal Reports supports exporting to spreadsheets, which would add value to the application in the area of output transmission by, email.

ii   Microsoft recommends Crystal Reports for building reports that can later be scaled up in terms of complexity. Crystal reports thus has better support for system evolution.

iii   Crystal Reports has charting and drill down features that are not available in the Data Report Designer. Drill down features open up opportunities for value addition in analysing fees figures and assessment reports.

iv   Unlike the Data Report Designer reports, Crystal Reports' reports can be maintained independently of the application. Modularity is a key design principle given its capacity to greatly reduce maintenance costs according to Somerville (1995).

Remote Database Management System

It is intended that infoQuest will be database system independent. A choice however has to be made on the RDBMS to use for the development. The developer has chosen MySQL primarily because it is the largest open source database system with four million active installations to date according to the RDBMS' site at http://www.mysql.com/company/. Given the limited budgets of educational institutions, it

is most likely that most of them will choose an open source option. A development on MySQL would thus ensure stability maximisation for such users.

Server Side Scripting

The choice for the server-side scripting environment has been the most difficult. Four technologies were available to the developer namely
- ➢ Active Server Pages (ASP)
- ➢ Common Gateway Interface (CGI)
- ➢ Java Server Pages (JSP)
- ➢ Personal Home Pages (PHP)

PHP and CGI were dismissed because of their failure to provide database system independence. This left JSP and ASP as the feasible options. Sun Microsystems in their article entitles "A Comparison of ASP and JSP" located at http://java.sun.com/products/jsp/jsp-asp.html give the following as advantages of using JSP instead of ASP.

JSP is platform and server independent. It is available on Solaris, Windows, Mac OS, Linux and other UNIX platforms with server support on Apache, Netscape, and IIS. ASP on the other hand is available on Microsoft Internet Information Server and Microsoft Personal Web Server only as a native language. Both servers are only available on Windows platforms. On other platforms, third party proprietary ActiveX extensions have to be used. JSP.

JSP, according to Sun Microsystems, has error detection and threading functionality that helps to reduce server crashes. Sun on the other hand argues that ASP does not provide memory leak protection that makes it likely to crash and to bring the server down with it.

Ironically, Sun Microsystems provides the Sun One Active Server pages that provides support for ASP on the Apache server effectively giving ASP cross platform support. Arguably, it is unreliable to use information from the producer of a competing product but to date; the information remains unchallenged by Microsoft and may thus be regarded to represent a true and fair view of the two scripting options.

Given such a scenario, it seems natural to use JSP for server side scripting. A compromise had however to be made to enable code reuse since ASP scripting may be written in VBScript, which is a subset of Visual Basic. The impact of this decision on performance is expected to be minimal given that it is unlikely that a school will have more that 5000 users logged in simultaneously. The developer would however want to emphasize that this decision is a compromise.

Client Side Scripting

Microsoft in an Andrew Clinick article entitled "VBScript or JScript" acknowledges the limited nature of VBScript in terms of browser support. In particular, VBScript is not supported almost exclusively on Mozilla browsers like Internet Explorer. It is thus

unsuitable in an environment in which the users may not standardise to Internet Explorer. The developer has thus decided to use JScript, which is widely supported.

Secured Login For Web Access

Two alternatives were available to the developer namely Secure Sockets Layer (SSL) which would encrypt all the contents over the communication link and MD5 encryption for data considered sensitive. The use of SSL would ensure a higher level of security but it would also be more costly as SSL certificates have to be purchased. MD5 algorithms for encrypting information and ensuring secure logins are on the other hand available in open source format. The developer will therefore use the version of MD5 developed by Paul Andrew Johnston. The authentication method is used by Yahoo! among other high-profile websites to ensure secure access and is documented at Johnston's site at http://pajhome.org.uk/crypt/md5/chaplogin.html. The following is a description of the login system as outlined by Johnston:

```
"I originally wrote the MD5 implementation to improve
security on a login form for a website I was making,
running on a web space account with no SSL capability.
You can use a secure hash function to avoid sending the
password as clear text; this makes it more secure than
htaccess access control. First, the web server sends a
random variable to the client. The client asks the user
for the password, and makes the MD5 hash of the random
variable and password, and sends this to the server. The
server makes the MD5 hash of the random variable and its
stored password. If the two hashes match, then the user
knew the correct password, and the server allows access.
At no point was the password transmitted in the clear. An
eavesdropped won't be able to do a replay attack as the
server will then expect a different random variable."
```

This login system will require that the system maintain a table with access data for users.

Application Database Login

The system will not manage user logins and permissions. These will be set by the administrator who will maintain a list of users and their permissions on the database. These accounts will only be required for those users who use non-web based applications. This decision was reached after considering the diversity of user management systems across databases and the unfeasibility of developing a secure login system without having to make database specific decisions on the retrieval of user password from the RDBMS access control tables. The developer will however provide guidance on the users permissions that should be given for each user type.

DATA MODELLING

Database Entities

Student Records

1. Enrolled Students
    (<u>Student ID</u>, Surname, F Name, M Name, Sex, DOB, Enrolled, Updated, Birth Certificate, PMS, MA Number, Religion, Language, Remark, <u>Guardian, Father, Mother, Class, Doctor, House, MA Type ID)</u>

2. Enrolled Siblings
    (<u>Student ID, Student ID</u>)

3. Parents
    (<u>Parent ID</u>, Title, Surname, First Name, Home Add, Home Phone, Employer, Business Add, Occupation, Business Phone, Business Fax, Mobile Phone, Email, Marital status)

4. Deregistered Students
    (<u>Student ID</u>, Surname, First name, Middle name, Sex, Date Of birth, Enrolled, Deregistered, Reason, Updated, Last Year, Birth Certificate, Religion, Language, Remark, Contact Address, Contact Phone Number, Email, <u>Father, Mother, Guardian, House</u>)

5. Medical Aid Types
    (<u>MA Type ID</u>, MA Name, MA Type, Remark)

6. Doctor's Details
    (Doctor ID, Name, Dr Phone, Dr Address, Email)

7. House Details
    (House ID, Name, Boarding Status, <u>House Master</u>)

Academic Registry

1. Classes
   (<u>Class ID</u>, Class Name, Academic year, <u>Form Teacher</u>)

2. Subjects offered
   (<u>Subject Code</u>, Title, Teacher, Academic Year, <u>Dept ID</u>)

3. Members of Staff
   (<u>Staff ID</u>, Title, Surname, Initials, <u>Dept ID</u>)

4. Heads of Department
   (<u>Dept ID</u>, <u>HOD</u>)
   NB: Separated from department in order to avoid an insertion anomaly

5. Department
   (<u>Dept ID</u>, Dept Title)

6. Student Subjects
   (<u>Student ID, Subject Code</u>)

Examinations Scheduling

1. Examination Papers
   (<u>Paper Code</u>, Paper Title, Level, Starts, Ends, Venue, Remark)

2. Invigilation
   (<u>Paper Code, Invigilator</u>)

3. Student Exams
   (<u>Paper Code, Student ID</u>)

Assessment Sessions

1. Pending Assessment Sessions
   (<u>Session ID</u>, Year, Term, Assessment Type, Assessment Stage, Starts, Ends, NCT)

2. Completed Assessment Sessions
   (<u>Session ID</u>, Year, Term, Assessment Type, NCT)

3. Major Assessment Class
   (<u>Session ID, Class</u>, Class Avg Age, Class Avg Mark, Class Size, <u>Admin Teacher</u>)

4. Major Assessment Report
   (<u>Session ID, Student ID</u>, Student Age, Absent, Avg Mark, Class Position, ECA Performance, FT Remark, HM Remark, SH Remarks, <u>Class</u>)

5. Major Assessment Report Subjects
   (<u>Session ID, Subject ID</u>, Subject Avg Mark, Teacher Initials)

6. Major Assessment Report Subject Performance
   (<u>Session ID, Student ID, Subject ID</u>, Mark, Subject Position, Effort Symbol, ST Remarks)

7. Minor Assessment Classes
   (<u>Session ID, Class</u>)

8. Minor Assessment Report
   (<u>Session ID, Student ID</u>, FT Remarks, <u>Class</u>)

9. Minor Assessment Report Subjects
   (<u>Session ID, Subject ID,</u> Teacher Initials)

10. Minor Assessment Report Subject Performance
    (<u>Session ID, Student ID, Subject ID</u>, Effort symbol, ST Remarks)

Fees Processing

1. Registered Fees
   (<u>Fee code</u>, Fee Name)

2. Fee Account Debits
   (<u>Transaction Date</u>, Amount, <u>Fee Code</u>, <u>Student ID</u>)

3. Fee Account Credits
   (<u>Transaction Date</u>, Amount, Receipt, <u>Student ID</u>)

4. Fees Templates
   (<u>Template Name</u>, Template Data)

5. Deposit Account Types
   (<u>AT Code</u>, AT Name, Minimum Balance, Remark)

6. Deposit Accounts
   (<u>Acc Number</u>, Acc Status, <u>AT Code</u>, <u>Student ID</u>)

7. Deposit Account Debits
   (<u>Transaction Date</u>, Dr Reason, Amount, <u>Student ID</u>, <u>Acc Number</u>)

8. Deposit Account Credits
   (<u>Transaction Date</u>, Amount, Receipt, <u>Student ID</u>, <u>Acc Number</u>)

Event Scheduling

1. Club Types
   (CT Code, Description)

2. Clubs
   (Club Code, Name, CT Code)

3. Club Supervisors
   (Club Code, Staff ID, Role)

4. Regular Club Meeting Times
   (Club Code, Start Date, End Date, Start Time, End Time, Recurrence Rule, Venue, Remark, Supervisor)

5. Club Fixtures
   (Start Date, Start Time, End Date, End Time, Venue, Remark, Club Code)

6. General Events
   (Start Date, Start Time, End Date, End Time, Event Details)

Library Management

1. External Patrons
    (<u>Patron ID</u>, Title, F Name, M Name, Surname, Address, Phone, Expiry Date, <u>Category ID</u>)

2. Membership Categories
    (<u>Category ID</u>, Category Name, Membership Rule, Borrowing Limit Rule)

3. Books
    (<u>Book ID</u>, Classification Code, Title, Edition, Publication Year, ISSN or ISBN, Date Acquired, Acquisition Condition, Remark, <u>Section ID, Location ID</u>)

4. Authors
    (Book ID, Author)

5. Locations
    (<u>Location ID</u>, Location Name, Remark)

6. Sections
    (<u>Section ID</u>, Section Name, Remark, Loan Period Rule, Overdue Fine Rule)

7. Book Loans
    (<u>Book ID, Patron ID</u>, Due Date)

8. Suspensions
    (<u>Patron ID</u>, Date Made, Remark, S Expires, <u>Staff ID</u>)

9. Requisitions
    (<u>Patron ID, Book ID</u>, R Effective, R Expires)

10. Unpaid Fines
    (<u>Patron ID, Book ID</u>, Date, Amount)

11. Paid Fines
    (<u>Receipt</u>, Date Paid, Amount, <u>Staff ID</u>)

11. Transfers Out
    (<u>Date</u>, Receipt, Amount, <u>Staff ID</u>)

Secured Logins

1. Online Logins
    (<u>User Name</u>, Password, Allow Nominal Roll, Allow Assessment Sessions)

Data Dictionary

Attribute	Type	Remark
AAS	Boolean	Allow access to assessment sessions records if true
Absent	Integer	Number of days in the terms absent
Academic year	Text – 1cc	[0 – 9]
Acc Number	Text – 10cc	Account number. First 7 letters from student's registration number followed by two digits generated by the system and a check letter
Acc Status	Text – 1cc	[O – Open | C – Closed]
Acquisition Condition	Text – 10cc	E.g. New
Address	Text – 90cc	Contact address
Admin Teacher	Text – 8cc	Of type Staff ID. The administrative member of staff responsible for entering the Admin comments.
Amount	Currency	
ANR	Boolean	Allow access to nominal roll if true
Assessment Stage	Text – 10cc	[Init – Initialised but not yet open | Subjects - Open for entering of subject data | Class – Open for Class teachers | Admin – Open for Admin staff comments | Closed – Awaiting final approval]
Assessment Type	Text – 5cc	[Major | Minor]
AT Code	Integer	System generated index for Account Types
AT Name	Text – 20cc	Account Type Name
Author	Text – 15cc	Book's author
Avg Mark	Float	Student's average mark across subjects.
Birth Certificate	Text – 15cc	Birth certificate number
Boarding status	Text – 1cc	[B-Border | D-Day Scholars]
Book ID	Text – 8cc	System generated index
Borrowing Limit Rule	Text – 90cc	SQL for specifying the limit of each subscriber
Business Add	Text – 90cc	Contact business address
Business Fax	Text – 20cc	Contact fax number
Business Phone	Text – 20cc	Contact business phone number
Category ID	Integer	System generated index
Category Name	Text – 15cc	Title of the membership category
Class	Text – 5cc	Class ID
Class Avg Age	Text – 4cc	Class' average age. First two letters for years and the last two for the months.
Class Avg Mark	Float	Average of the students' individual average marks.
Class ID	Integer	System generated indexing field
Class Name	Text – 5cc	Unique user specified class name
Class Position	Integer	Student's position in class based on average marks

Attribute	Type	Remark
Class Size	Integer	Number of students in class
Classification Code	Text – 15cc	As to library's classification system e.g. Dewey or some other custom system
Club Code	Integer	System generated ID for indexing clubs, societies and sports.
Club Name	Text – 15cc	Society's name
CT Code	Integer	System generated ID for indexing club types
CT Description	Text – 15cc	Description of the Club type category e.g. Society
Date Acquired	Date	
Date Made	Date	Date on which patron was suspended
Date Paid	Date	Date on which the fine was paid
Dept	Text – 5cc	Dept ID
Dept ID	Text – 5cc	Unique system generated indexing key
Dept Title	Text – 20cc	Official title of the department
Deregistered	Date	Date student was deregistered
DOB	Date	Date of Birth
Doctor	Text – 8cc	Doctor ID
Doctor ID	Text – 8cc	Unique system generated ID for doctor
Dr Address	Text – 90cc	Doctor's contact mailing address
Dr Phone	Text – 20cc	Doctor's contact phone number
Dr Reason	Text – 30cc	Brief description of reason for debit e.g. Lost text book
Due Date	Date	Includes time information
ECA Performance	Text – 50cc	Comment on student's performance in extra-curricula activities.
Edition	Text – 10cc	Publication edition
Effort Symbol	Text – 1cc	[A – F]
Email	Text – 25cc	Contact email address
Employer	Text – 30cc	Name of employer
End Date	Date	Date on which period covered by schedule entry ends.
End Time	Time	Time at which recurring event ends
Ends	Date	Ending time. Includes time information
Enrolled	Date	Date student was enrolled
Event Details	Text – 50cc	Description of general event.
Expiry date	Date	Date on which subscription expires
F Name	Text – 20cc	First name
Father	Text – 8cc	Parent ID
Fee Code	Integer	Indexes fee table. Generated by system.
Fee Name	Text – 30cc	Official title of a fee.
Form Teacher	Text – 8cc	Staff ID
FT Remark	Text – 50cc	Form teacher's remarks
Guardian	Text – 8cc	Parent ID

Attribute	Type	Remark
HM Remarks	Text – 50cc	House master's remarks
HOD	Text – 8cc	Head of department – Staff ID
Home Add	Text – 90cc	Address of primary residence
Home Phone	Text – 20cc	Phone number at primary residence
House	Text – 5cc	House ID
House ID	Text – 5cc	Unique system generated ID for House
House Master	Text – 8cc	Staff ID
Initials	Text – 6cc	Initials of the forenames
Invigilator	Text – 8cc	Staff ID
ISSN or ISBN	Text – 20cc	
Language	Text – 10cc	Primary language spoken at home
Last Year	Integer	Last academic year
Level	Text – 2cc	Academic level of candidates e.g. O, F4 or F1
Loan Period Rule	Text – 90cc	Rule for specifying the date and time at which the book will be due
Location ID	Integer	System generated index
Location Name	Text – 20cc	E.g. Main Library Reserve
M Name	Text – 20cc	Second/Middle name
MA Name	Text – 20cc	Official name of the medical aid society e.g. CIMAS
MA Number	Text – 20cc	Medical aid number
MA Type	Text – 20cc	Official name of the scheme to which the student is a member e.g. Premium Plan
MA Type ID	Text – 5cc	Unique system generated ID number for medical aid type.
Marital Status	Text – 10cc	
Mark	Float	
Membership Rule	Text – 90cc	SQL for specifying members who qualify for the category
Minimum Balance	Currency	Minimum balance for the Deposit Account type
Mobile Phone	Text – 20cc	Contact mobile phone number
Mother	Text – 8cc	Parent ID
NCT	Text – 1cc	Number in current term e.g. 3 for 3rd assessment report in the first term.
Occupation	Text – 30cc	Official job title
Overdue Fine Rule	Text – 90cc	Rule for determining the fine
Paper Code	Text – 8cc	User specified unique indexing key
Paper Title	Text – 20cc	Official title of the examination paper
Password	Text – 20cc	
Patron ID	Text – 8cc	System generated index
Period Ends	Date	Date on which period covered by transfers ends.
Period Starts	Date	Date on which period covered by transfers starts.

Attribute	Type	Remark
Phone	Text – 20cc	Contact Phone Number
PMS	Text – 10cc	Parents' marital status [Married | Divorced | Widowed | Single]
Publication Year	Integer	
R Effective	Date	Date on which the request becomes effective
R Expires	Date	Date on which request expires
Reason	Text – 12cc	Reason for deregistration [Completed | Transferred | Died | Expelled]
Receipt	Text – 15cc	Receipt Number
Recurrence Rule	Text – 50cc	Rule for specifying how the event will recur
Ref Code	Text – 15cc	Reference code of letter from disciplinary committee authorising debit.
Religion	Text – 15cc	Family's religion
Remark	Text – 50cc	Free text comment
Role	Text – 15cc	Supervisor's role in the club e.g. U16 Coach
S Expires	Date	Date on which suspension expires
Section ID	Integer	System generated index
Section Name	Text – 20cc	E.g. Fiction
Session ID	Text – 8cc	Unique system generated index
Sex	Text – 1cc	M – Male; F – Female
SH Remarks	Text – 50cc	School head's remarks
ST Remark	Text – 50cc	Subject teacher's comment
Start date	Date	Date on which period covered by schedule entry begins.
Start Time	Time	Time at which recurring event starts
Starts	Date	Starting time. Includes time information
Student Age	Text – 4cc	Student's age in complete years and months. First two letters for years and the last two for the months.
Student ID	Text – 8cc	Unique Student ID Code
Subject Avg Marks	Float	Average of the students' individual marks
Subject Code	Text – 8cc	Unique system generated ID code
Subject Position	Integer	Position in subject based on subject marks
Subject Title	Text – 20cc	Subject's official title e.g. F4A1 English
Supervisor	Text – 8cc	Staff ID
Surname	Text – 20cc	Family name
Teacher	Text – 8cc	Staff ID
Teacher Initials	Text – 6cc	Initials of the subject teacher including both forenames and surname.
Template Data	Memo	A string of fee codes and amounts in the format {Fee Code, Fee Amount;}
Template Name	Text – 30cc	A unique descriptive name for each template e.g. F1 & F2 Registration Fees

Attribute	Type	Remark
Term	Text – 1cc	School term [1 – 3]
Title	Text – 5cc	Name title E.g. Prof
Title	Text – 30cc	Book's title
Transaction Date	Date	Includes time information
Updated	Date	Date record was last updated
User Name	Text – 8cc	Student, Staff or Patron ID
Venue	Text – 10cc	
Venue	Text – 15cc	Venue of event
Year	Integer	Calendar year

ARCHITECTURAL DESIGN
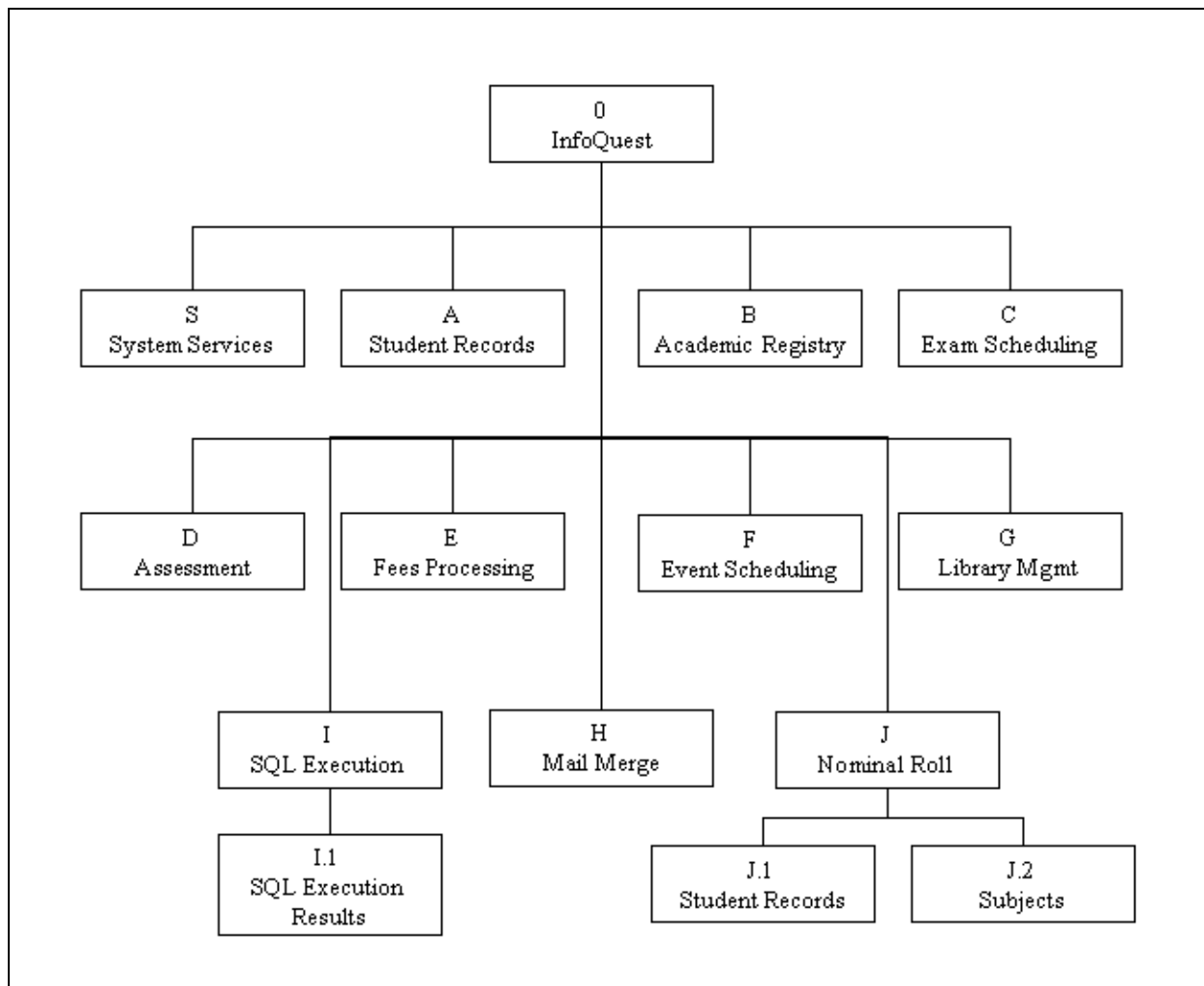
Figure 10: Level 0 Structure Diagram
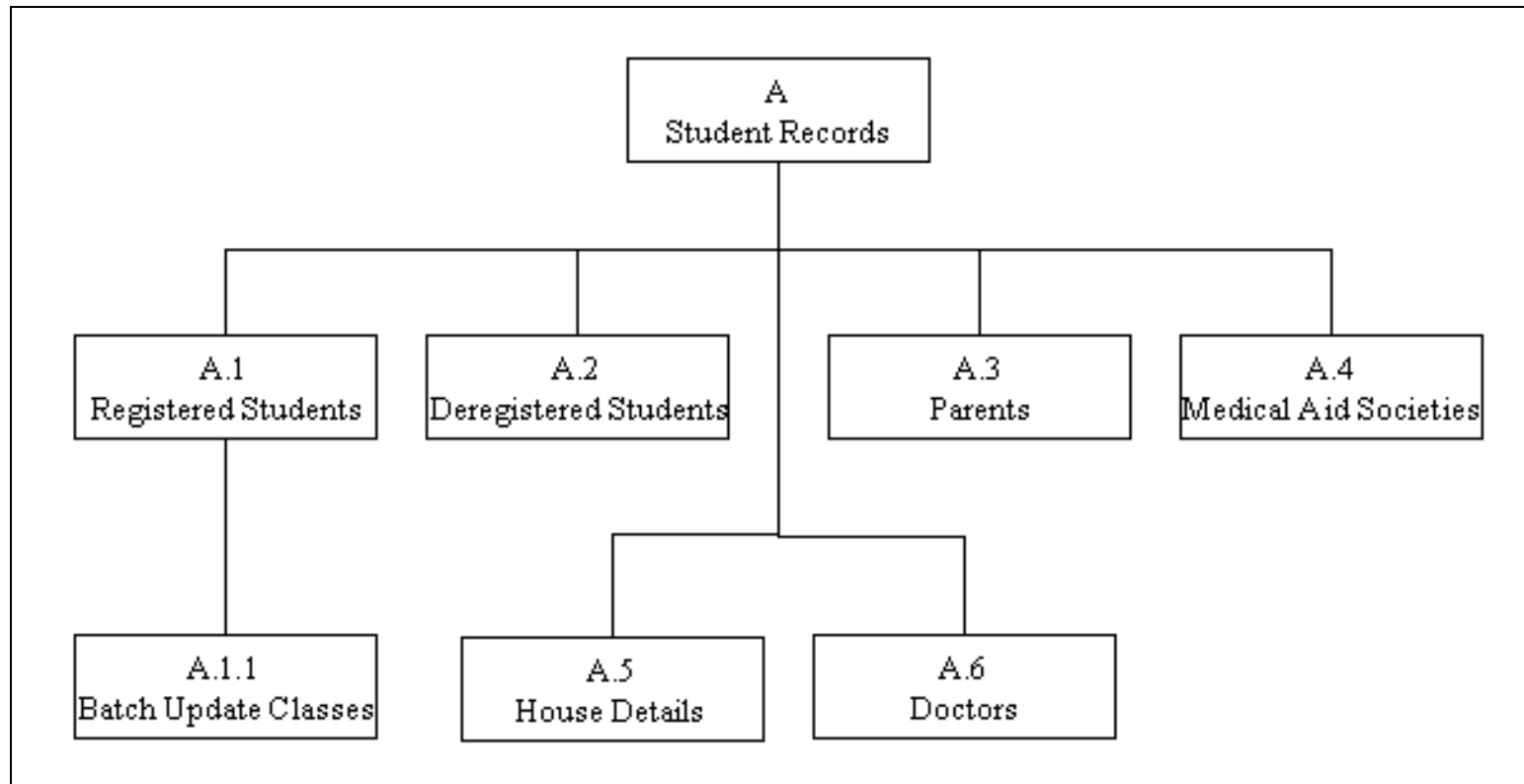
Figure 11: A - Student Records
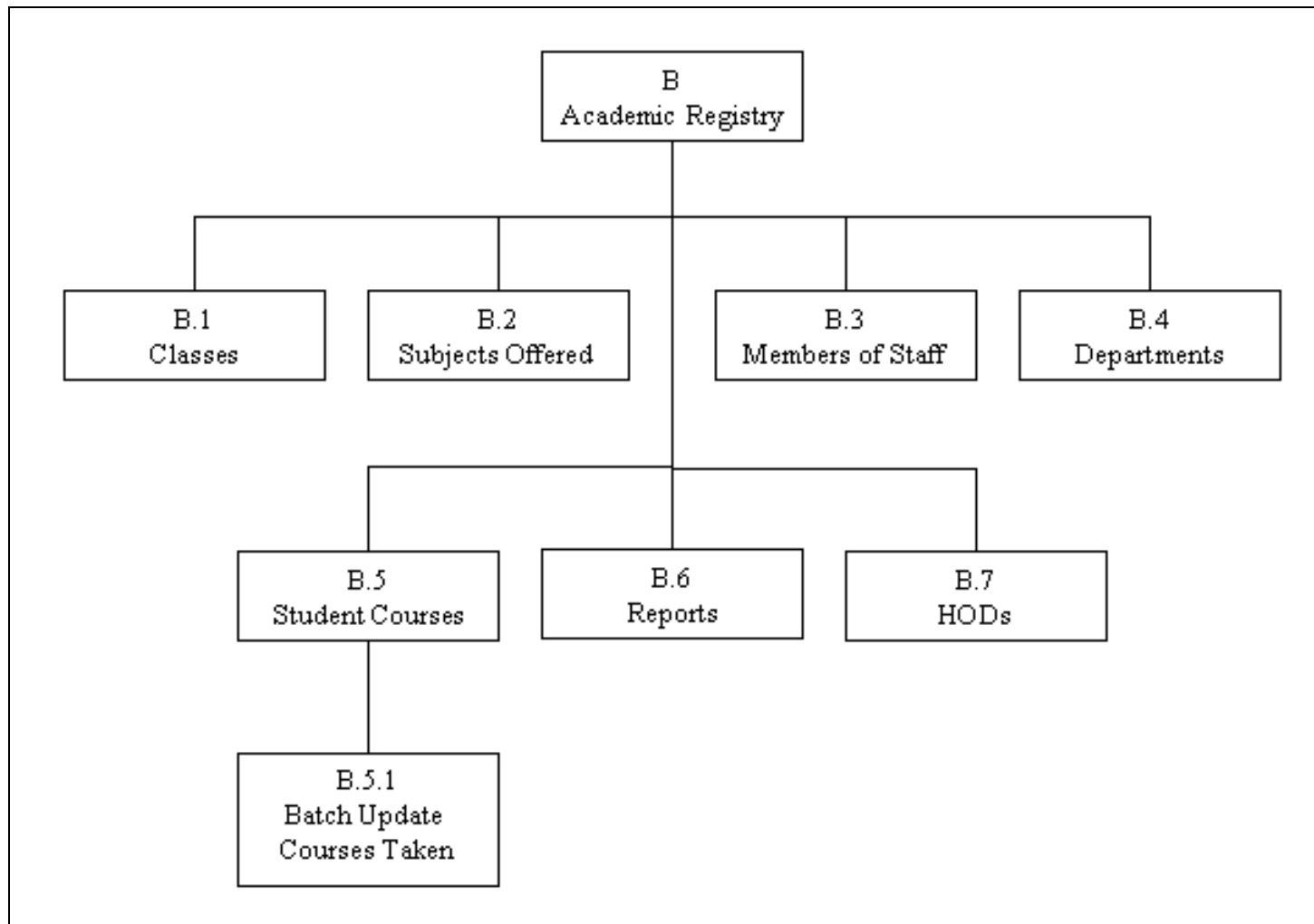
Figure 12: B - Academic Registry
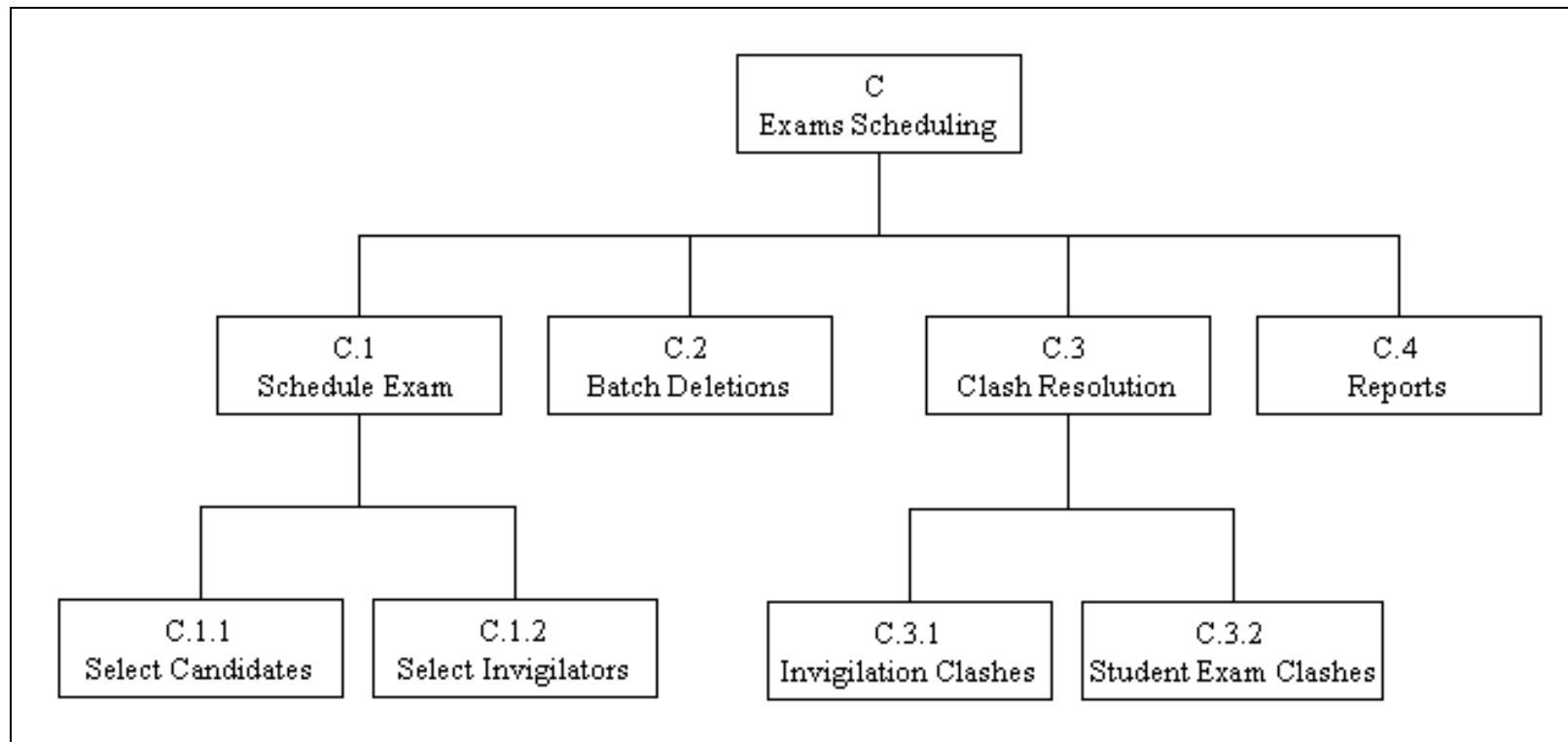
Figure 13: C - Exams Scheduling
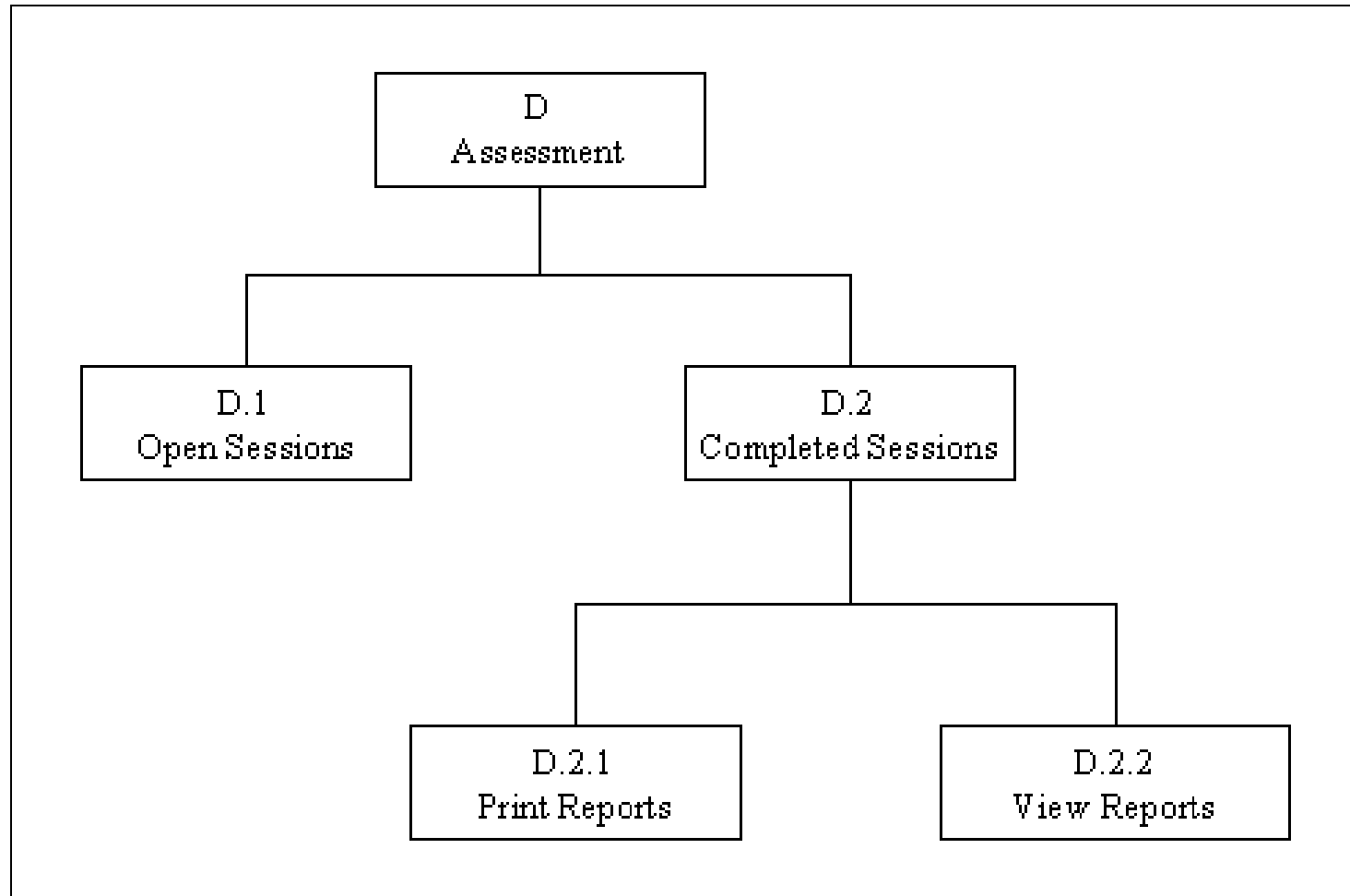
Figure 14: D - Assessment
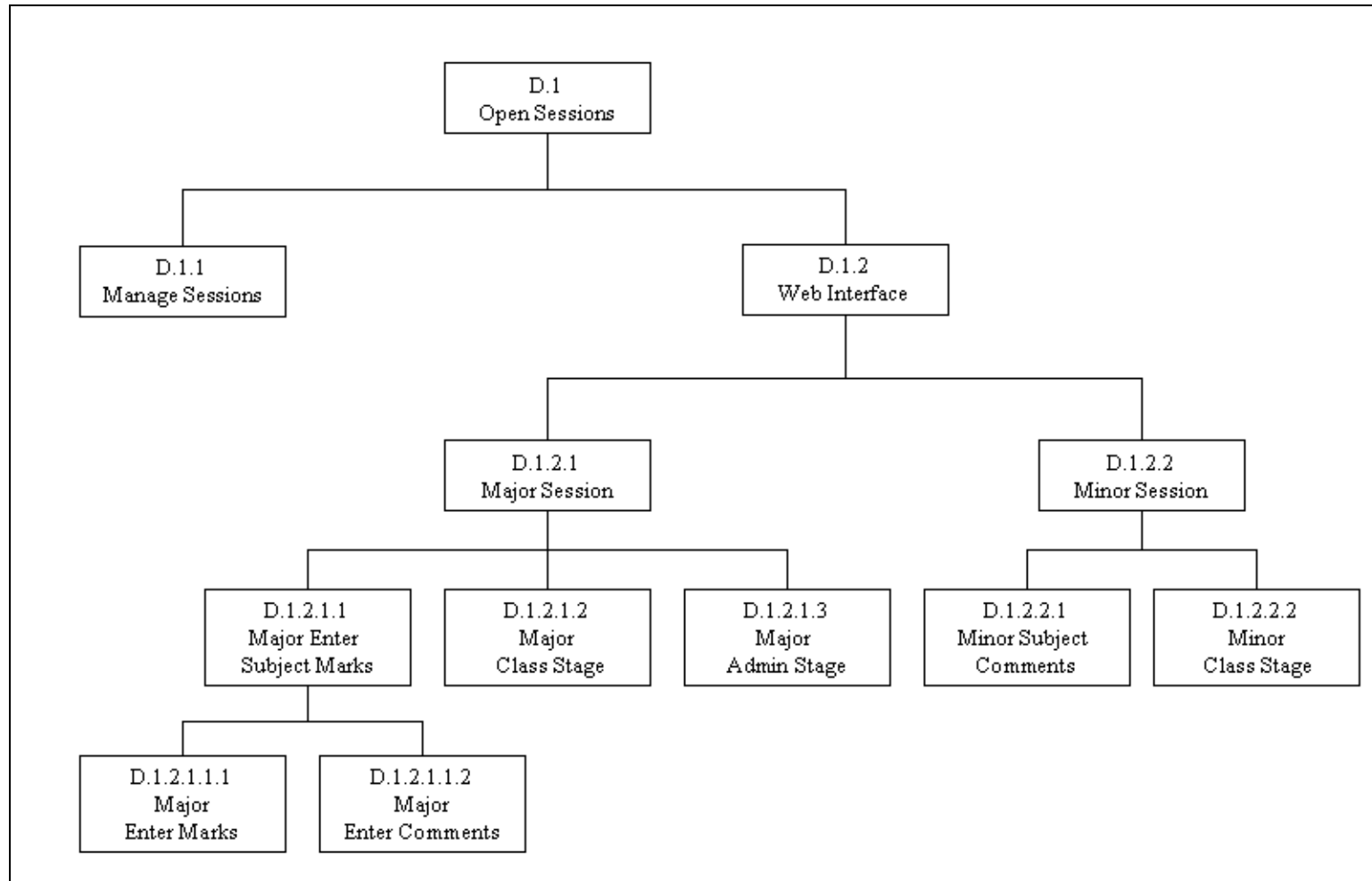
Figure 15: D1 - Open Sessions
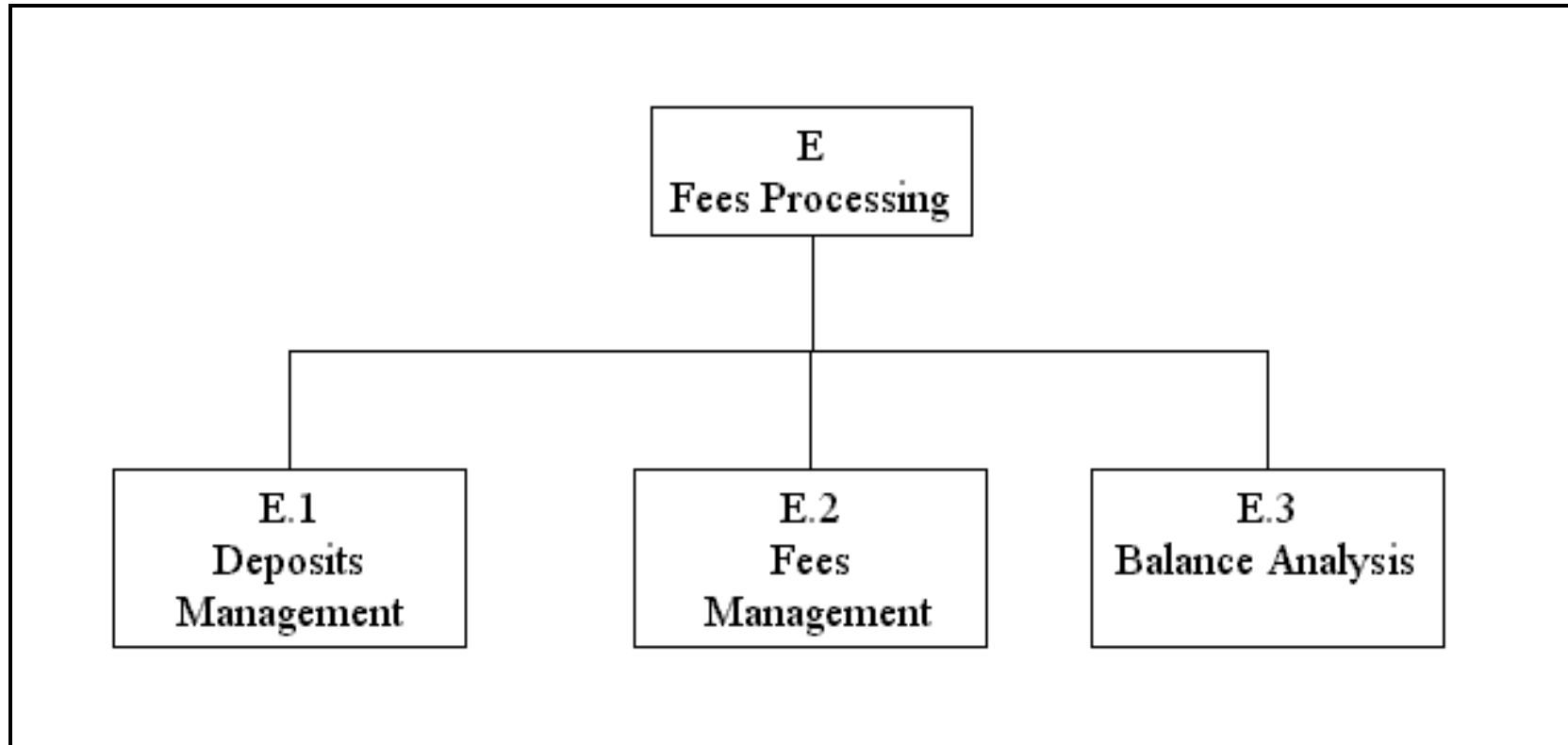
Figure 16: E - Fees Processing
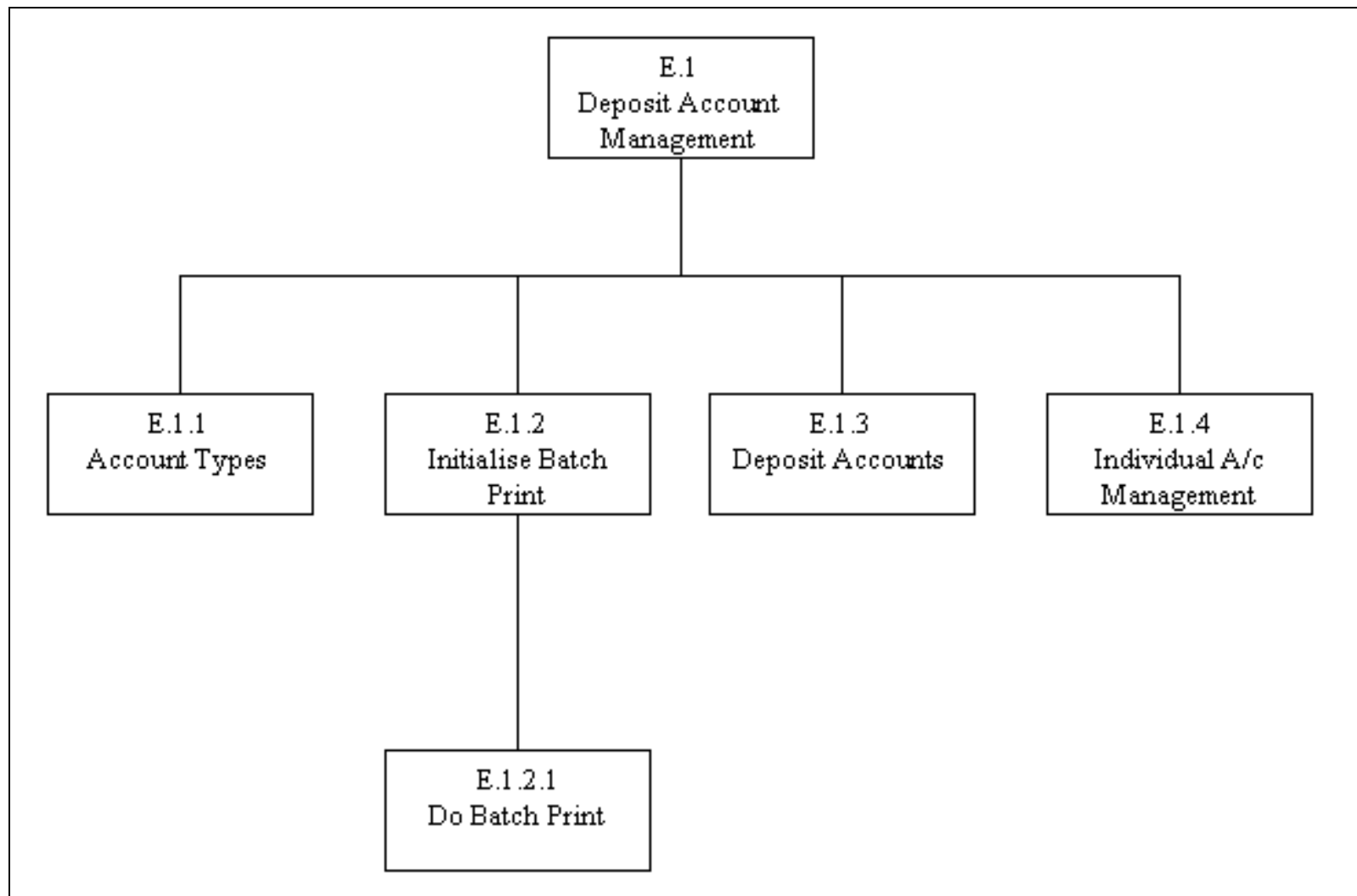
Figure 17: E1 - Deposit Account Management
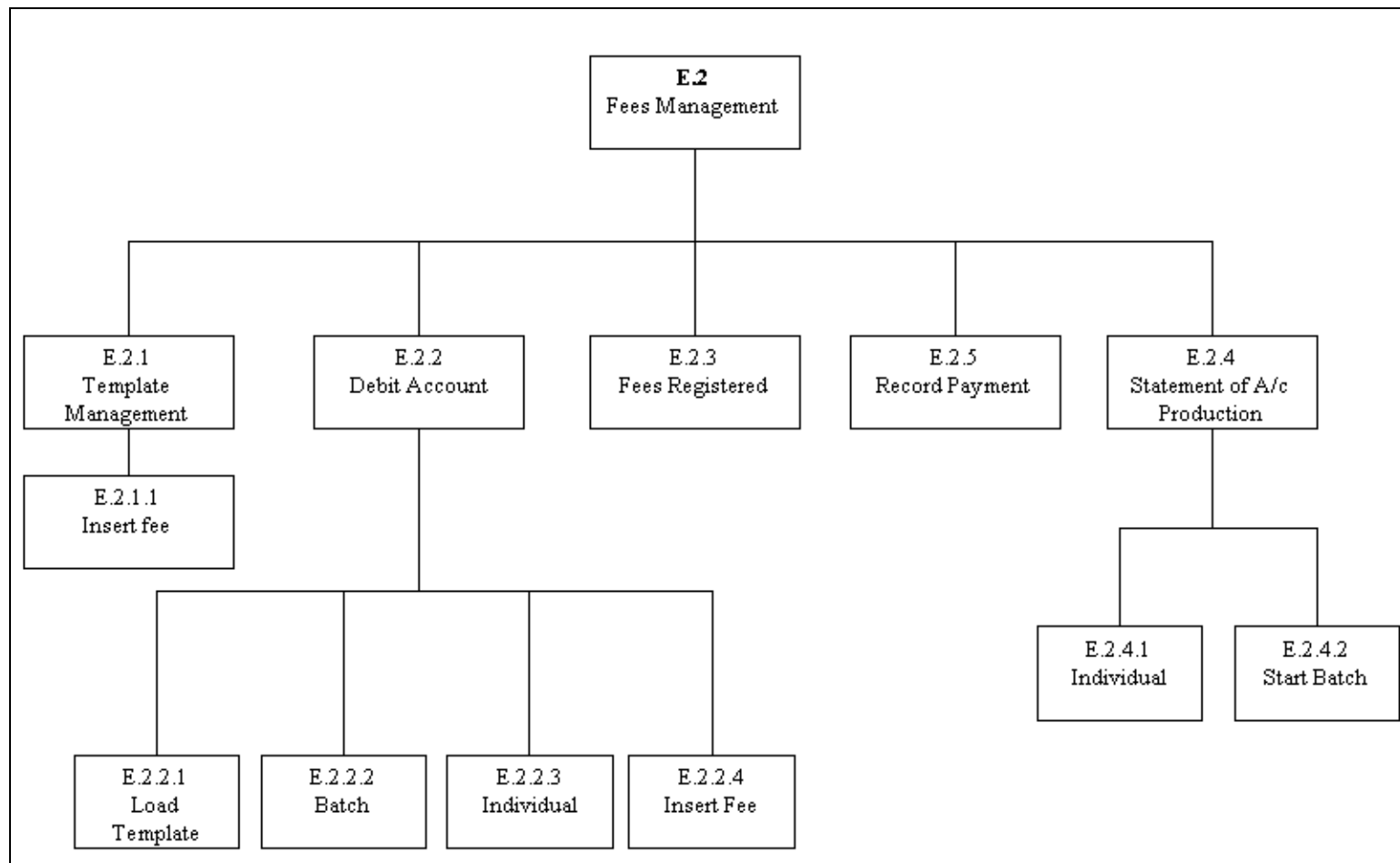
Figure 18: E2 - Fees Management
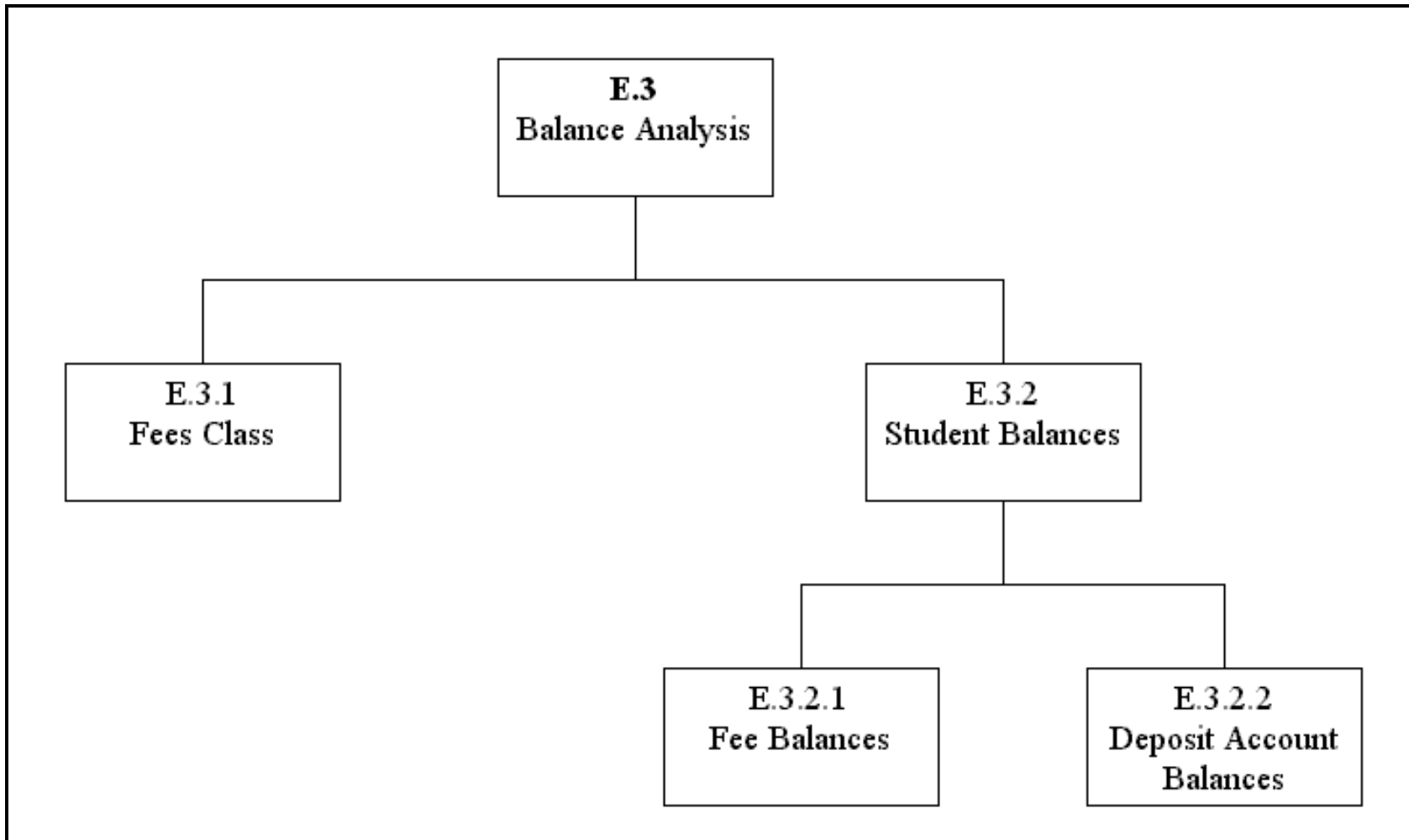
Figure 19: E3 - Balance Analysis

75
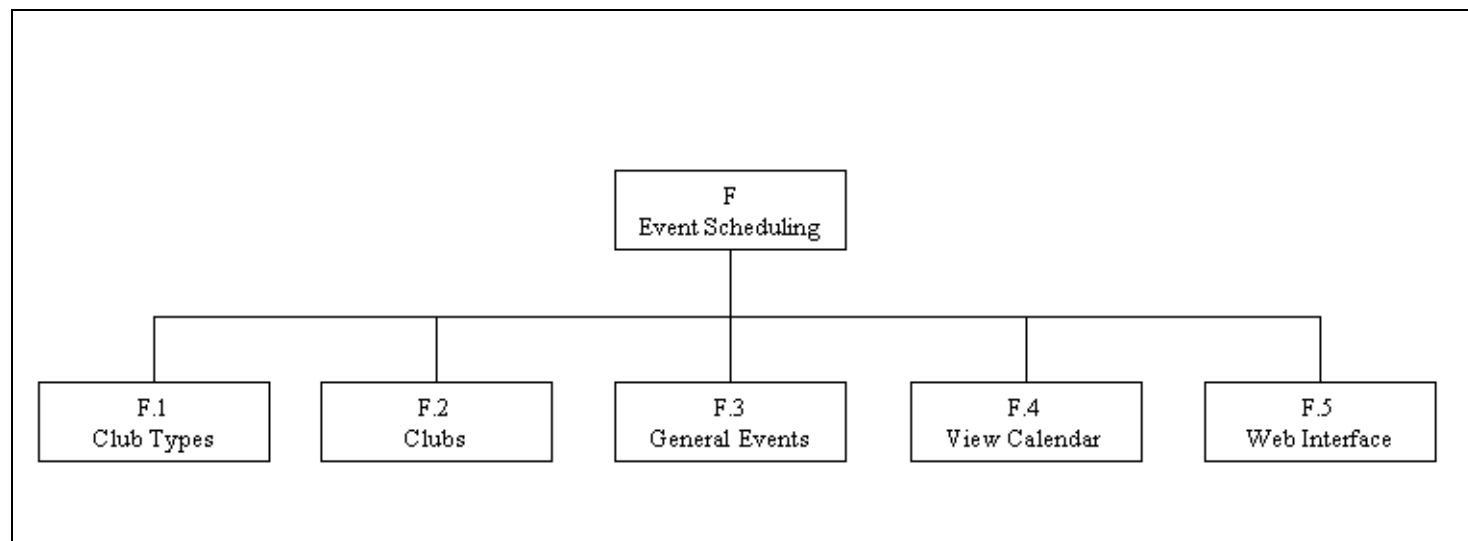
Figure 20: F - Event Scheduling
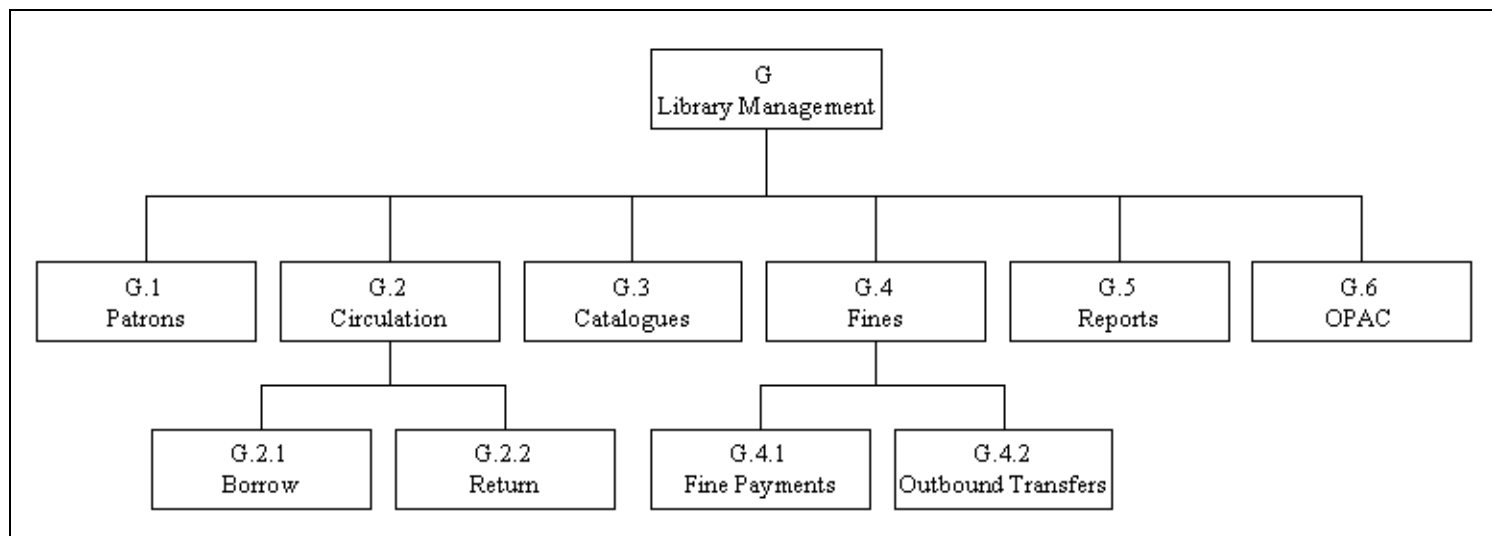
Figure 21: G - Library Management
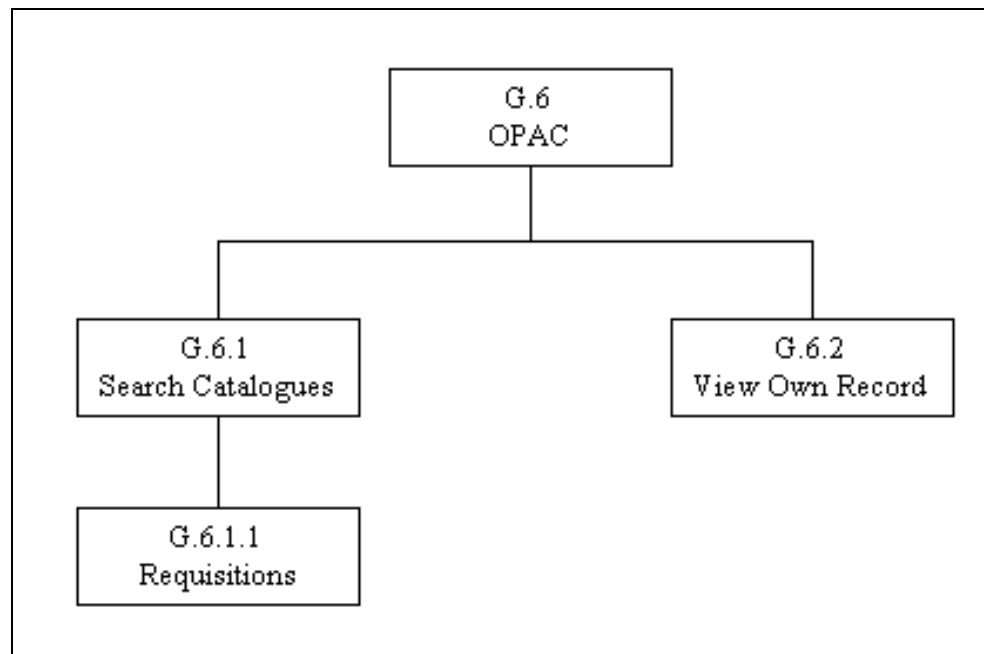
Figure 22: G6 - OPAC
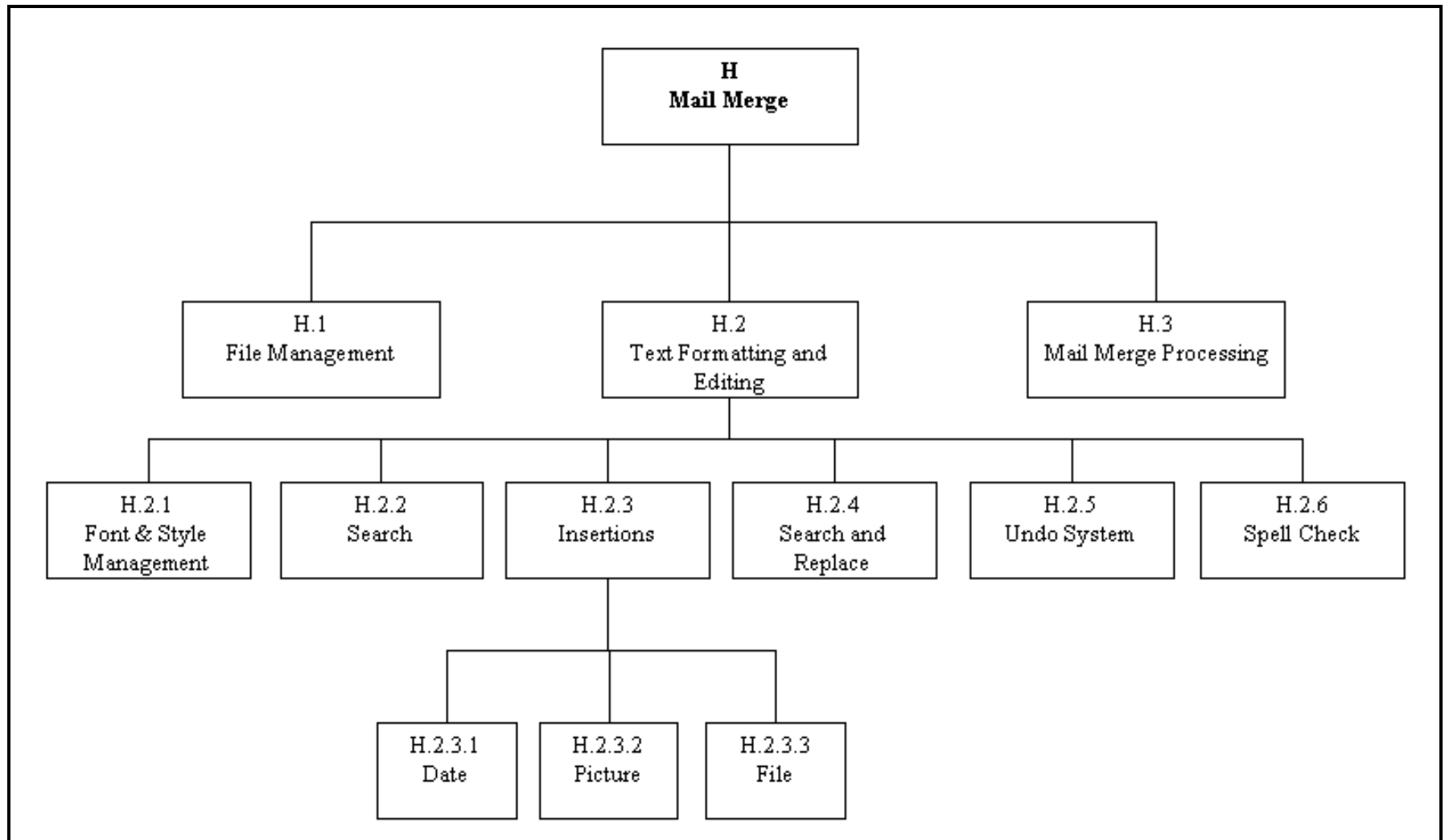
Figure 23: H - Mail Merge

79
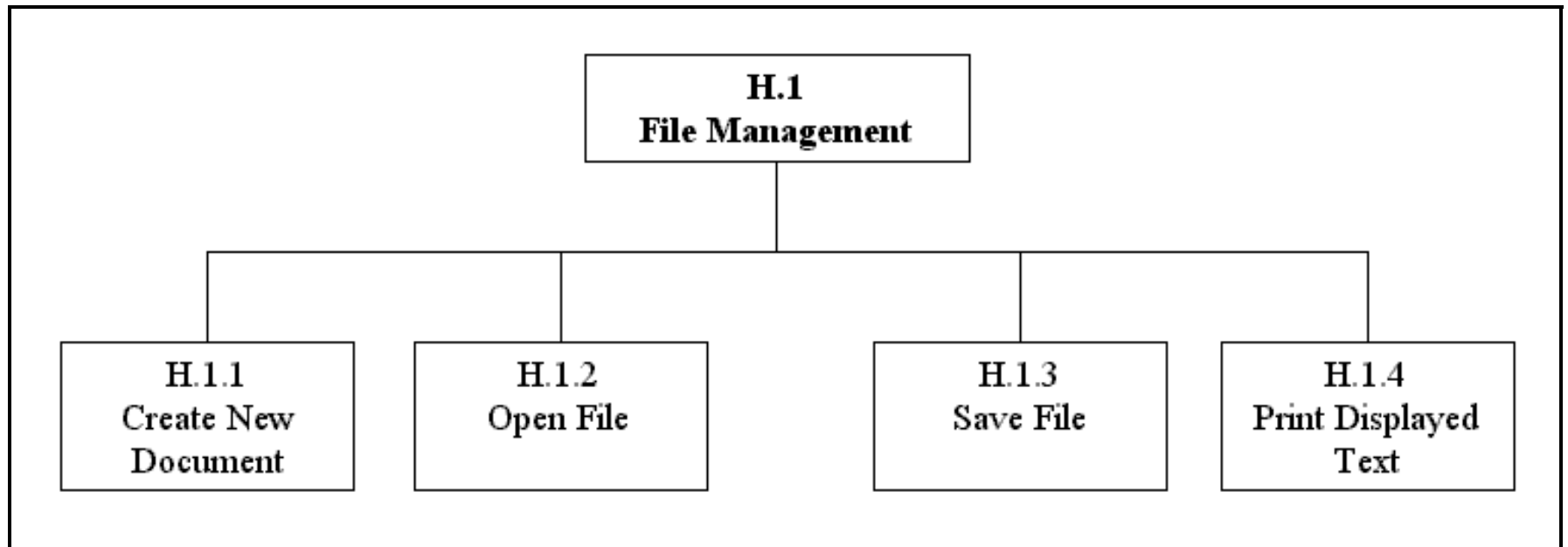
Figure 24: H1 - File Management
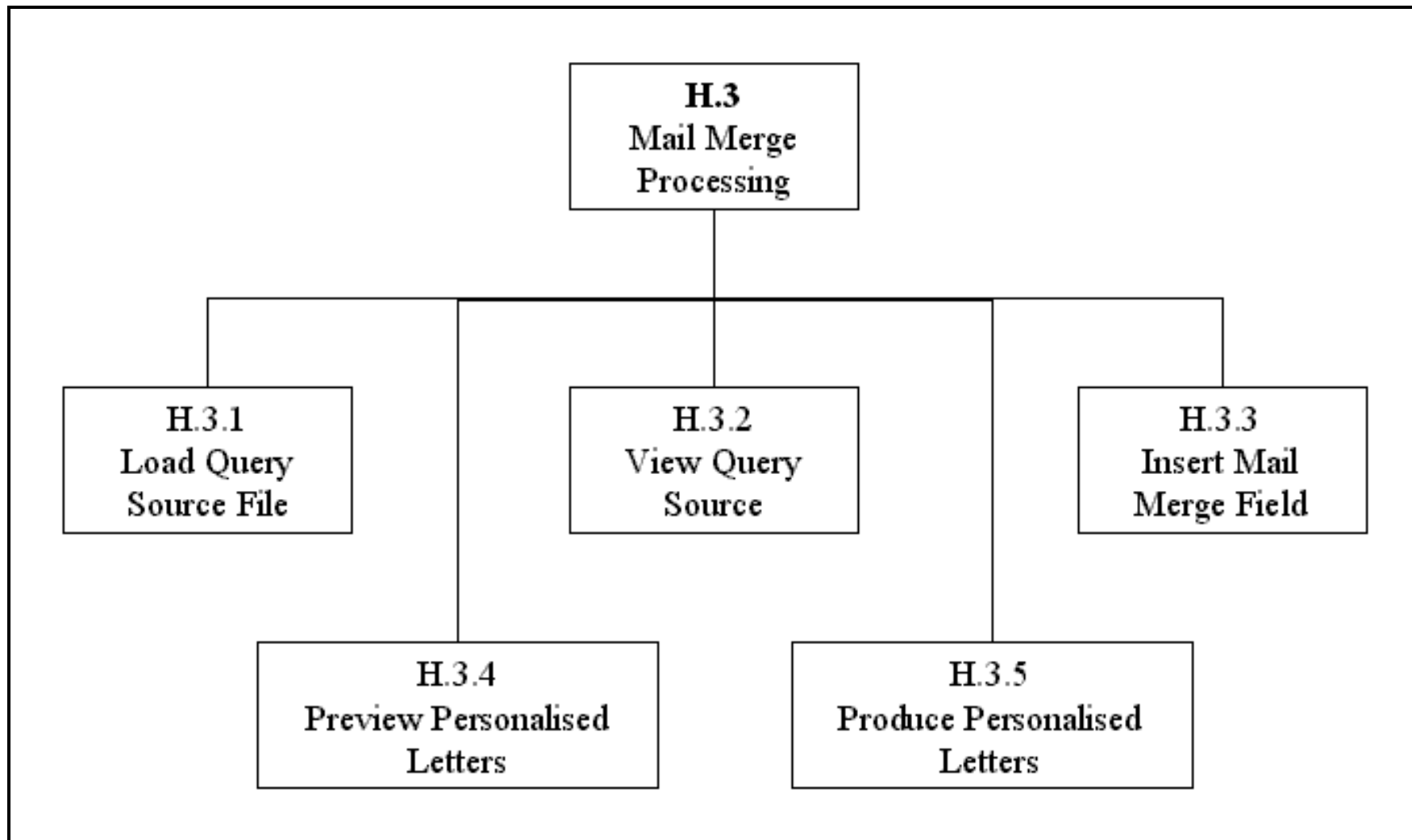
Figure 25: H3 - Mail Merge Processing
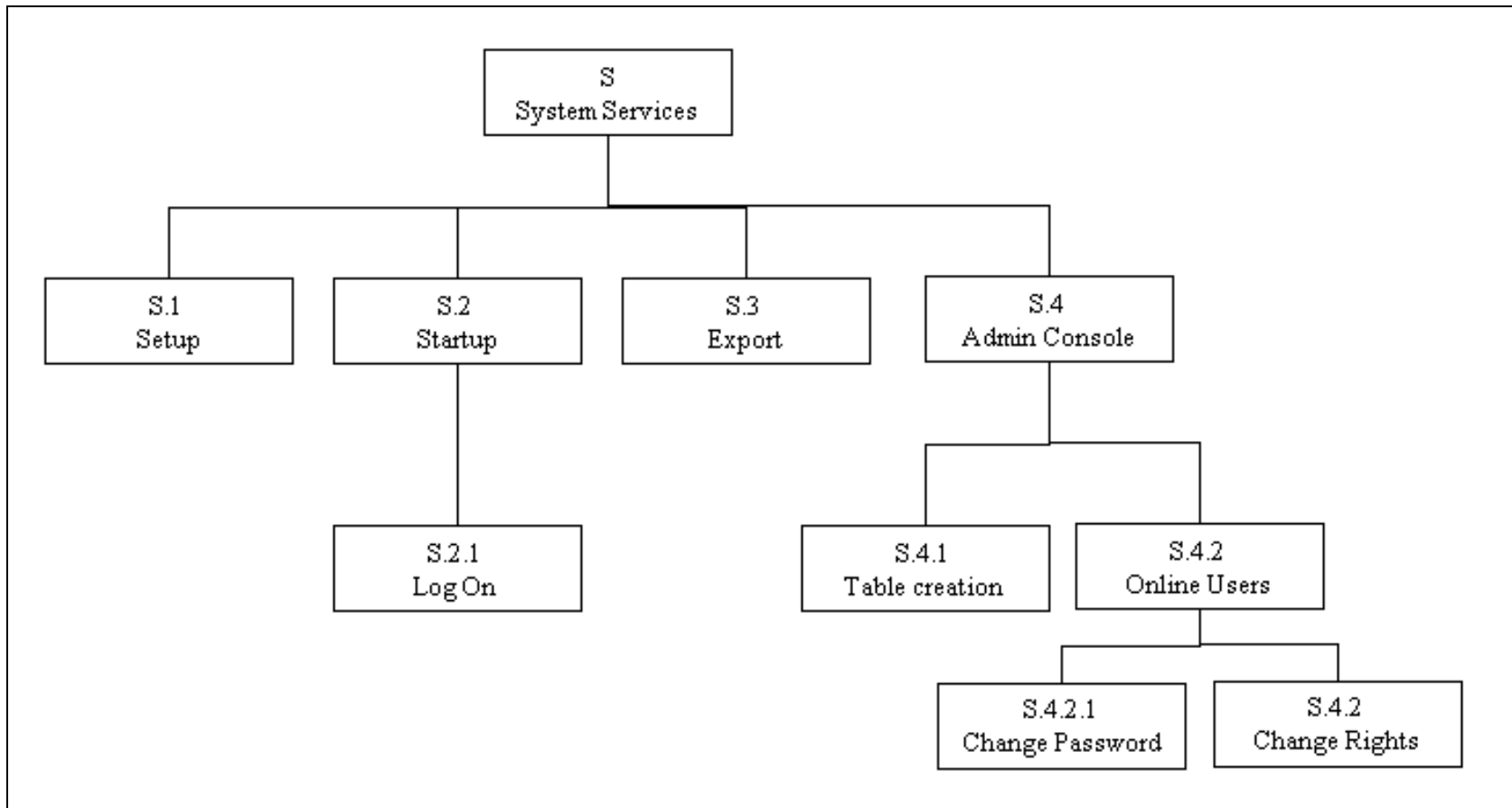
Figure 26: S - System Services

COMPONENT DESIGN

0 - infoQuest

Rationale
Provides access to all the other components

Parent
None

Descendants
S – System services
A – Student Records
B – Academic Registry
C – Exam Scheduling
D – Assessment
E – Fees Processing
F – Event Scheduling
G – Library Management
H – Mail Merge
I – SQL Execution
J – Nominal Roll

Dependencies
None

User Interface
Windows Program Group

Algorithms
None

A – Student Records

Rationale
Provides access to descendents

Parent
0 - infoQuest

Descendants
A.1   - Registered students
A.2   - Deregistered students
A.3   - Parents
A.4   - Medical Aid Societies
A.5   - House Details
A.6   - Doctors

Dependencies
S2   - Used to obtain connection to database

User Interface

Input Name	Input Type	Remarks
Registered Students	Button	• Access only if the Class, Medical Aid Societies, Parents and Doctors Tables are not empty
Deregistered Students	Button	• Access only if the deregistered students table is not empty
House Details	Button	• Access only if Members of staff table is not empty

Algorithms
None

A.1 – Registered Students

Rationale
Managing student records as to requirement A1

Parent
A   - Student Records

Descendants
A.1.1 - Batch Update Classes

Dependencies
None

User Interface

infoQuest: Students Records							
New	Edit	Deregister	Locate	Sort	Subset	Report	Batch Update

Student	Guardian	Father	Mother	Doctor	Medical Aid	Siblings	Class	House

Input Name	Input Type	Remarks
Locate	Button	• Finds a record
Sort	Button	• Sorts records
Subset	Button	• Displays a subset of the records
Batch Update	Button	• Call the batch Update Classes component

A.1.1 – Batch Update Classes

Rationale
Managing student records as to requirement A.7

Parent
A.1   - Registered Students

Descendants
None

Dependencies
None

User Interface

Batch Update Classes
Current Class [▼]
New Class [▼]
[Update]   [Help]   [Cancel]

A.2 – Deregistered Students

Rationale
Managing student records as to requirement A2

Parent
A   - Student Records

Descendants
None

Dependencies
A1   - Registered students (Called to handle the reregistering of a student)

User Interface

infoQuest: Students Records						
New	Edit	Reregister	Locate	Sort	Subset	Report

Student Guardian Father Mother House

Input Name	Input Type	Remarks
Locate	Button	• Finds a record
Sort	Button	• Sorts records
Subset	Button	• Displays a subset of the records
Reregister	Button	• Moves the record to the registered students table

A.3 - Parents

Rationale
Managing student records as to requirement A3

Parent
A   - Student Records

Descendants
None

Dependencies
None

User Interface
Simple listing of all details with buttons for supporting access methods

A.4 – Medical Aid Societies

Rationale
Managing student records as to requirement A4

Parent
A   - Student Records

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

A.5 – House Details

Rationale
Managing student records as to requirement A6

Parent
A   - Student Records

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

A.6 – Doctors

Rationale
Managing student records as to requirement A5

Parent
A   - Student Records

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

B – Academic Registry

Rationale
Provides access to descendents

Parent
0 - infoQuest

Descendants
B.1   - Classes
B.2   - Subjects Offered
B.3   - Teachers
B.4   - Departments
B.5   - Student Courses
B.6   - Reports
B.7   - HODs

Dependencies
S2   - Used to obtain connection to database

User Interface

infoQuest: Academic Registry	
Classes	
Subjects Offered	
Members of Staff	
Departments	
HODs	
Course Assignments	
Reports	Help Log Off Exit

Input Name	Input Type	Remarks
Members of Staff	Button	• Access only if Departments table is not empty
HODs	Button	• Access only if Departments table is not empty
Course Assignments	Button	• Access only if Student Subjects are not empty
Subjects offered		• Access only if members of Staff table is not empty
Classes	Button	• Access only if members of Staff table is not empty

Algorithms
None

B.1 - Classes

Rationale
Managing class records as to requirement B.1.a

Parent
B   - Academic registry

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

B.2 – Subjects Offered

Rationale
Managing subjects' records as to requirement B.1.b

Parent
B   - Academic registry

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

B.3 – Members of Staff

Rationale
Managing members of staff records as to requirement B.1.c

Parent
B   - Academic registry

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

B.4 - Departments

Rationale
Managing Department records as to requirement B.1.d

Parent
B   - Academic registry

Descendants
None

Dependencies
None

User Interface
Simple   listing   of   all   details   with   buttons   for   supporting   access   methods

B.5 – Student Courses

Rationale
Managing student-subject associations as to requirement B.1.e

Parent
B   - Academic registry

Descendants
B.5.1   - Batch Update Courses Taken

Dependencies
None

User Interface
1. Subject View
    ➢ List all the students taking a given subject
    ➢ Allow user to add or remove students from the subject list
2. Student View
    ➢ List all the subject being taken by a given student
    ➢ Allow   user   to   add   or   remove   subjects   from   the   list

B.5.1 – batch Update Courses Taken

Rationale
Requirement B.2

Parent
B.5   - Student Courses

Descendants
None

Dependencies
None

User Interface
Wizard interface

Algorithm
1. Select from one of the following
     - o Replace Course
     - o Add course
     - o Remove course
2. If replacing course,
     - o Select course or courses to be replaced
     - o Specify course or courses to replace with
     - o Confirm replacement
     - o Do replacement
3. If adding course
     - o Specify selection criteria in terms of a combination of the following
         - ▪ Academic year
         - ▪ Class
         - ▪ Courses already taken
     - o Specify whether the criteria is based on the student taking all of the courses specified or a given number of those specified
     - o Specify course or courses to add
     - o Confirm addition
     - o Do Addition

4. If removing course
    o Specify selection criteria in terms of a combination of the following
        ▪ Academic year
        ▪ Class
    o Specify course or courses to remove
    o Confirm deletion
    o Do deletion

B.6 – Reports

Rationale
Producing the reports specified in requirement B3

Parent
B   - Academic registry

Descendants
None

Dependencies
None

User Interface

```
┌────────────────────────────────────────────────────────┐
│          ┌─────────────────────────────┐               │
│          │       Prepare Reports        │               │
│          ├─────────────────────────────┴──────┐        │
│                                                         │
│   Select Report    ┌───────────────────────┬──┐        │
│                     │                       │ ↓│        │
│                     └───────────────────────┴──┘        │
│                                                         │
│        ┌──────────┐    ┌──────┐    ┌────────┐          │
│        │ Generate │    │ Help │    │ Cancel │          │
│        └──────────┘    └──────┘    └────────┘          │
│                                                         │
└────────────────────────────────────────────────────────┘
```

B.7 – HODs

Rationale
Managing the list of HODs as to requirement B.1.d

Parent
B   - Academic registry

Descendants
None

Dependencies
None

User Interface
List of fields

C – Exam Scheduling

Rationale
Provides access to descendents

Parent
0 - infoQuest

Descendants
C.1   - Schedule Exam
C.2   - Batch Deletions
C.3   - Clash Resolutions
C.4   - Reports

Dependencies
S2   - Used to obtain connection to database

User Interface

Input Name	Input Type	Remarks
Schedule Exams	Button	• Student Courses must not be empty
Resolve Clashes	Button	• Access only if Students Exams table is not empty
Batch Deletions	Button	• Access only if Students Exams table is not empty
Reports	Button	• Access only if Students Exams table is not empty

C1 – Schedule Exams

Rationale
Requirements C1 and C2

Parent
C   - Exam Scheduling

Descendants
C.1.1   - Select Candidates
C.1.2   - Select Invigilators

Dependencies
None

User Interface

C.1.1 – Select Candidates

Rationale
Requirements C.1.g and C.2

Parent
C.1   - Schedule Exams

Descendants
None

Dependencies
None

User Interface
Wizard Interface

Algorithm
1. Select Selection to be on the basis of one of the following
      a. Individual Student
      b. Classes of students
      c. Subjects
2. If selection on the basis of individual student
      a. Provide form to search for student on the basis of
            i. Surname
            ii. Registration number
3. If selection on the basis of class
      a. Provide list boxes, Available and Selected, for selecting the classes
4. If selection on the basis of subjects,
      a. Provide list boxes, Available and Selected, for selecting the subjects
5. Provide list of students found who match the criteria and allow selection of student by selecting from an **Available** grid to a **Selected** grid. For each selection, ensure that the student is not already entered for the exam by checking the grid on the parent form. Allow multiple select.
6. Enter the student into the *parent*
7. Redisplay parent form

C.1.2 – Select Invigilators

Rationale
Requirements C.1.f

Parent
C.1   - Schedule Exams

Descendants
None

Dependencies
None

User Interface
Wizard Interface

Algorithm
8. Select Selection to be on the basis of one of the following
    a. Individual Teacher
    b. Teachers in a given Department
    c. Teachers for a given set of Subjects
9. If selection on the basis of an individual teacher
    a. Provide form to search for the member of staff on the basis of
        i. Surname
        ii. Registration number
10. If selection on the basis of departments
    a. Provide list boxes, Available and Selected, for selecting the departments
11. If selection on the basis of subjects,
    a. Provide list boxes, Available and Selected, for selecting the subjects
12. Provide list of members of staff found who match the criteria and allow selection of student by selecting from an **Available** grid to a **Selected** grid. For each selection, ensure that the member of staff is not already entered as an invigilator for the exam by checking the parent form. Allow multiple select.
13. Enter the members of staff into the parent form.
14. Redisplay parent form

C.2 – Batch Deletions

Rationale
Requirements C.2.b

Parent
C   - Exam Scheduling

Descendants
None

Dependencies
None

User Interface
Wizard Interface

Algorithm
15. Select Selection to be on the basis of one of the following
    a.   Entries for given papers
    b.   Exam scheduled to be sat over a given period
16. If selection on the basis of examination papers
    a.   Provide grids for selecting the papers i.e. an Available and a Selected grid
    b.   Add all the papers to the Available Grid
17. If selection on the basis of a period of time
    a.   Provide a form for selecting the time period
    b.   Provide grids for selecting the papers i.e. an Available and a Selected grid
    c.   Add the papers that match the criteria to the Available grid
18. Allow the selection of the papers
19. Confirm batch deletion
20. Do the batch deletion and exit the wizard

C.3 – Clash Resolutions

Rationale
Abstract grouping object

Parent
C   - Exam Scheduling

Descendants
C.3.1   - Invigilation Clashes
C.3.2   - Student Exam Clashes

Dependencies
None

User Interface
None

Algorithm
None

C.3.1 – Invigilation Clashes

Rationale
Requirements C.3.c

Parent
C.3   - Clash Resolution

Descendants
None

Dependencies
None

User Interface

infoQuest: Invigilation Clashes

Invigilators

Staff ID	Name	Paper 1	Paper 2

Paper Details

Resolution

● Remove Paper 1
○ Remove Paper 2
○ Substitute Paper 1 with [▼]
○ Substitute Paper 2 with [▼]

[Resolve]

Notes to the Interface
1. Paper details
   a. Displays the data for the current paper 1 and paper 2
   b. Changes when the selected invigilator changes
2. Lists for substitution of papers should be based on papers not clashing with the paper not be replaced and all the other papers that the member of staff will be invigilating.

C.3.2 – Student Exam Clashes

Rationale
Requirements C.3.a and C.3.b

Parent
C.3   - Clash Resolution

Descendants
None

Dependencies
None

User Interface – Student View

User Interface – Exam Paper View

infoQuest: Student Exam Clashes

Exam Papers With Clashes

Students

Student ID	Name	Clashes With

Exam Paper Details

Move Affected Students To

Resolve

C.4 – Reports

Rationale
Producing the reports specified in requirement C4

Parent
C   - Exam Scheduling

Descendants
None

Dependencies
None

User Interface

Prepare Reports

Select Report [↓]

Generate   Help   Cancel

D – Assessment

Rationale
Abstract grouping object

Parent
0   - infoQuest

Descendants
D.1   - Open Sessions
D.2   - Closed Sessions

Dependencies
None

User Interface
None

Algorithm
None

D.1 – Open Sessions

Rationale
Abstract grouping object

Parent
D   - Assessment

Descendants
D.1.1   - Manage Sessions
D.1.2   - Web Interface

Dependencies
None

User Interface
None

Algorithm
None

D.1.1 – manage Sessions

Rationale
Provides access to descendents

Parent
D.1   - Open Sessions

Descendants
None

Dependencies
None

User Interface

User Interface Notes
1. Open Sessions list all the open sessions.
2. Rest of the interface depends on the session currently selected in the list of open sessions.
3. NCT stands for Number In Current Term. For major sessions, the value should be one and not editable.
4. Stage upon the initiation of a new session should be Init. The field can only be changed by selecting Advance Stage.

5. The list for the available classes will contain a list of classes. The Selected list will contain two fields, one for the class and the other for the administrative member of staff responsible for entering the Admin comments. This will only apply if the Assessment session is a Major assessment session.
6. Upon selection from the Available list, the user should be presented with a wizard for selecting the administrator by searching using a Staff ID or surname. This will only apply if the Assessment session is a Major assessment session.

D.1.2 – Web Interface

Rationale
Provides secured access to descendents

Parent
D.1   - Open Sessions

Descendants
D.1.2.1- Major Sessions
D.1.2.2- Minor Sessions

Dependencies
None

User Interface
1. Log On Page
    a. User Name
    b. Password
2. List sessions open to the member of staff as to each of the following
    a. Major Assessment – Subject Stage
        i.   Session ID
        ii. Subject Code
        iii. Subject Title
    b. Major Assessment – Class stage
        i.   Session ID
        ii. Class being assessed
    c. Major Assessment – Admin Stage
        i.   Session ID
        ii. Class being assessed
    d. Minor Assessment – Subject Stage
        i.   Session ID
        ii. Subject Code
        iii. Subject Title
    e. Minor Assessment – Class Stage
        i.   Session ID
        ii. Class being assessed
3. Allow user to load the appropriate editing interface by clicking on the session links

D.1.2.1 – Major Assessment Session

Rationale
Abstract grouping object

Parent
D.1.2   - Web Interface

Descendants
D.1.2.1.1   - Major Enter Subject Marks
D.1.2.1.2   - Major Class Stage
D.1.2.1.3   - Major Admin Stage

Dependencies
None

User Interface
None

D.1.2.1.1 – Major Enter Subject Marks

Rationale
Abstract grouping object

Parent
D.1.2.1   - Major Session

Descendants
D.1.2.1.1.1   - Major Enter Marks
D.1.2.1.1.2   - Major Enter Comments

Dependencies
None

User Interface
None

D.1.2.1.1.1 – Major Enter Marks

Rationale
Requirement D.4.b

Parent
D.1.2.1.1   - Major Enter Subject Marks

Descendants
None

Dependencies
None

User Interface
Server generated web page with the following at the header
1. Name of the member of staff
2. Subject Title
3. Session ID
4. Session Type

Algorithm
1. If the database does not contain a record of the marks then display a list of the student each with a blank textbox for entering the mark using a server side script.
2. If the database contains a record of the marks then display a list of the students taking the subject with textboxes containing the mark previously recorded using a server side script.
3. Give the user an opportunity to enter/edit the marks.
4. Run a client side script to ensure that all the marks have been entered and that all the entries are valid numerical values between 0 and 100 inclusive.
5. Submit the form to a server side script given as component D.1.2.1.1.2

D.1.2.1.1.2 – Major Enter Comments

Rationale
Requirement D.4.b

Parent
D.1.2.1.1   - Major Enter Subject Marks

Descendants
None

Dependencies
D.1.2.1.1.1   - Supplies data used by the script to update the marks in the database

User Interface
Server generated web page with the following at the header
5. Name of the member of staff
6. Subject Title
7. Subject's Average Mark
8. Number of Students enrolled for the Subject
9. Session ID
10. Session Type

Algorithm
6. Update the marks entered in the submitting script.
7. Calculate the average mark for the subject
8. Calculate each student's position in the class
9. If the database does not contain a record of the comments then display a list of the student each with a blank textbox for entering the comment using a server side script.
10. If the database contains a record of the comments then display a list of the students taking the subject with textboxes containing the comment previously recorded using a server side script.
11. Give the user an opportunity to enter/edit the comments.
12. Run a client side script to ensure that all the comments have been entered.
13. Submit the form to a server side script to update the position, subject average and comments.

D.1.2.1.2 – Major Class Stage

Rationale
Requirement D.4.c

Parent
D.1.2.1   - Major Session

Descendants
None

Dependencies
None

User Interface
Server-side script generated html pages with the following
1. Main Page
    a. Class
    b. Form Teacher's name
    c. Session ID
    d. Session Type i.e. Major
    e. Number of Students In Class
    f. Class' Average Age
    g. Class' Average Mark
    h. For Each Student In The Class
        i.   Name of the Student (Hyper linked)
        ii. Registration Number
        iii. Age
        iv. Position In Class
        v.   Statement of whether all the comments have been entered
2. Sub Page for entering student's details
    a. Name of the Student
    b. Registration Number
    c. Age
    d. Class' Average Age
    e. Class
    f. Form Teacher's name
    g. Session ID
    h. Session Type i.e. Major
    i. Position In Class
    j. Number of Students In Class
    k. Student's Average Mark
    l. Class' Average Mark
    m. Hyperlink for viewing past reports in a separate window

n. Subject performance details
   i. Subject Title
   ii. Students Mark
   iii. Subject's Average Mark
   iv. Student's position in the subject
o. Fields for entering the following
   i. Number of Days Absent
   ii. Performance In Extra-curricula activities
   iii. General Form Teacher's remarks

D.1.2.1.3 – Major Admin Stage

Rationale
Requirement D.4.d

Parent
D.1.2.1   - Major Session

Descendants
None

Dependencies
None

User Interface
Server-side script generated html pages with the following
3. Main Page
    a. Class
    b. Form Teacher's name
    c. Session ID
    d. Session Type i.e. Major
    e. Number of Students In Class
    f. Class' Average Age
    g. Class' Average Mark
    h. For Each Student In The Class
        i. Name of the Student (Hyper linked)
        ii. Registration Number
        iii. Age
        iv. Position In Class
        v. Statement of whether all the comments have been entered
4. Sub Page for entering admin comment details
    a. Name of the Student
    b. Registration Number
    c. Age
    d. Class' Average Age
    e. Class
    f. Form Teacher's name
    g. Session ID
    h. Session Type i.e. Major
    i. Position In Class
    j. Number of Students In Class
    k. Student's Average Mark
    l. Class' Average Mark
    m. Number of Days Absent
    n. Hyperlink for viewing past reports in a separate window

o.   Subject performance details
  i.   Subject Title
  ii.   Students Mark
  iii.   Subject's Average Mark
  iv.   Student's position in the subject
p.   Form teacher remarks
  i.   Performance In Extra-curricula activities
  ii.   General Form Teacher's remarks
q.   A field for entering the admin remark

D.1.2.2 – Minor Assessment Session

Rationale
Abstract grouping object

Parent
D.1.2   - Web Interface

Descendants
D.1.2.2.1   - Minor Subject Comments
D.1.2.2.2   - Minor Class Stage

Dependencies
None

User Interface
None

D.1.2.2.1 – Minor Subject Comments

Rationale
Requirement D.4.b

Parent
D.1.2.1.1   - Major Enter Subject Marks

Descendants
None

Dependencies
None

User Interface
Server generated web page with the following
1. Main Page
    a. Header
        i.   Name of the subject teacher
        ii. Subject Title
        iii. Number of Students enrolled for the Subject
        iv. Session ID
        v.   Session Type
    b. List of students with the following details for each student
        i.   Registration Number
        ii. Class
        iii. Details of whether a comment has been entered or not
2. Sub Page for entering student comment
    a. Details under Header above
    b. Details under Student Details above
    c. Hyperlink for opening past reports in a separate window
    d. Fields for entering the following
        i.   Comment
        ii. Effort symbol

D.1.2.2.2 – Minor Class Stage

Rationale
Requirement D.4.c

Parent
D.1.2.1.1   - Major Enter Subject Marks

Descendants
None

Dependencies
None

User Interface
Server generated web page with the following
3. Main Page
    a. Header
        i.   Name of the subject teacher
        ii. Subject Title
        iii. Number of Students enrolled for the Subject
        iv. Session ID
        v.   Session Type
    b. List of students with the following details for each student
        i.   Name (Hyper Linked)
        ii. Registration Number
        iii. Class
        iv. Details of whether a comment has been entered or not
    c. Hyperlink for opening past reports in a separate window
    d. Subject performance
        i.   Subject teacher's Comment
        ii. Effort symbol
4. Sub page for entering the form teacher's remarks

D.2 – Completed Sessions

Rationale
Abstract grouping object

Parent
D   - Assessment

Descendants
D.2.1   - Print Reports
D.2.2   - View Reports

Dependencies
None

User Interface
None

Algorithm
None

D.2.1 – Print Reports

Rationale
Requirement D.6

Parent
D.2   - Completed Sessions

Descendants
None

Dependencies
None

User Interface
Wizard Interface

Algorithm
1. Select the type of report to print
2. Select the basis of the report printing
     a. Student
     b. Session
3. If student based session,
     a. Select between
          i.   Single report
          ii. All reports between a given period
     b. If a single report is required, provide a list to select from
     c. If all the reports over a given period are required,
          i.   Provide a means for specifying the start
          ii. Display all the reports that match the criteria and allow the user to select the reports desired
4. If session based,
     a. Provide a list for specifying the session from those available
     b. Allow refinement of criteria by selecting classes
5. Build hierarchical recordset for generating the report
6. Bind the data to Crystal Reports
7. Display the reports

D.2.2 – View Reports

Rationale
Requirement D.6

Parent
D.2   - Completed Sessions

Descendants
None

Dependencies
None

User Interface
Server side script generated interface

Algorithm
1. Login Page
       a. Username
       b. Password
2. Specify student by mean of one of the following
       a. Entering the student's surname and selecting from a list of matches
       b. Entering the student's registration number
3. Display all the available reports for the student
       a. Order the reports chronologically
       b. For each report give the following details
              i. Session ID (Hyperlink to get complete report)
             ii. Assessment type
            iii. Year
             iv. Term
              v. Number In Term
             vi. Position In Class
4. For each complete report, display the assessment type specific details.

E – Fees Processing

Rationale
Logical grouping of fees processing components

Parent
0   - infoQuest

Descendants
E1   - Deposits Management
E2   - Fees Management
E3   - Balance Analysis

Dependencies
S.2.1   - System Log In

User Interface

infoQuest : Fees Processing System		
System Fees Deposits Balances		Help
<<Date>>		<< Time>>

Menus

System	[Log Off , Exit]
Fees	[Registered Fees, Fees Templates / Charge , Record Payment / Individual Statement of Account, Batch Produce Statements of Account]
Deposits	[Account Types, Deposit Accounts / Manage Individual Account, Batch Produce Statements of Account]
Balances	[Fee Class Analysis / Fee Balances, Student Deposits Balances]
Help	[Contents, About]

E.1 – Deposit Account Management

Rationale
Abstract logical grouping

Parent
E   - Fees Processing System

Descendants
E.1.1   - Account Types
E.1.2   - Initialise Batch Print
E.1.3   - Deposit Accounts
E.1.4   - Individual Deposit Account

Dependencies
None

User Interface
None

Algorithms
None

E.1.1 – Deposit Account Management

Rationale
Requirement E.7.a.i

Parent
E.1   - Deposit Account Management

Descendants
None

Dependencies
None

User Interface
Full editing support for the Deposit Account Types table

Algorithms
None

E.1.2 – Initialise Batch Print

Rationale
Requirement E.7.e

Parent
E.1   - Deposit Account Management

Descendants
E.2.1   - Do Batch Print

Dependencies
None

User Interface

Batch Print Deposit Statements of Accounts	

Account Type

Start Date

End Date

Balance Type

All Students   ◉
Above Minimum   ○
Below Minimum   ○
At Minimum   ○

Classes

Available   Selected

→
←

Houses

Available   Selected

→
←

Preview   Help   Cancel

E.1.2.1 – Do Batch Print

Rationale
Requirement E.7.e

Parent
E.1.2   - Initialise Batch Print

Descendants
None

Dependencies
None

User Interface
Crystal reports preview pane

E.1.3 – Deposit Accounts

Rationale
Requirement E.7.a.ii

Parent
E.1   - Deposit Accounts Management

Descendants
None

Dependencies
None

User Interface
Interface to support full editing of Deposit accounts table with support for searching for the student to whom the account will belong.

E.1.4 – Individual Account Management

Rationale
Logical grouping for descendants

Parent
E.1   - Deposit Accounts Management

Descendants
None

Dependencies
None

User Interface

User Interface Notes
1. Account loaded using wizard with the following stages
    a. Enter Student's surname or registration number
    b. If using surname, provide list of matching students
    c. Provide list of accounts for selected student
2. Statement frame contains the start and end dates along with a button to generate a Crystal Reports based statement
3. Student's frame contains student's details in a text box.

E.2 – Fees Management

Rationale
Abstract logical grouping

Parent
E   - Fees Processing System

Descendants
E.2.1   - Template Management
E.2.2   - Debit Account
E.2.3   - Registered Fees
E.2.4   - Record Payment
E.2.5   - Statement of Account

Dependencies
None

User Interface
None

Algorithms
None

E.2.1 – Template Management

Rationale
Requirement E.2.b.i

Parent
E.2   - Fees Management

Descendants
E.2.1.1 - Insert Fee Into Template

Dependencies
None

User Interface

Algorithms
None

E.2.1.1 – Insert Fee Into Template

Rationale
Requirement E.2.b.i

Parent
E.2.1   - Template Management

Descendants
None

Dependencies
None

User Interface

Algorithms
None

E.2.2 – Debit Fees Account

Rationale
Requirement E.2

Parent
E.2   - Fees Account Management

Descendants
E.2.2.1   - Load Template
E.2.2.2   - Batch Charge
E.2.2.3   - Individual Charge
E.2.2.4   - Insert Fee

Dependencies
None

User Interface

User Interface Remarks

Input Name	Input Type	Remarks
	Button	Load a template
	Button	Insert individual fee
	Button	Remove the selected fee from the list
R	Button	Reset (Clear) the list of fees

Algorithms
None

E.2.2.1 – Load Fee Template

Rationale
Requirement E.2.b.i
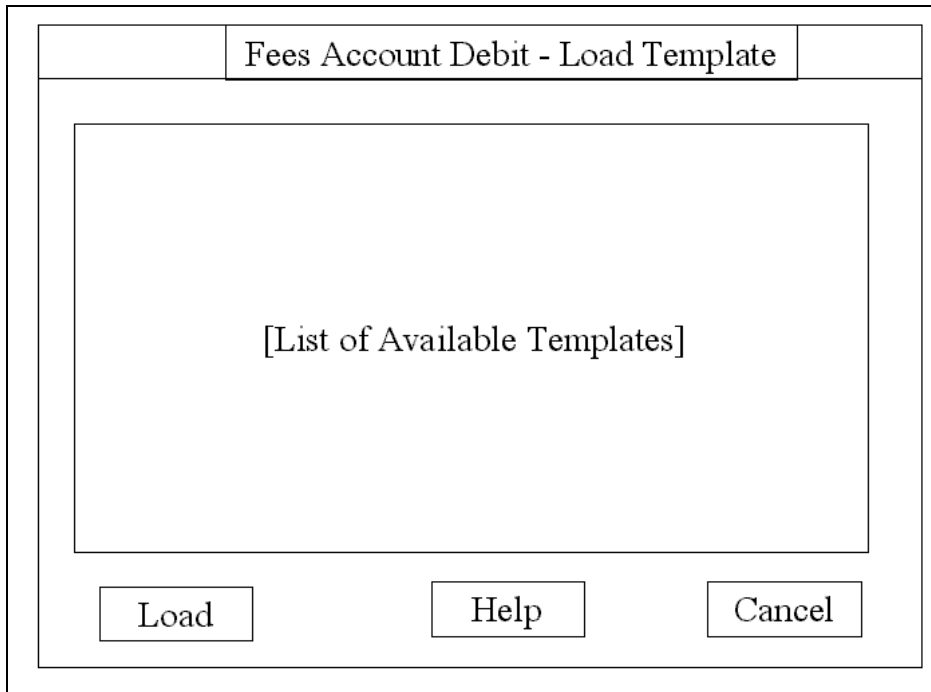
Parent
E.2.2   Debit Fees Account

Descendants
None

User Interface

Fees Account Debit - Load Template

[List of Available Templates]

Load   Help   Cancel

E.2.2.2 Batch Debit Fee Accounts

Rationale
Requirement E.2.c

Parent
E.2.2   - Debit Fee Accounts

Descendants
None

Known Dependencies
None

User Interface
Progress Bar during debiting of accounts

E.2.2.3 Charge Individual Levy Account

Rationale
Requirement E.2.a

Parent
E.2.2   - Debit Fees Accounts

Descendants
None

User Interface
None

E.2.2.4 – Insert Individual Fee

Rationale
Requirement E.2.a

Parent
C.2.2   - Debit Fee Accounts

Descendants
None

Known Dependencies
None

User Interface

Fees Account Debit: Insert Fee	
Fee Name	↓
Amount	
Insert	Help	Cancel

E.2.3 – Registered Fees

Rationale
Required for managing the Registered Fees table specified in the data modelling.

Parent
E.2   - Fees Management
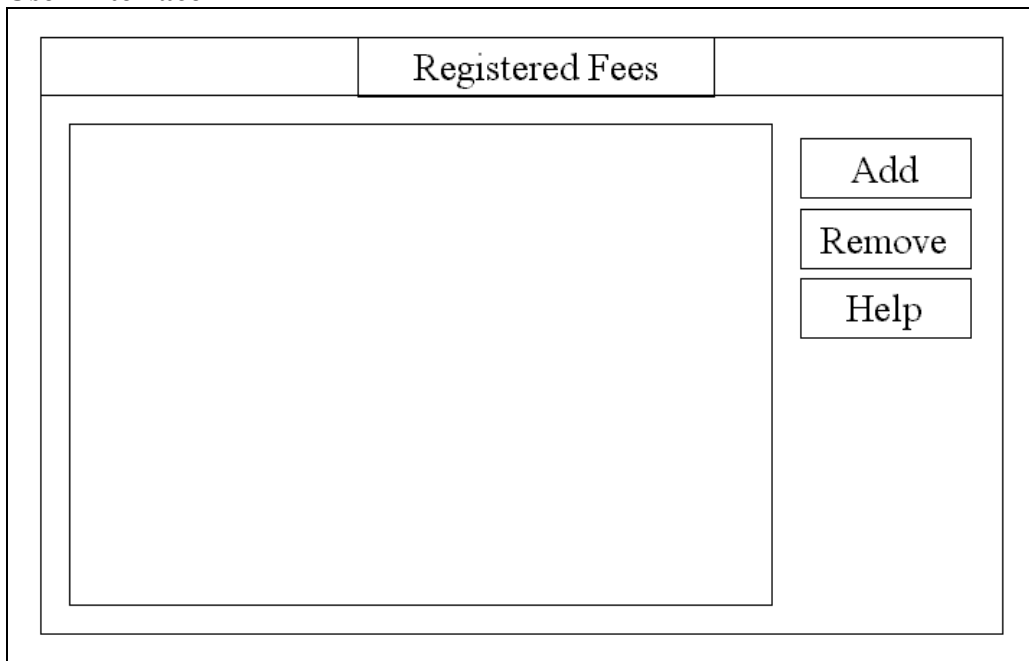
Descendants
None

Dependencies
None

User Interface

E.2.4 - Statement of Account Production

Rationale
Abstract logical grouping

Parent
E.2   - Fees Management

Descendants
E.2.4.1   - Individual
E.2.4.2   - Start Batch

Known Dependencies
None

User Interface
None

E.2.4.1 - Individual Statement of Account

Rationale
Requirement E.4.a

Parent
E.2.4   Statement of Account Production
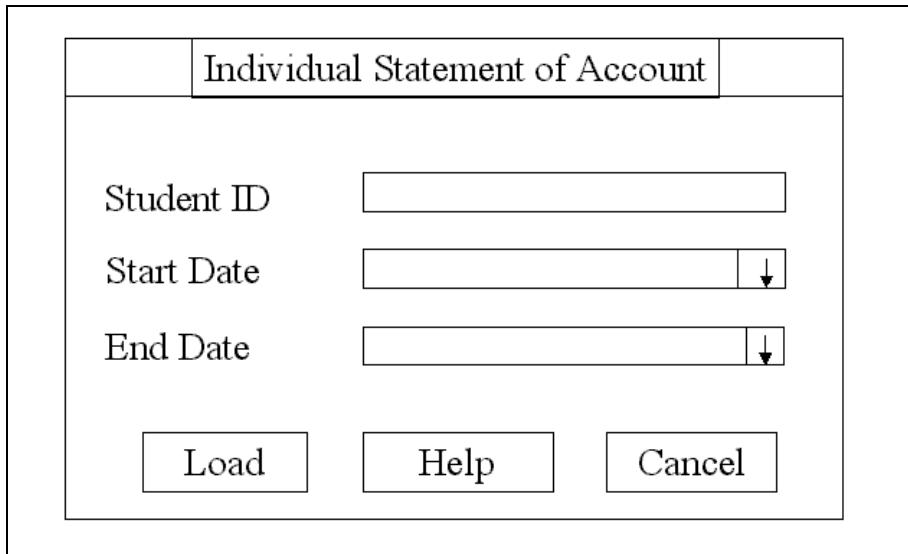
Descendants
None

Known Dependencies
None

User Interface

Notes On Interface
1. Loads a crystal reports based statement of account.
2. Includes option for searching for student ID based on surname

E.2.4.2 - Batch Statement of Account

Rationale
Requirement E.4.b

Parent
E.2.4   Statement of Account Production

Descendants
None

Known Dependencies
None

User Interface

E.2.5 - Record Payment

Rationale
Requirement E.3

Parent
E.2   - Fees Management

Descendants
None

Known Dependencies
None

User Interface

E.3 - Balance Analysis

Rationale
Abstract logical grouping

Parent
E   - Fees Management

Descendants
E.3.1   - Fee Class
E.3.2   - Student Balances

User Interface
None

E.3.1 - Fee Class

Rationale
Requirement E.5

Parent
E.3   - Balance Analysis

Descendants
None

User Interface

infoQuest Fee Class Balance Analysis			
Analyse	Print	Help	Close

Start Date

End Date

Title	
Start Date	
End date	
Created	

Details	Amount

Output Specification
1. Header
   a. Title i.e. Fee Class Balance Analysis
   b. Start Date
   c. End Date
   d. Created i.e. Time at which analysis started
2. Details Section
   a. Total payments over period
   b. Current Accruals
   c. Current Prepayments
   d. Amount charged under each fee

E.3.2 - Student Balances

Rationale
Abstract logical grouping

Parent
E.3　　　　　- Balance Analysis

Descendants
E.3.2.1　　　- Levy Balances
E.3.2.2　　　- Deposit Balances

User Interface
None

E.3.2.1 - Fees Account Balance Analysis

Rationale
Requirement E.6

Parent
E.3.2   - Student Balances

Descendants
None

Known Dependencies
None

User Interface

Output Specification
1. Student ID
2. Surname
3. Forenames
4. Class
5. House
6. Debits
7. Credits
8. Balance

E.3.2.2 – Analyse Deposit Account Balances

Rationale
Requirement E.6

Parent
E.3.2   - Student Balances

Descendants
None

Known Dependencies
None

User Interface

Output Specification
9.   Student ID
10. Surname
11. Forenames
12. Class
13. House
14. Debits
15. Credits
16. Balance

F – Event Scheduling

Rationale
Provides access to descendants.

Parent
0   - infoQuest

Descendants
F.1   - Club Types
F.2   - Clubs
F.3   - General Events
F.4   - View Calendar
F.5   - Web Interface

Dependencies
None

User Interface

	infoQuest: Event Scheduling	
Club Types		
Clubs		
Meeting Times		
Fixtures		
General Events		
View Calendar	Help | Log Off | Exit	

F1 – Club Types

Rationale
Requirement F.1

Parent
F   - Event Scheduling

Descendants
None

Dependencies
None

User Interface

F2 – Clubs

Rationale
Requirement F.2

Parent
F   - Event Scheduling

Descendants
None

Dependencies
None

User Interface

Notes On Interface
1. Wizard for adding a Supervisor
    a. Enter Surname or Staff ID
    b. Display list of matching members of staff and allow user to select
    c. Enter the supervisor's role.
2. It should not be possible to remove a supervisor who is still registered as having duties in the club.

3. Meeting Times Interface is a grid with the following columns
   a. Description
   b. Start Date
   c. End Date
   d. Start Time
   e. End Time
   f. Supervisor
   g. Primary recurrence rule i.e. [None | Daily | Monthly]
   h. Secondary recurrence rule e.g. Every Tuesday for weekly recurrence rule
4. Wizard for adding meeting times
   a. Enter the following details
      i. Description
      ii. Start Date
      iii. End Date
      iv. Start Time
      v. End Time
      vi. Supervisor
   b. Specify if event recurs
   c. If event recurs then select primary recurrence rule from the following
      i. Daily
      ii. Monthly
   d. If the event recurs then specify secondary recurrence rule
      i. If daily recurrence rule then specify the number of days after which it recurs.
      ii. If monthly recurrence then specify one of the following secondary recurrence rules
         ❖ Day [Integer] of every [Integer] months
         ❖ [Integer] [Saturday] of every [Integer] months
   e. Display "Finished" screen
5. Fixtures added using a window for specifying the details which is also used for editing a fixture

F.3 – General Events

Rationale
Requirement F.4

Parent
F   - Event Scheduling

Descendants
None

Dependencies
None

User Interface

```
┌──────────────────────────────────────────────────────────┐
│        ┌──────────────────────────────────┐              │
│        │   infoQuest: General Events       │              │
│  ┌─────┬──────┬────────┬──────┬───────┐    │              │
│  │ New │ Edit │ Delete │ Help │ Close │              │
│  └─────┴──────┴────────┴──────┴───────┘                  │
│                                                          │
│      Description  ┌────────────────────────────────┐     │
│      Start Date   ┌────────────────────────────────┐     │
│      End Date     ┌────────────────────────────────┐     │
│      Start Time   ┌────────────────────────────────┐     │
│      End Time     ┌────────────────────────────────┐     │
│                                                          │
│      ┌──────────────┐  ┌──────────────┐  ┌──────────┐    │
│      │ << Previous  │  │   Next >>     │  │  Close   │    │
│      └──────────────┘  └──────────────┘  └──────────┘    │
└──────────────────────────────────────────────────────────┘
```

F.4 – View Calendar

Rationale
Integrated view

Parent
F   - Event Scheduling

Descendants
None

Dependencies
None

User Interface

From	To	Venue	Description

infoQuest: Calendar

Date

{Month View Control}

F.5 – Web Interface

Rationale
Requirement F.

Parent
F   - Event Scheduling

Descendants
None

Dependencies
None

User Interface
Server generated html search form
1. Type of event(s) to search for
    a. Examinations
    b. Club meetings
    c. Club Fixtures
    d. General events
2. List for selecting clubs on which to base search on
3. Academic level(s) of examinations to return
4. Start date
5. End date

Output Specification
1. Colour code events as follows
    a. Examinations - Green
    b. Club meetings - Blue
    c. Club Fixtures - Yellow
    d. General events- Brown
2. Show the following fields
    a. Date
    b. Start Time
    c. End Time
    d. Type [Exam | Meeting | Fixture | General]
    e. Description
3. Provide link back to original form

G – Library Management

Rationale
Provides a logical grouping for descendants

Parent
0   - infoQuest

Descendants
G.1   - Patrons
G.2   - Circulation
G.3   - Catalogues
G.4   - Fines
G.5   - Reports

Dependencies
S.2.1   - System logon

User Interface

infoQuest: Library Management	
Patrons	
Borrow	
Return	
Catalogues	
Fine Payments	
Fine Transfers	
Reports	Help Log Off Exit

G.1 - Patrons

Rationale
Requirements G.1, G.2, G.3 and G.4

Parent
G   - Library Management

Descendants
None

Dependencies
None

User Interface

infoQuest: Library Patrons

| New | Edit | Suspend | Locate | Sort | Subset | Report | Help | Close |

| Registered Students | Deregistered Students | Members of Staff | External Patrons | Membership Categories | Suspensions |

168

User Interface Notes
1. Registered students tab displays the following details
   a. Student ID
   b. Full Name
   c. Class
   d. House
   e. Membership Category
   f. Status [Suspended | Active]
   g. Suspension details if applicable
2. Deregistered students tab displays the following details
   a. Student ID
   b. Full name
   c. Membership Category
   d. Status [Suspended | Active]
   e. Suspension details if applicable
3. Members of Staff tab displays the following details
   a. Staff ID
   b. Full Name
   c. Department
   d. Membership Category
   e. Status [Suspended | Active]
   f. Suspension details if applicable
4. External Patrons tab
   a. Patron ID
   b. Title
   c. First Name
   d. Middle Name
   e. Surname
   f. Address
   g. Phone
   h. Expiry Date
   i. Membership Category
   j. Status [Suspended | Active]
   k. Suspension details if applicable
5. Membership Category tab
   a. Category Name
   b. Membership rule
   c. Borrowing limit rule
6. Suspensions tab contains a list with the following details
   a. Patron, student or staff ID
   b. Type [Registered student | Deregistered Student | Members of Staff | External Patron]
   c. Full Name
   d. Suspension details
7. Subset view as to membership category

8. Wizard for specifying membership rule
    a. Specifies the type of patron to which the applies i.e. [Registered student | Deregistered Student | Members of Staff | External Patron]
    b. If registered student then select the house and class to which the student should belong
    c. If deregistered student then specify the reason for deregistration that should form the basis for membership.
    d. If member of staff then specify the department to which the member of staff should belong.
    e. Run a check to ensure that the category does not conflict with any other category.
    f. Display a "Finished" dialogue
9. Wizard for borrowing limit rule
    a. Specify the overall total number of books that maybe borrowed
    b. Specify the maximum that may be borrowed form each section

G.2 - Circulation

Rationale
Abstract grouping object

Parent
G   - Library Management

Descendants
G.2.1   - Borrow
G.2.2   - Return

Dependencies
None

User Interface
None

G.2.1 - Borrow

Rationale
Requirement G.8.a

Parent
G.2   - Circulation
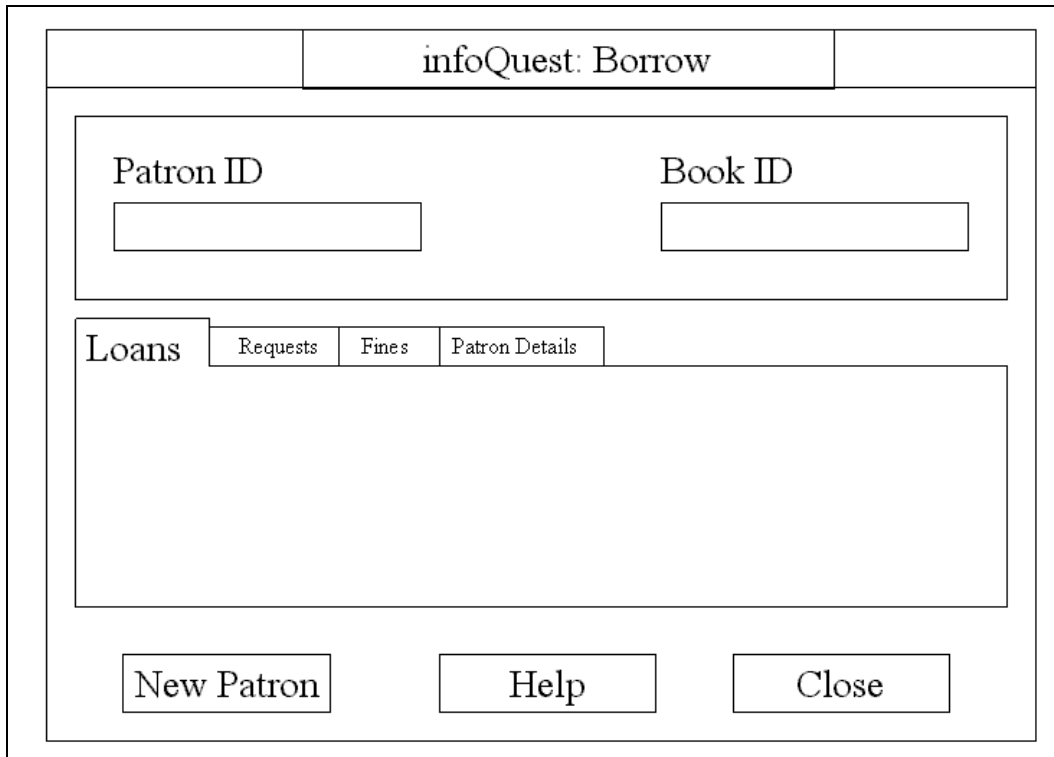
Descendants
None

Dependencies
None

User Interface

Algorithms

1. On entering patron ID,
   a. Ensure that subscription ID is valid
   b. Check patron's suspension status
   c. If external patron, check if subscription has expired
   d. Populate the tabs

2. On entering book ID
   a. Repeat patron ID checks
   b. Ensure that the Book ID is valid
   c. Ensure that the book belongs to the location from which it is being borrowed.
   d. Ensure that the book is not recorded as being on loan
   e. Ensure that the book has not been requested by someone else. This only applies to request that have become effective and have not yet expired.
   f. If a requisition, indicate that a request is to be processed.
   g. Check the subscriber's limit adjusting appropriately if a request is to be processed.
   h. Calculate due date
   i. Enter loan record and if applicable, delete requisition record.
   j. Clear text box for book id
   k. Update loans tab
   l. If a request was processed during the loan, update the requisitions tab.

G.2.2 - Return

Rationale
Requirement G.8.b and G.8.c

Parent
G.2   - Circulation

Descendants
None

Dependencies
None

User Interface

infoQuest: Return		

Book ID

Patron Name	Book Title	Fine

Help   Close

Algorithms
1. Verify that the book is on loan
2. Verify that the book belongs to the location to which it is being returned
3. If book is overdue
      a. Calculate the fine due
      b. Create a fine record
4. Delete the loan entry
5. Update   grid

G.3 - Catalogues

Rationale
Requirements G.5 and G.6

Parent
G   - Library Management

Descendants
None

Dependencies
None

User Interface

G.4 - Fines

Rationale
Requirement G.7

Parent
G   - Library Management

Descendants
G.4.1   - Fine Payments
G.4.2   - Transfers to Bursar

Dependencies
None

User Interface

infoQuest: Catalogues						
Pay Fine	Transfer To Bursar	Locate	Sort	Subset	Help	Close

Pending Fines | Pending Transfers | Transfers to Bursar

G.4.1 – Fine Payments

Rationale
Requirement G.7.b

Parent
G.4   - Fines

Descendants
None

Dependencies
None

User Interface

Algorithm
1. Display message box with the following details
    a. Receipt Number
    b. Date
    c. Patron ID
    d. Patron name
    e. Book ID
    f. Title of Book
    g. Fine due on book
2. Confirm fine payment
3. Print receipt if required
4. Close window

G.4.2 – Outbound Transfers

Rationale
Requirement G.7.c

Parent
G.4   - Fines

Descendants
None

Dependencies
None

User Interface

G.5 – Reports

Rationale
Requirements G.10 and G.11

Parent
G   - Library Management

Descendants
None

Dependencies
None

User Interface

Prepare Reports
Select Report [   ] ↓
Generate   Help   Cancel

G.6 - OPAC

Rationale
Provides access to descendants

Parent
G　　- Library Management

Descendants
G.6.1　- Search Catalogues
G.6.1　- View Own Record

Dependencies
None

User Interface
HTML page with links to descendants

G.6.1 – Search Catalogues

Rationale
Requirements G.12.a and G.12.b

Parent
G.6 - OPAC

Descendants
G.6.1.1- Requisitions

Dependencies
None

User Interface
HTML form with the following fields
`K -`   Accession number of the book
`L -`   Author
`M -`   Title
`N -`   ISSN for periodicals and ISBN for books

Output Specification
1. A table with the following fields
   a. Book ID (Hyper linked)
   b. Author(s)
   c. Title
   d. Availability i.e. Available or Date due
2. Book View
   a. Book ID
   b. Classification Code
   c. Title
   d. Author(s)
   e. Edition
   f. Publication Year
   g. ISSN or ISBN
   h. Section
   i. Location
   j. Link to enable patron to request the book
   k. Link back to original page

G.6.1.1 – Requisitions

Rationale
Requirement G.12.d

Parent
G.6.1   - Search Catalogues

Descendants
None

Dependencies
None

User Interface
HTML page with the following
`O` – Accession number of the book
`P` – Author
`Q` – Title
`R` – ISSN for periodicals and ISBN for books
   `S` – Requisitions on the book to date with the following details
      1. Effective from
      2. Expires
   `T` – HTML form with the following fields
      1. Patron ID
      2. Online password
      3. Date on which requisition should become effective
      4. Date on which requisition will expire

Remarks
1. Requisition period must not overlap with any current requisition and must not be for more than one week.
2. Patron must not already have a pending request on the book
3. Requests count as loans and borrowing limits should thus be enforced.

G.6.2 – View Own Record

Rationale
Requirement G.12.c

Parent
G.6   - OPAC

Descendants
None

Dependencies
None

User Interface
HTML form with the following fields
ʊ –   Patron ID
v –   Online Password

Output Specification
1. Patron Details
      a. Patron ID
      b. Full Name
      c. Membership Category
2. Loans
      a. Book ID
      b. Title
      c. Date Due
3. Requests
      a. Book ID
      b. Title
      c. Date on which it becomes effective
      d. Expiry Date
      e. Link to enable patron to delete the request
4. Fines owing
      a. Book ID
      b. Title
      c. Amount Owing

H - Mail Merge

Rationale
Provides user interface for accessing the mail merge components.

Parent
0   - infoQuest

Descendants
H.1   - File Management
H.2   - Text formatting and editing
H.3   - Mail Merge processing

User Interface

infoQuest Mail Merge: <<File Name>>

System   File   Edit   Format   Insert   Mail Merge   Help

<<Tool Bar>>

{ RTF Text Box}

Menus

System	[Log Off , Exit]
File	[New, Open / Save, Save As / Print]
Edit	[Cut, Copy, Paste / New Search, Find Next, Search and Replace / Spell Check]
Format	[Font, To Upper Case, To Lower Case]
Insert	[Date, File, Picture]
Mail Merge	[New Session / Preview Personalized Letters, Produce Personalized Letters, Insert Merge field / View Source SQL]
Help	[Contents, About]

H.1 - File Management

Rationale
Abstract logical grouping of file management components

Parent
C2   - Mail merge

Descendants
H.1.1   - Create New File
H.1.2   - Open file
H.1.3   - Save existing file
H.1.4   - Print Current File

User Interface
None

H.1.1 Create new Document

Rationale
Requirement H.3.a

Parent
H.1   - File Management

Descendants
None

Known Dependencies
None

User Interface
None

Algorithm
1. If the currently loads file has changed since being last saved, give option to save.
2. Set default font settings to
    a. Times New Roman 12pt
    b. Left justified
    c. Not bold
    d. Not italic
    e. Not underlined
    f. Not struck through
    g. Not in bullet style
    h. Black text fore colour
    i. No indent
3. Blank out the document area

H.1.2 Open Document

Rationale
Requirement H.3.b

Parent
H.1   - File Management

Descendants
None

User Interface
System open dialogue

Algorithm
1. If the currently loaded file has changed since being last saved, give option to save.
2. Blank out the document area
3. Load the specified document using the **LoadFile** method of the RTF text box.

H.1.3 Save Document

Rationale
Requirement H.3.b

Parent
H.1   - File Management

Descendants
None

User Interface
System save dialogue

Remarks
Use RTF text box's **SaveFile** method. If the loaded file already has a file name associated with it, then proceed to use this file name without prompting unless the user has explicitly requested that this not be done.

H.1.4 – Print File

Rationale
Requirement H.3.b

Parent
H.1   - File Management

Descendants
None

User Interface
System Print dialogue

Remarks
Use Pierre-Emmanuel Gross' routine for printing RTF files. Do not use the default **PrintRTF** method because it does not support margins.

Code source at http://www.codeguru.com/vb/openfaq/comments/159.shtml

H.2 – Text Formatting and Editing

Rationale
Abstract logical grouping of file management components

Parent
H   - Mail merge

Descendants
H.2.1 - Font and Style Management
H.2.2 - Search
H.2.3 - Insertions
H.2.4 - Search and Replace
H.2.5 - Undo System
H.2.6 - Spell Check

User Interface
None

H.2.1 – Font and Style Management

Rationale
Requirement H.3.g

Parent
H.2   - Text Formatting and Editing

Descendants
None

User Interface
System Font dialogue

Remarks
The Font dialogue should be loaded with the current style before being displayed. Appropriate treatment of nulls should be provided.

H.2.2 – Search

Rationale
Requirement H.3.d

Parent
H.2   - Text Formatting and Editing

Descendants
None

User Interface

Remarks
Should carry a reset option to enable new searches. On finding a match, the focus should be returned to the document with the match highlighted.

H.2.3 – Insertions

Rationale
Abstract grouping of components used in insertions

Parent
H.2   - Text Formatting and Editing

Descendants
H.2.3.1- Date Insertion
H.2.3.2- Picture Insertion
H.2.3.3- File Insertion

User Interface
None

H.2.3.1 – Insert Date

Rationale
Requirement H.6.a

Parent
H.2.3   - Insertions

Descendants
None

User Interface
Month view control for the current year. Input should be by double clicking a date.

H.2.3.2 – Insert Picture

Rationale
Requirement H.6.b

Parent
H.2.3   - Insertions

Descendants
None

User Interface
System's open dialogue with a filter for the following formats
- Bitmap (Default)
- JPEG Graphic Interchange Format
- Graphic Interchange Format
- All picture files

H.2.3.3 – Insert File

Rationale
Requirement H.6.c

Parent
H.2.3   - Insertions

Descendants
None

User Interface
System's open dialogue with a filter for the following formats
- Rich Text Format (Default)
- Standard Text

Remarks
Use a working area form with an RTF control to load the file and then copy the contents of this control and paste them into the current document at the current insertion position.

H.2.4 – Search and Replace

Rationale
Requirement H.3.d

Parent
H.2   - Text Formatting and Editing

Descendants
None

User Interface

H.2.5 – Undo System

Rationale
Requirement H.3.e

Parent
H.2   - Text Formatting and Editing

Descendants
None

Dependencies
None

Programming Interface
1.   Public Error variables
     a.   errUndoStackEmpty
          ❖ Attempted to obtain an undo snapshot when none exist
     b.   errRedoStackEmpty
          ❖ Attempted to obtain an redo snapshot when none exist
     c.   errStackAlreadyInitialised
          ❖ Attempted to initialise an already initialised undo object
     d.   ErrInvalidParameter
          ❖ Parameter passed is invalid
     e.   errStacksNotIntialised
          ❖ Attempted to access undo system before initialising it
2.   InitialiseUndoSystem (intMaxSize As Integer)
     a.   Initialises the Undo system
     b.   May only be called once.
     c.   All calls other than the first throw an errStackAlreadyInitialised exception
     d.   IntMaxSize must be a positive integer
3.   ReInitialiseUndoSystem()
     a.   Reinitialises the undo system e.g. when a new file is opened
     b.   Undo system must have been initialised otherwise an errStacksNotIntialised
          exception will be thrown.

4.   AddUndoSnapshot(strSnapshot As String, Optional blnDontFlush As Boolean)
     a.   Adds an image of the current text to facilitate undoing later
     b.   strSnapshot is the text to be stored
     c.   blnDontFlush stops the Redo system from being reinitialised because a Redo
          caused the change.

5. AddRedoSnapshot(strSnapshot As String)
    a. After an Undo, a snapshot is added to ensure that the Undo can be undone.

6. getUndoSnapShotsAvailable() As Integer
    a. Returns the number of Undo calls that can be made at the current moment
    b. Helpful for updating the interface by enabling or disabling the Undo buttons

7. getRedoSnapShotsAvailable() As Integer
    a. Returns the number of Redo calls that can be made at the current moment
    b. Helpful for updating the interface by enabling or disabling the Redo buttons

8. GetUndoSnapshot(strRedoSnapshot As String) As String
    a. Returns an RTF formatted string that can be used to carry out an Undo
    b. The undo system must have been initialised and not empty otherwise an errUndoStackEmpty exception is thrown.
    c. Adds a redo snapshot

9. GetRedoSnapshot(strUndoSnapshot As String) As String
    a. Returns an RTF formatted string that can be used to carry out an Redo
    b. The undo system must have been initialised and not empty otherwise an errUndoStackEmpty exception is thrown.

H.2.6 – Spell Check

Rationale
Requirement H.7

Parent
H.2   - Text Formatting and Editing

Descendants
None

Dependencies
None

Programming Interface
1. Error variables
    a. errCorrectWord
        ❖ Attempt made to get a list of suggestions on a word that is in the dictionary.
2. InitiliseSpellChecking ()
    a. Initialises the spell checking system
3. Valid_word (strWord as String) as Boolean
    a. Return true if the word is in the dictionary
4. getSuggestions (strWord as String) as Array
    a. Returns an array of spelling suggestions.
    b. Array is based at 1
    c. Throws an errCorrectWord if the word is in the dictionary

H.3 – Mail Merge Processing

Rationale

Abstract logical grouping of Mail merge processing routines.

Parent

H   - Mail Merge

Descendants

H.3.1   - Load Query Source File
H.3.2   - View Query Source
H.3.3   - Insert Mail Merge Field
H.3.4   - Preview Personalised Letters
H.3.5   - Produce Personalised Letters

User Interface

None

H.3.1 – Load Query Source File

Rationale
Requirement H.1

Parent
H.3   - Mail Merge Processing

Descendants
None

User Interface
None

Algorithms
1. Use the System Open File dialogue box to get the file name of the source SQL file
2. Assign query file's contents to a command object. If assignment fails because of a syntax error, give the user an option to try loading a different file.
3. Check that the specified query does not modify records. If it does, give the option to try another query file.
4. Execute the stored query and assign the results to a global recordset object.

H.3.2 – View Query Source

Rationale
Requirement H.1

Parent
H.3   - Mail Merge Processing

Descendants
None

User Interface
A form containing a text box with both scroll bars.

Algorithms
Retrieve the text in the SQL property of the command object being used by the Mail Merge routines and assign it to the text box on the form.

H.3.3 – Insert Merge Field

Rationale
Requirement H.2

Parent
H.3   - Mail Merge Processing

Descendants
None

User Interface

```
┌─────────────────────────────────────────────┐
│        │ Insert Mail Merge Field │           │
│        └─────────────────────────┘           │
│  ┌─────────────────────────────────────┐     │
│  │                                     │     │
│  │                                     │     │
│  │                                     │     │
│  │          {List of Fields}           │     │
│  │                                     │     │
│  │                                     │     │
│  │                                     │     │
│  └─────────────────────────────────────┘     │
│                                               │
│   ┌────────┐   ┌────────┐   ┌─────────┐       │
│   │   OK   │   │  Help  │   │ Cancel  │       │
│   └────────┘   └────────┘   └─────────┘       │
└─────────────────────────────────────────────┘
```
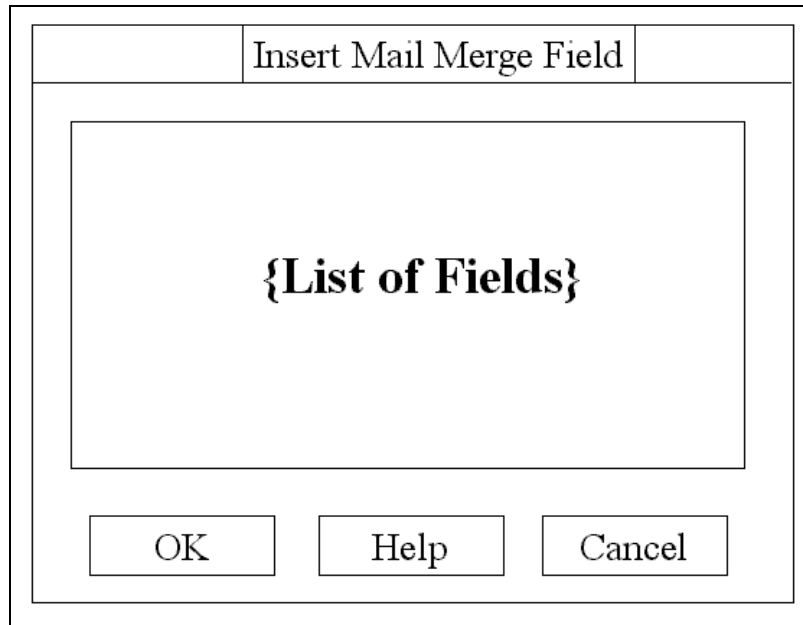
Algorithms
1. Component Load
    a. Retrieve all the fields in the query result and place them in the List of Fields
2. OK Chosen
    a. Paste the caption of the selected list item into the document surrounded by start and end of field characters.

H.3.4 – Preview Personalised Letters

Rationale
Requirement H.5

Parent
H.3   - Mail Merge Processing

Descendants
None

User Interface
Form containing an RTF component with both scroll bars.

Algorithm
Check if every field specified exists in the query result
If an invalid field is encountered
      Inform user and abort operation
Else
      For each record in the query result
            Make a copy of the input document in a working area
            For each field in the source query
                  Replace field name in working area copy with personalised details
            Next
            Send result to Output form
      Next
      Display output form
End

H.3.5 – Produce Personalised Letters

Rationale
Requirement H

Parent
H.3   - Mail Merge Processing

Descendants
None

User Interface
System print dialogue

Algorithm
Check if every field specified exists in the query result
If an invalid field is encountered
   Inform user and abort operation
Else
   For each record in the query result
      Make a copy of the input document in a working area
      For each field in the source query
         Replace field name in working area copy with personalised details
      Next
      Send result to printer starting on a new page
   Next
End

I – SQL Execution

Rationale
Requirement I.1

Parent
0   - infoQuest

Descendants
I.1   - SQL Execution Results

User Interface

infoQuest: SQL Execution

Tools

{Textbox for Entering SQL}

Menu

Tools   [New Query, Open SQL File, Save SQL File, Save SQL File As /
         Execute Query / Help / Exit]

Input Name	Input Type	Remarks
SQL text box	Text box	• Should have both scroll bars
New Query	Menu	• Clear the text box
Open SQL file	Menu	• Use the system Open File dialogue to open a text file
Save SQL File	Menu	• If the loaded file already has a filename associated with it then save the loaded text under that file • Otherwise use the system Save As dialogue to save the file
Save SQL File As	Menu	• Use the system Save As dialogue to save the file
Execute Query	Menu	• Assign the text to a query object's SQL property. if the query contains any errors, an exception will be thrown. • Ensure that the query does not have the ability to modify data before execution • Load component C5.1 to display the results
Print	Menu	• Use the system Print dialogue to print the SQL text.
Help	Menu	• Load context sensitive help
Exit	Menu	• Unload the component.

I.1 – SQL Execution Results

Rationale
Requirement I
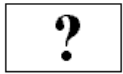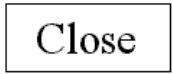
Parent
I   - SQL Execution

Descendants
None

User Interface

SQL Execution Results

Tools

{Grid of Results}

← → ? Close **Page X of Y**

Menu

Tools   [Adjust row height, Export to Excel, Export to Generic CDF / Help / Close]

Algorithms

1. Display the Results
   a. Execute the query and assign the results to a recordset
   b. If the recordset is empty then abort
   c. Calculate the number of pages returned by the result
   d. Enter the column headers by iterating through the field object of the recordset
   e. Display the first page
2. Adjust Row Height
   a. Use input box control to get the preferred number of lines of text to display
   b. Convert text height to scale mode units
   c. For each row in the grid excluding the header row, set the new height to the calculated height
3. Export To Excel
   a. Get the filename of the target export
   b. If the file exists, delete it after confirming with the user
   c. Create an Excel Workbook with a Single Worksheet
   d. Export the header row
   e. Load each page from page 1 and export its contents excluding the Header page
   f. Save and release the workbook
   g. Redisplay the page that was being viewed just before the exporting began.
4. Export to Generic CDT
   a. Get the filename of the target export
   b. If the file exists, delete it after confirming with the user
   c. Create a text file to export the data to
   d. Export the header row
   e. Load each page from page 1 and export its contents excluding the Header page
   f. Save the text file

J – Nominal Roll

Rationale
Requirement B.3.e

Parent
0   - infoQuest

Descendants
J.1   - Student record
J.2   - Subjects

Dependencies
None

User Interface
Two HTML pages
1. Log in page
   a. Staff ID
   b. Online Password
2. Links to the descendants

J.1 - Student Records

Rationale
Requirement B.3.e

Parent
J   - Nominal Roll

Descendants
None

Dependencies
None

User Interface
Server generated HTML pages
1. HTML form with the following fields
    a. Student ID
    b. Student Surname
    c. Guardian's Surname
    d. Guardian's ID
    e. Mother's surname
    f. Mother's ID
    g. Father's surname
    h. Father's ID
    i. Doctor's Name
    j. Medical Aid Type
    k. Class
    l. House
2. Results summary with the following details
    a. Student ID (Hyper linked)
    b. Student's Full Name
    c. Class
    d. House

3. Detailed results (Only applicable sections will be shown depending on deregistration status
   a. Student Details
      ❖ Student ID
      ❖ Surname
      ❖ First Name
      ❖ Middle name
      ❖ Sex
      ❖ Date of Birth
      ❖ Date Enrolled
      ❖ Status [Deregistered / Enrolled]
      ❖ Date Record was last Updated
      ❖ Birth Certificate Number
      ❖ Parent's Marital Status
      ❖ Medical Aid Number
      ❖ Religion
      ❖ Primary Language at Home
      ❖ Remark,
   b. Guardian's Details
      ❖ Parent ID
      ❖ Title
      ❖ Surname
      ❖ First Name
      ❖ Home Address
      ❖ Home Phone Number
      ❖ Employer
      ❖ Business Address
      ❖ Occupation
      ❖ Business Phone Number
      ❖ Business Fax Number
      ❖ Mobile Phone Number
      ❖ Email Address
      ❖ Marital status
   c. Father
      ❖ Same as those for the Guardian
   d. Mother
      ❖ Same as those for the Guardian
   e. Class
      ❖ Class
      ❖ Academic year
      ❖ Form Teacher ID
      ❖ Form Teacher's Name

f. Doctor
- ❖ Doctor ID
- ❖ Name
- ❖ Phone Number
- ❖ Address
- ❖ Email

g. House
- ❖ House Name
- ❖ Name
- ❖ Boarding Status
- ❖ House Master's Staff ID and Name

h. Medical Aid Type
- ❖ Name
- ❖ Type
- ❖ Remark

i. Subjects Taken
- ❖ Subject Code
- ❖ Title
- ❖ Teacher's name and ID
- ❖ Department

j. Deregistration Details
- ❖ Date Deregistered
- ❖ Reason
- ❖ Contact Address
- ❖ Contact Phone Number
- ❖ Contact email address

J.2 - Subjects

Rationale
Requirement B.3.e

Parent
J   - Nominal Roll

Descendants
None

Dependencies
None

User Interface
Server generated HTML forms
1. Search form
    a. Subject Code
    b. Subject Title
    c. Teacher's ID
    d. Teacher's surname
    e. Department
    f. Academic Year
2. List of Subjects matching criteria
    a. Subject Code (Hyper Linked)
    b. Subject Title
    c. Teacher's Name
    d. Department
    e. Academic Year
3. Subject details
    a. Subject Code
    b. Subject Title
    c. Teacher's ID
    d. Teacher's surname
    e. Department
    f. Academic Year
    g. List of Students
        i.   Student ID (Hyper Linked)
        ii. Name
        iii. Class

S – System Services

Rationale
Abstract grouping object for components that are shared by the entire system

Parent
0   - infoQuest

Descendants
S.1   - Setup
S.2   - System Start Up
S.3   - Export
S.4   - Table Creation

Dependencies
None

User Interface
None

S.1 – Setup

Rationale
System deployment

Parent
S   - System services

Descendants
None

Dependencies
None

User Interface
Microsoft Visual Basic Package and Deployment system generated installer

S.2 – System Start Up

Rationale
Provides a generic system start up sequence for all the components

Parent
S   - System Services

Descendants
S.2.1   - System Logon

User Interface
Splash screen

Algorithm
If another copy of the loading subsystem is already running
        Inform user and abort subsystem start up.
Else
        Get connection from registry
        Load Logon system subcomponent to obtain username and password
        Attempt to connect to the database
        If connection fails, inform user and reload logon screen
End if

S.2.1 – System Logon

Rationale

Used to supply the username and password used to connect to the database
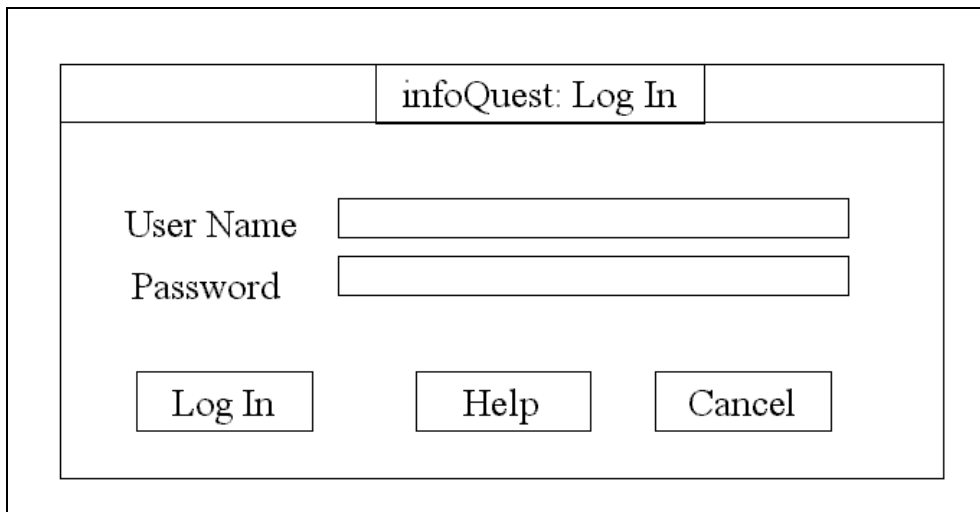
Parent

S.2   - System Startup

Descendants

None

User Interface

S.3 – Export

Rationale
Required to facilitate the attainment of requirement J

Parent
S   - System Services

Descendants
None

User Interface
None

Programming Interface
The components should provide two methods, one for exporting to Excel 95 and the other to Generic Comma Delaminated Text format. The methods should accept as a compulsory argument, a grid containing the body of the tabulated data to be exported. There should also be a second optional argument specifying a grid with tabulated text to be included as a header to the report. During export, a progress display should be provided.

S4 – Admin Console

Rationale
Provides access to descendants

Parent
S   - System Services

Descendants
S.4.1   - Table Creation
S.4.2   - Online Users

Dependencies
None
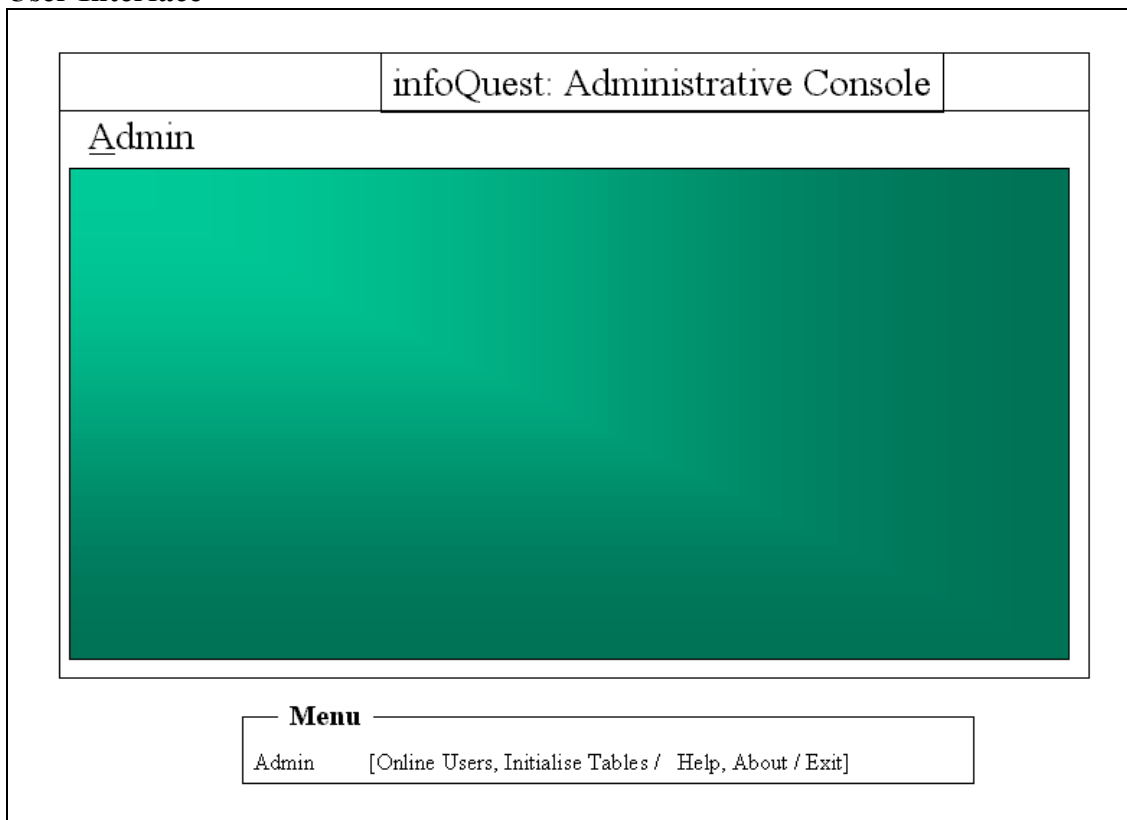
User Interface

S.4.1 – Table Creation

Rationale
Ease creation of the system tables

Parent
S.4   - Admin Console

Descendants
None

Dependencies
None

User Interface

Initialising infoQuest Database Tables
Cancel
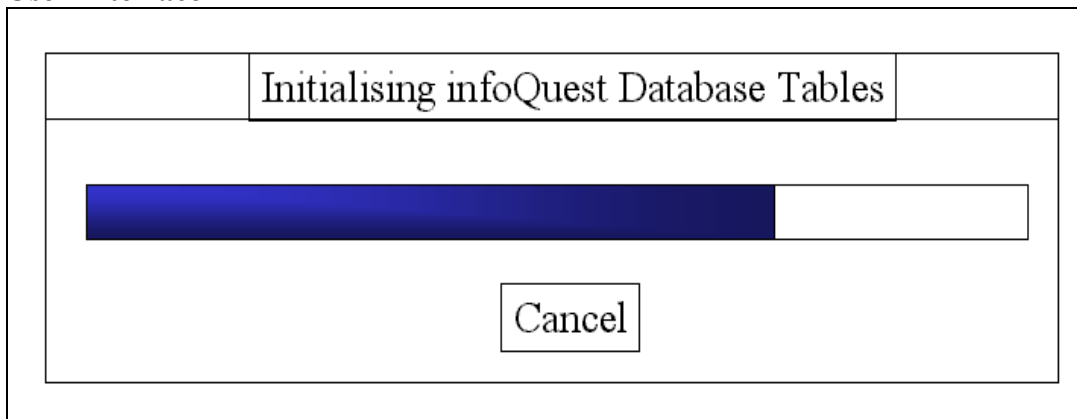
Remarks
1. Confirm creation of tables
2. Target database must be empty. Do not replace existing tables.

S.4.2 – Online Users

Rationale
Facilitate secured login into the online system

Parent
S.4   - Admin Console

Descendants
S.4.2.1 - Change Password
S.4.2.2 - Change Rights

Dependencies
None

User Interface

infoQuest: Online Users' Accounts		
Change Password	Change Rights	Help	Close

User Name	Full Name	User Type

S.4.2.1 – Change Password

Rationale
Allow users to edit their online passwords. Not done over the Internet because of the risk of eavesdropping and replay attacks.

Parent
S.4.2   - Online Users

Descendants
None

Dependencies
None

User Interface

Change Password	
User Name	
New Password	
Re-enter Password	
Save	Help	Cancel

S.4.2.2 – Change Rights

Rationale
Control access to nominal roll and assessment records

Parent
S.4.2   - Online Users

Descendants
None

Dependencies
None

User Interface

EVALUATION

Progress on Implementation

The design draft for infoQuest contains designs for eleven subsystems. To date, five of these have been implemented namely

- ❖ Administrative Console and DSN Configuration
- ❖ Student Records Management
- ❖ Academic Registry Management
- ❖ Mail Merge
- ❖ Advanced Search

The above components comprise the core of the system that must be implemented by all those who wish to use the optional features.

Attainment of The Requirements

It is envisaged that the extent to which the requirements have been met and the extent to which the requirements themselves capture the needs of users will be evaluated during the first phase of acceptance testing. This testing is scheduled for August to September 2004. To date, the implementation has been checked to ensure that it is consistent with the requirements.

Design-Implementation Coherency

A considerable effort has been made to adhere to the design and to update it when it was overtaken by events. Some elements of the design are however no longer coherent with the implementation of some of the core components. In particular, a change was made to the interfaces after difficulties were experienced in implementing the vertical text only tab on MDI forms. The tabs were subsequently replaced with menus but the designs have not been updated to reflect this. The data dictionary is also no longer up to date particularly with regards to the lengths of text fields. These anomalies have been traced to the failure to adhere to the design philosophy with regards to updates due to pressure arising from implementation deadlines.

Quality of Implementation

The final implementation is of reasonable quality for a first release. The following inadequacies are however notable

a. Some of the Academic Registry routines use raw SQL for implementing database updates. This makes the code difficult to read and may lead to maintenance problems in the future

b. Student Records routines regularly update their recordsets to ensure they have up to date data. This technique should be extended to the routines in the Academic Registry

c. Error handling is quite coarse grained. An effort should be made to refine it so that errors are analysed before being output to the user. The primary obstacle in tackling this problem to date has been the tendency of ADO to throw a generic object error for all the problems it experiences.

d. Crystal reports regularly causes page fault errors in Windows. The causes of this problem have not yet been isolated.

Plans For Future Development

It is envisaged that the project will be submitted for acceptance testing over the months of August and September. Reviews will then be incorporated to ensure that a production release of the core components is available by the end of the year. The optional components will then be developed over the six months from January to June 2005. Production releases are slated for December 2005.

USER MANUAL

Chapter 1: Getting Started

What the CD Includes

The installation CD includes the following
1. An installation program for the infoQuest Core Clients. The installer includes un-installation support.
2. System Source Code (Visual Basic 6.0)
3. The System Development Manual in the Microsoft Word 2000 and Adobe PDF formats. For printing, the PDF version is recommended.
4. The User Manual in the Microsoft Word 2000 and Adobe PDF formats. For printing, the PDF version is recommended.
5. Original structure charts and form layouts in Microsoft PowerPoint format.
6. Windows 2000 Service Pack 3.
7. Adobe Acrobat Reader 5.0
8. Microsoft Word Viewer for Word 97/2000
9. PowerPoint Viewer for PowerPoint 2000
10. Installation files for the MySQL remote database system.

System Requirements

The following configuration is based on environments used during development. As such, they should be treated as mere recommendations as opposed to instructions. Where a specific brand of software is recommended, the recommendation should not be assumed to be the assumption of any liability, by the author that may arise from the use of the software.

InfoQuest Core Clients

The infoQuest core clients comprise DSN System configuration, Administrative Console, Student Records Management, Academic Registry Management, Advanced Search and the Mail Merge utility. The following is the recommended configuration
1. An IBM compatible Personal Computer running Microsoft Windows 95, 98, 2000 (SP3), ME, or XP
2. A printer capable of printing onto A4 paper.
3. CDROM Drive for facilitating installations.
4. Microsoft Word 2000. Please not that if you do not have Microsoft Word, you will not be able to spell-check your Mail Merge documents. All other mail merge functions will however be available.
5. Microsoft Excel 95. Please not that without Microsoft Excel, you will not be able to export advanced search results into the Excel format. You will however be able to export the other reports generated by infoQuest core clients.

Database Server
The infoQuest clients are able to connect to any database system for which an ODBC or OLEDB driver exists. This includes popular remote database systems such as MySQL, Microsoft SQL Server and Oracle. The system is also compatible with desktop database systems such as Microsoft Access but their use is not recommended, as they are not suitable for multi-user environments.

Given the freedom that infoQuest gives you in choosing your database system, you should carefully consider your alternatives before committing to any database system. In particular, you should consider the experience of your IT personnel, the financial resources you will be in a position to commit towards licensing fees and the system requirements of the database system. It is also important to consider the level of technical support that will be available for each database system under the licensing agreement that you will subscribe to.

Installing The System

In the following section, it is assumed that your CD-ROM is on drive D. If this is not the case then substitute you CD-ROM drive letter in the appropriate places. Please note that if you are installing on a Windows 2000 machine, you must first install Windows 2000 Service Pack 3. The installation file for WSP3 is included on the CD-ROM under D:\W2KSP3. You should also ensure that the database system has been properly configured.

(STOP)	Do not install the service pack unless you are **absolutely** sure that your machine does not already have WSP3 or any other later version of the Windows Service Pack. In addition, the Service Pack is meant for will cause unpredictable results with the worst-case scenario being the total loss of puter.

Installing the infoQuest Core Clients
a. Close any applications that you maybe running
b. Insert the CD into your CD-ROM drive
c. Copy the folder **D:\Core Installation** to your hard disk.
d. Navigate to the folder that you will have copied the files to and run Setup.exe
e. Follow the onscreen instructions.

(STOP)	Please note that attempting to install the program from read only media such as CD-ROMs may cause the system to crash.

Configuring The DSN

Before you can use the clients, you must configure a Data Source Name (DSN) that infoQuest will use to connect to your database system. To do this

a. Install the database system's ODBC or OLEDB driver on the machine on which you want to install the client(s).
b. Refer to your Operating System's documentation for information on setting up a Data Source Name.
c. After setting up the DSN, start the infoQuest DSN Configuration Wizard. The wizard is automatically installed when you install any of the Core Clients. To start it, select it from the Windows Start Menu under Programs in the infoQuest tab. You will be presented with the following screen.

d. Choose the **Next** button. You will then be presented with the following screen.

e. Choose the **DSN** button. This will open your operating system's DSN selection window. For instructions on selecting the DSN, click the window's **Help** button. If you select a DSN successfully, its details will be copied onto the white area just above the **DSN** button. You should not specify a username or password at this stage.

f. Choose the **Next** button and you will be presented with the following window

g. Enter the username and password that you will have been given by your Database Administrator. If infoQuest fails to connect to the data source, you will be prompted to resolve any of the problems that the system will be experiencing. If infoQuest successfully tests the connection, you will be presented with the following window

h. Choose **Finish** to end the Wizard.

	Where a database system provides you with a choice between an ODBC and an OLEDB connector, you are likely to experience better system performance if you use the OLEDB connector.

Logging Into The System

To ensure that only those who are authorised to use the system are the ones who gain access to it, infoQuest requires all users to log on and be authenticated by the database system. When you start any of the system components, you will be presented with the following screen

To log on,
1. Enter your username.
2. Enter your password.
3. Click on the **Log In** button.

	InfoQuest relies on your database system to authenticate users. As such, you should ensure that the database is configured to provide the level of security that you require. In particular, you should ensure that it does not allow anonymous logins.

InfoQuest Core Clients

Components	Remarks
Administrative Console	• Used to configure the system. • Can only be used by the administrator. • Explained in Chapter 2.
Academic Registry	• Used to administer the academic registry. • Explained in Chapter 3
Student Records Administration	• Used to administer student records. • Explained in Chapter 4
Advanced Search	• Used to execute SQL • Explained in Chapter 5
Mail Merge	• Used for automatic mailings • Explained in Chapter 6

Getting Assistance While You Work

All the windows that you are presented with within the system have online help support. You can get a quick summary of how to use the window by either

➢ Clicking on the **Help button**.
➢ Choosing **Help** from the **menu** if the window has one
➢ Or pressing the **F1** key.

In addition, most buttons and input boxes will give you a quick summary of what they do if you place the mouse over them for a few seconds.

Although the online system is handy, it is advisable that you skim through the entire user manual (Available on the CD) before trying anything out. You are also free to print out the manual.

If you are stuck or discover a bug within the system, you may be able to get assistance by contacting the system developer at tgmotsi@yahoo.com. You may also want to have a look at the infoQuest website at www.geocities.com/tgmotsi which will contain updates for the system.

Chapter 2: The Administrative Console

The Administrative Console provides you with tools for setting up the database tables that infoQuest uses and configuring user accounts for online access. Please note that the user accounts configured here are not the same as the one you will have to setup for each user who will be using the Core Clients. The console is available from the infoQuest program group.

Creating System Tables

a. Select **Initialise Table** from the **Admin** menu
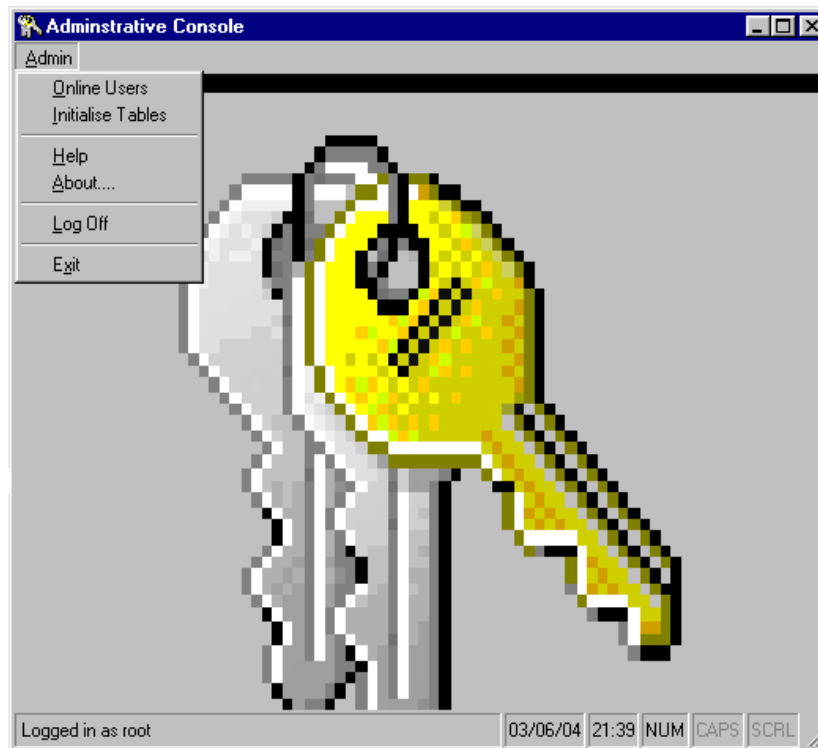
b. The Table Initialisation will start and you will be presented with the following window. Please note that if your database already contains infoQuest tables, the initialisation will fail.
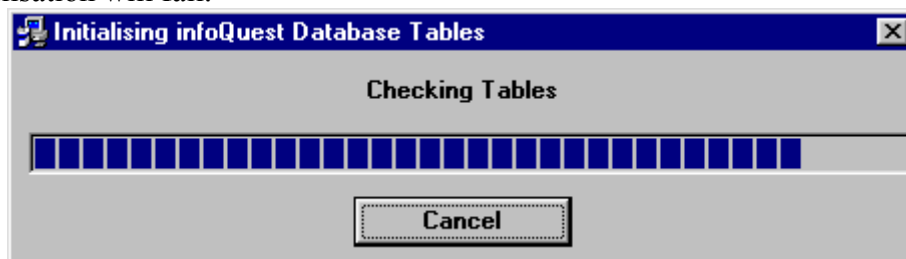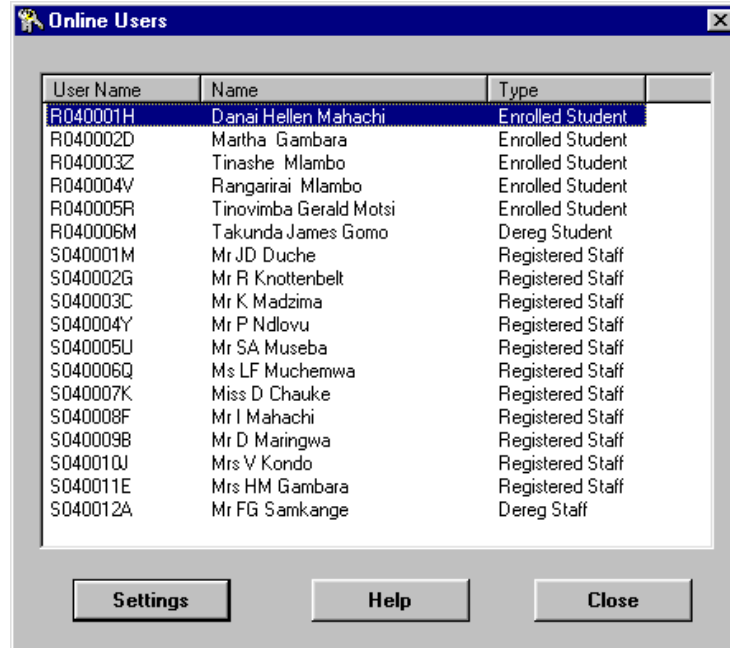
c. If the table initialisation is successful, you will be presented with a message to that effect.

Managing Online Users

Online user accounts are used for logging onto web based clients llike the Nominal Roll. In general, all system users should have an online user account. Those who use the Core Clients should in addition to the online account, have a database account preferable with a different password.

Adding New Users

The process of adding new users is carried out by infoQuest on your behalf when you start the Online User Administration service.

User Name	Name	Type
R040001H	Danai Hellen Mahachi	Enrolled Student
R040002D	Martha Gambara	Enrolled Student
R040003Z	Tinashe Mlambo	Enrolled Student
R040004V	Rangarirai Mlambo	Enrolled Student
R040005R	Tinovimba Gerald Motsi	Enrolled Student
R040006M	Takunda James Gomo	Dereg Student
S040001M	Mr JD Duche	Registered Staff
S040002G	Mr R Knottenbelt	Registered Staff
S040003C	Mr K Madzima	Registered Staff
S040004Y	Mr P Ndlovu	Registered Staff
S040005U	Mr SA Museba	Registered Staff
S040006Q	Ms LF Muchemwa	Registered Staff
S040007K	Miss D Chauke	Registered Staff
S040008F	Mr I Mahachi	Registered Staff
S040009B	Mr D Maringwa	Registered Staff
S040010J	Mrs V Kondo	Registered Staff
S040011E	Mrs HM Gambara	Registered Staff
S040012A	Mr FG Samkange	Dereg Staff

| Settings | Help | Close |

InfoQuest scans the database for registered students, deregistered students, registered members of staff and deregistered members of staff who will not have online accounts and then adds them. In the process, it assigns a random password to each user which should be reset as outlined below before the user can begin using the system.

Configuring User Accounts

a. Select the **Settings** button. You will be presented with the following window

b. To reset the user's password, enter the new password in the **Password** field and re-enter it in the **Confirm Password** field.
c. To allow the user to access the Nominal Roll web service, click the **Nominal Roll** box
d. To allow the user to access student's past assessment records, click the **Assessment** box.
e. Choose **Save**

Chapter 3: The Academic Registry

The Academic Registry comprises Departments, Members of staff, Classes and Subjects. Because you cannot register students until you have registered some classes, you should start by entering all of the Academic Registry information with the exception of the Subjects information which should be entered after you have registered some students.
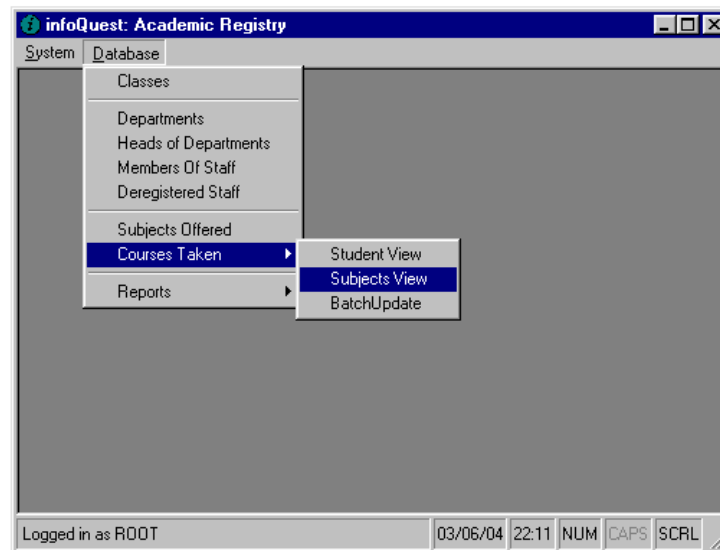
A Guide to infoQuest Icons

In this and the next chapter, reference will be made to the following standard infoQuest icons.

Icon	Description
	Create a new record
	Edit the currently displayed record
	Delete the displayed record. For students and member of staff, the icon is for deregistration.
	Reregister a deregistered member of staff or student.
	Find a record
	Obtain a system wide report.
	Display online help for the task you are carrying out.
	Close the window.

Departments
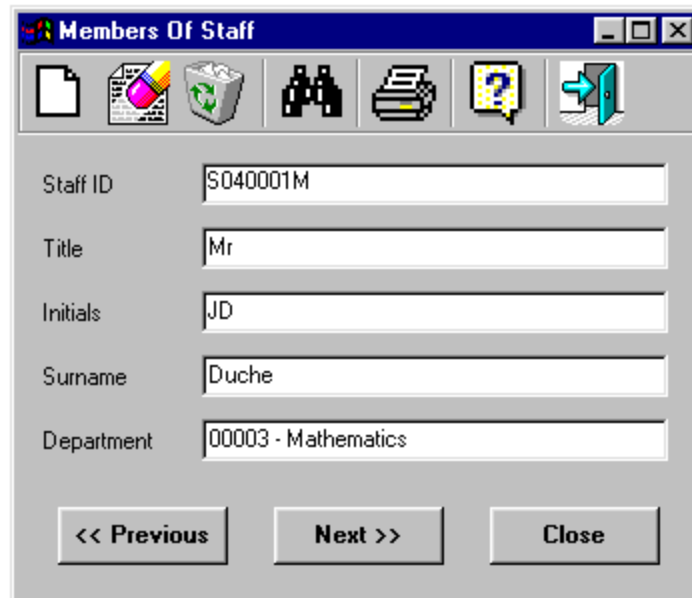
To access the Departments' window, choose **Departments** from the **Database** menu shown above. You will be presented with the following Window
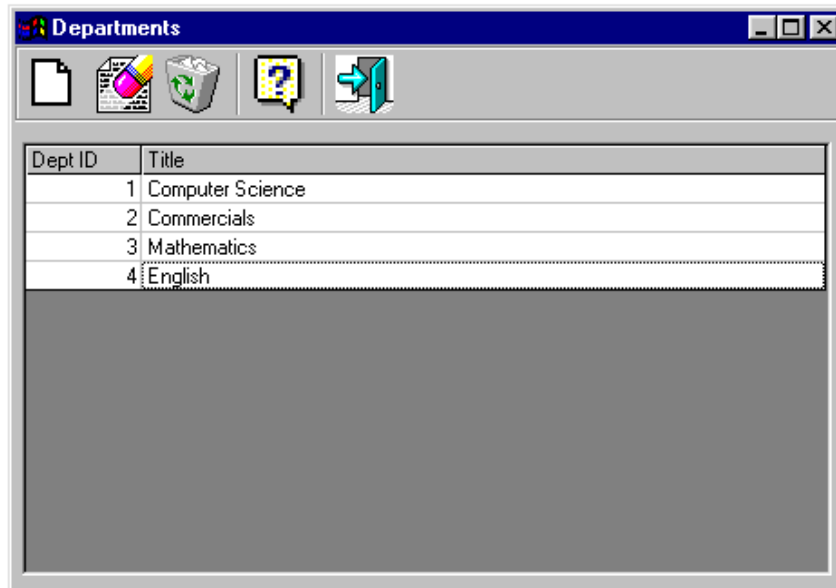
1. To add a new department
   a. Choose the **New** icon
   b. You will be prompted to enter the name of the department. If the name is not unique, it will be rejected.
   c. If the addition is successful, the new department's name will appear in the list of departments.
2. To edit a department's name
   a. Choose the **Edit** icon
   b. You will be prompted to enter the new name for the department. If the name is not unique, it will be rejected.
   c. If the edit is successful, the department's new name will appear in the list of departments.
3. To delete a department
   a. Select the **Delete** icon
   b. You will be prompted to confirm the deletion. Please note that if there are any members of staff who will be registered as being members of that departments of if there any subjects that will still be registered under the department, the deletion will be rejected.
   c. If the deletion is successful, the updated list of departments will not contain the department.

Members of Staff

For employees at the school to have online user accounts, they must be registered as members of staff. You should therefore register all members of staff who may need to use the system even if they are not teachers. To access the Members of Staff window, choose **Members of Staff** from the Academic Registry's Database menu.

1. To browse through the records you can use the **Previous**, **Next** and the **Search** button. The search button allows you to jump to the record of any member of staff if you know his/her Staff ID.
2. To register a new member of staff
   a. Choose the **New** icon

b. You will be presented with a form that will be blank with the exception of the **Staff ID**. Enter the title, initials and surname of the member of staff in the spaces provided.
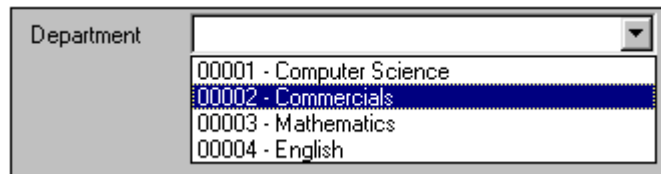
c. Select the member of staff from the list provided

    d. Choose the **Save** button

3. To change a member of staff's details
    a. Choose the **Edit** button
    b. Change the details you want to change. Please note that you will not be able to change the Staff ID.
    c. Choose Save

4. To deregister a member of staff
    a. Choose the **Deregister** icon.
    b. You will be prompted to confirm the deregistration. Please note that the deregistration will fail if the member of staff is still being referenced in the system. You can check the references by request a system wide report as outlined below.
    c. If the deregistration is successful, you will get a message to that effect.

5. Obtaining a system wide report
    a. Choose the **Report** icon

b. InfoQuest will scan the system for information about the member of staff and then provide the system to you in a comprehensive report that you will be able to save to your computer and/or print
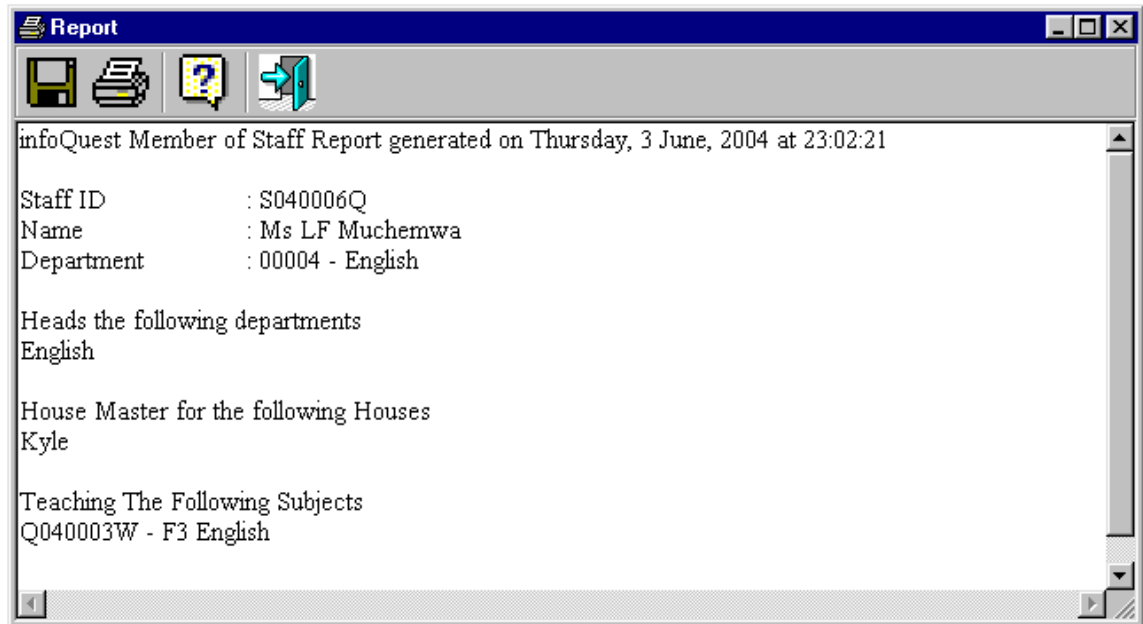
c. Icons for saving and printing respectively

Heads of Departments

To open the HODs window, choose **Heads of Department** from the **Database** menu. You will be presented with the following window
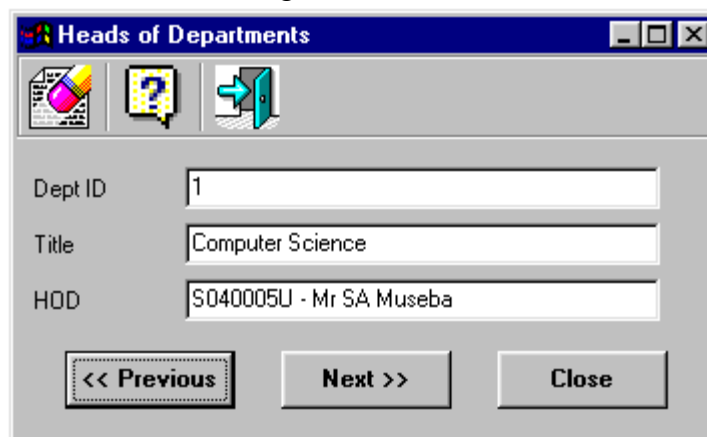
This window enables you to specify the HODs for the departments you will have created using the **Departments** window. To specify a Head of Department
    a. Locate the record of the department by browsing through the records.
    b. Choose the **Edit** icon

c. Select the HOD from the list that will be provided

d. Choose **Save**

Deregistered Members of Staff

The records of deregistered members of staff are provided for reference only. You can neither delete nor edit them.
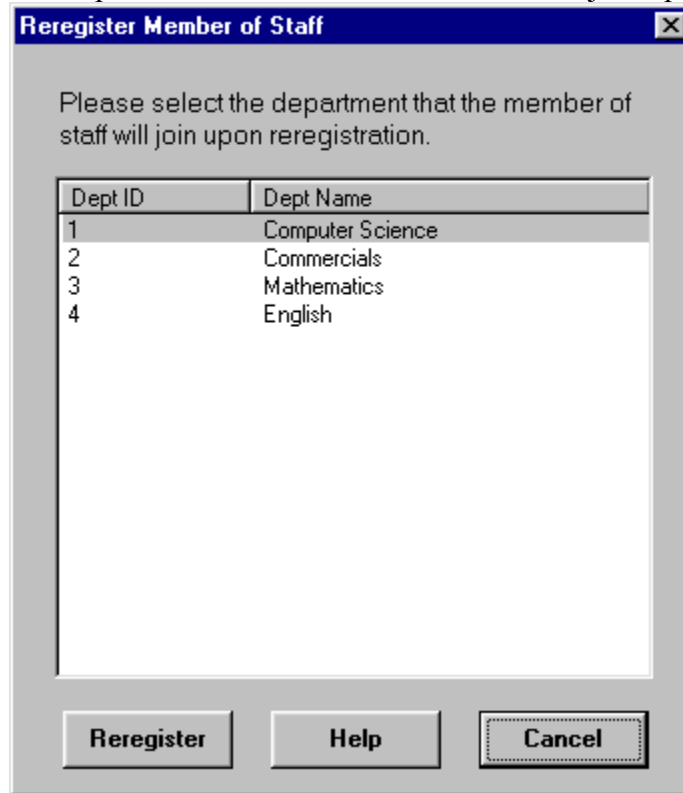
To add a new record, you must deregister an existing member of staff. In the event that you wish to reregister a member of staff whom you will have deregistered, proceed as follows

a. Select the **Reregister** icon

b. You will be presented with a list of departments from which you should choose the department that the member of staff will join upon re-registration.

c. Choose the **Reregister** button

Classes

1. To access the classes' window, choose **Classes** from the **Database** menu.
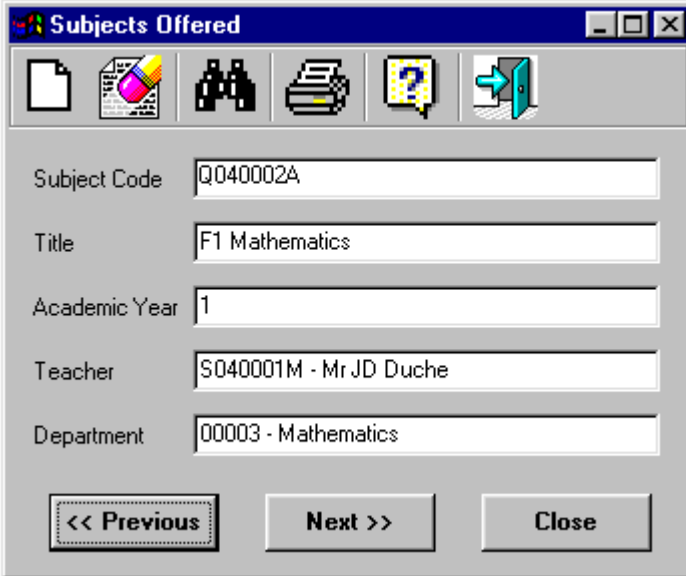
2. To add a new class
    a. Choose the **New** icon. You will obtain a form pre-filled with the Class ID
    b. Enter the class name which should be unique
    c. Enter the academic year
    d. Select the Form Teacher from the list provided
    e. Choose **Save**

3. To edit a record
    a. Choose the **Edit** icon.
    b. Change the details you want to change. Please note that you cannot change the Class ID.
    c. Choose **Save**
4. To delete a class, choose the **Delete** icon. Please note that you cannot delete a class that is in use.
5. To obtain a report detailing the students in the class, choose the **Report** icon.

Subjects Offered

The manage the list of subjects offered by the school, select **Subjects Offered** from the Database menu. You will be presented with the following window

You can use the window to add new subjects and to edit the details of subjects offered. You however cannot delete a subject once you have created it.

Subjects Taken

There are three ways of managing the subjects taken namely Student View, Subject View and Batch Update. To access the services select **Subjects Taken** from the Database menu. You will be presented with a sub-menu from which you can then select the service you want.

Student View

The student view shows you all the subjects for which a given student will be registered. You can add and remove subjects from this list. Please note that you cannot change the student details displayed.

1. To add a subject to the list
   a. Select the **Add Subject** button. You will be presented with the window shown below.
   b. Specify how you would like to search from the subject from one of the following
      i. Subject code
      ii. Subject Title
      iii. All registered subjects
   c. If you are searching by subject code or title, enter the subject code or title.
   d. Select the **Search** button.
   e. The matching subjects will be listed in the **Available** box. You can move them to the **Selected** list by using the arrows. To find out what each arrow does, place your mouse over it for a few seconds.
   f. Note that you can repeat steps (b) to (e) to find other subjects which you can add to the list you will have.
   g. Whey you have finished selecting the subject you wish to add, choose the **Add** button. The subjects will be added with duplicates being rejected.
2. To remove a subject, select it and then choose the **Remove Subject** button.

3. To remove all the subjects at once, choose the **Remove All Subjects** button.

Add Subjects

Available

- ● Subject Code
- ○ Subject Title
- ○ All Subjects

Search

Subject Code	Title	Teacher	

↓ ⬇ ↑ ⬆

Selected

Subject Code	Title	Teacher	

Add **Help** **Cancel**

Subject View

The subject view is similar to the student view and the same techniques used in the **Student View** may be used in the subject view.

Subjects Taken - Subjects View

Subject Code: Q040002A
Title: F1 Mathematics
Teacher: S040001M - Mr JD Duche

Students

Student ID	Surname	First Name	Class
R040002D	Gambara	Martha	2A1
R040001H	Mahachi	Danai	2A1
R040005R	Motsi	Tinovimba	2A1
R040003Z	Mlambo	Tinashe	2A1

<< Previous **Next >>** **Close**

The Batch Update Wizard
1. Start the Batch Update wizard. You will be presented with the following windows
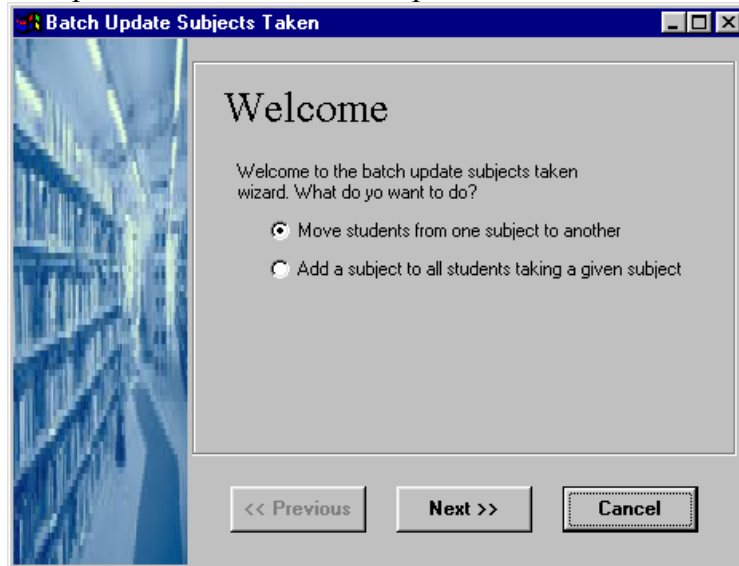
2. Choose one of the options given and then choose **Next**. You will be presented with another window for specifying either the subject you want to replace or the one to which you want to add another subject. Select the subject and then choose **Next**
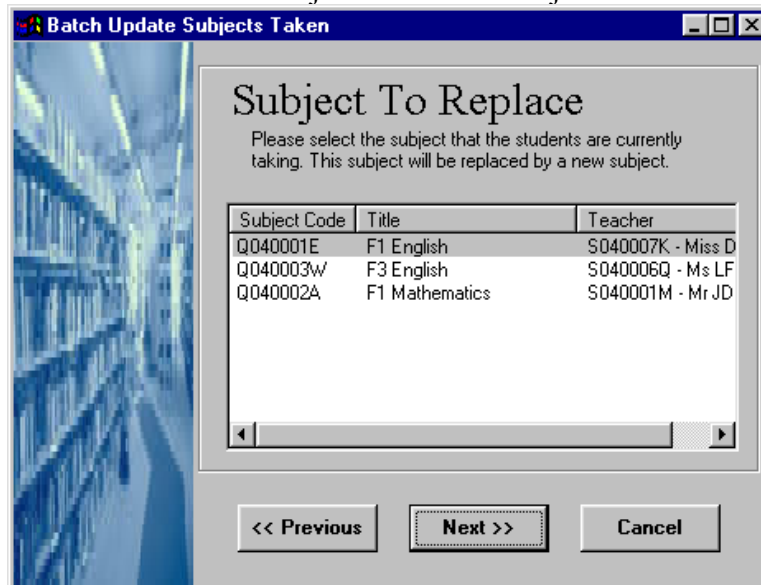
3. You will be presented with another window for specifying the subject to replace with or to add. Select the subject and then choose **Next**
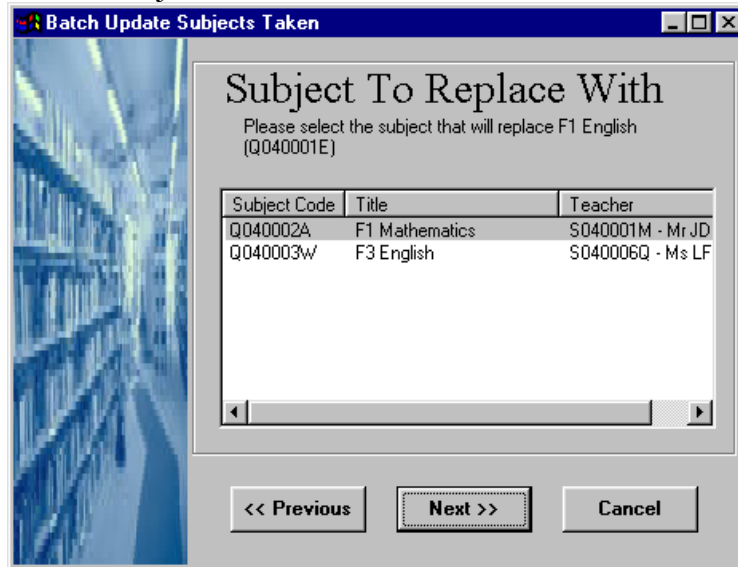
4. If the replacement/addition is successful, you will be presented with a window displaying a message to that effect.

Reports

In addition to the reports that are available from the individual windows, you can also obtain other reports by choosing **Reports** from the **Database** menu of the Academic Registry. A sample reports showing a list of all the classes is show below.
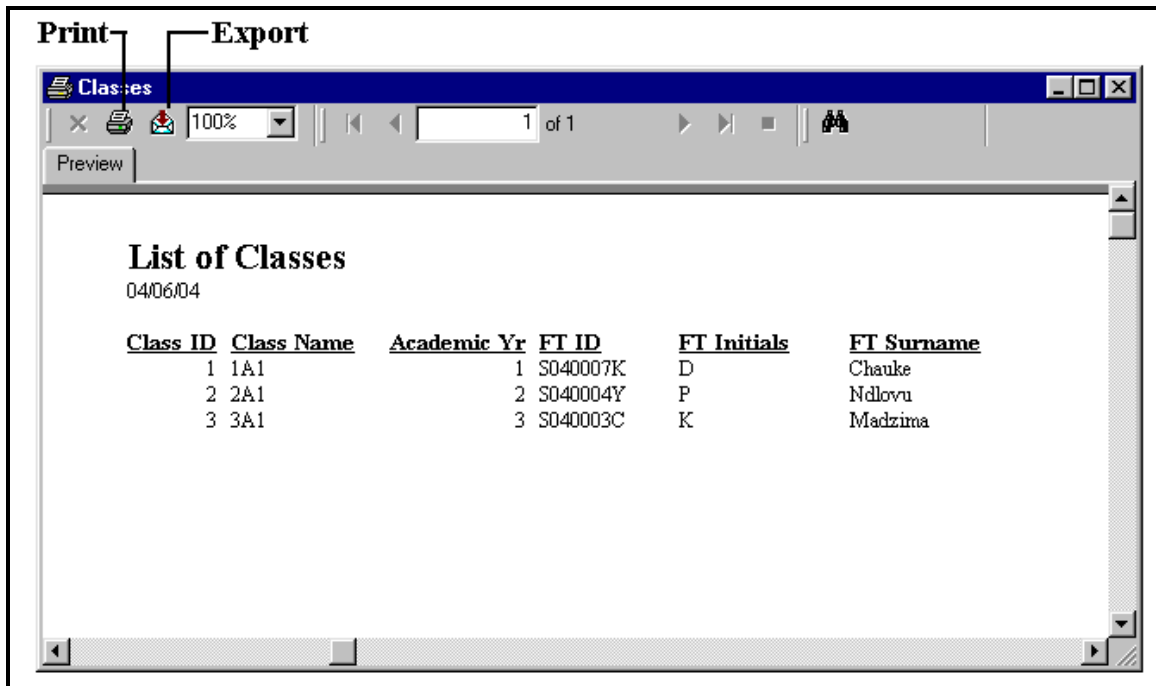
The reports can be printed or exported to a variety of formats by choosing the Print or Export buttons respectively. Formats supported for exporting include Excel, HTML and PDF among others.

Chapter 4: Student Records

The Student Records client provides facilities for managing details relating to students such as Doctors, Parents, Medical Aid Details and Houses in addition to the records of the students themselves.
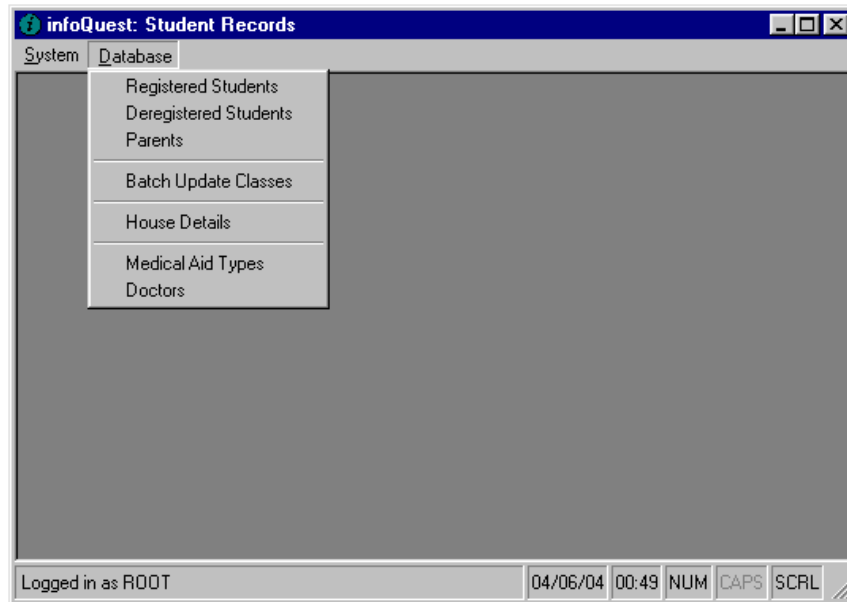
Doctors' Details

To open the Doctors' window, select Doctors from the Database menu show above. You will then be presented with the following window.
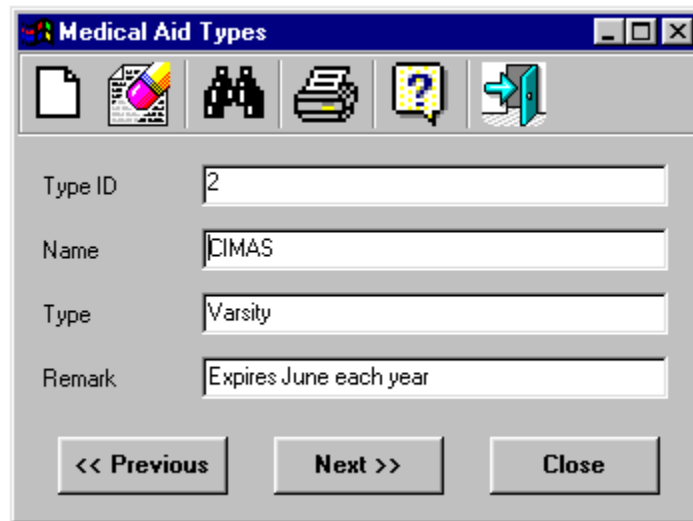
1. To browse through the records, you can use the **Previous**, **Next** and **Search**

2. To register a new doctor
    a. Select the **New** icon
    b. The Doctor ID will be pre-filled for you. Enter the other details
    c. Choose the **Save** button
3. To edit a doctor's record, choose the **Edit** button. Please note that you cannot change the Doctor ID.
4. Please note that you cannot delete Doctors' records
5. To obtain a list of the students who will be clients of the doctor, choose **Report** icon

Medical Aid Details

The Medical Aid window is accessible from the Database menu and looks similar to the window show below
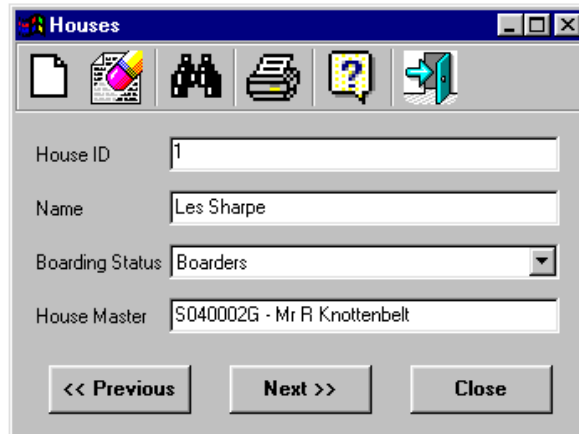
The window behaves in a similar way to that for doctors. It should however be noted that there is a restriction on the values you may enter. The combination of the name and type must be unique. For example, given the record shown above, one could register another medical aid type with a type of **varsity** as long as its name is not **CIMAS**.

House Details

The House Details window is accessible from the Database menu and looks similar to the window shown below.

This window also behaves in way that is similar to the Doctors' window. However, when entering a new record or editing an existing one, you can only type in the House Name. For the **Boarding Status** and the **House Master**, you must select your entry from the lists that will be provided. The boarding status should be one of the following: Boarders, Day Scholars, Mixed or Weekly Boarders. The house master list will contain all the members of staff so the housemaster must be a registered member of staff.

Parents

To access the Parent's window, choose **Parents** from the **Database** menu. Please note that when a member of staff is also a parent and/or a guardian, you will need to register him/her as a parent in addition to the registration as a member of staff. Also note that all the details in window will be required during editing and the making of new records. However, you can enter a '-' for all the details you will not be having. A sample Parent window is shown below.

Student Records

This window provides you with a facility for recording student details. Please note however that it does not allow you to add or change the details of parents, doctors, medical aid types or houses. You must therefore register all these details before you attempt to register a student who requires them. A screen shot of the window is show below.

When entering details on the form please note that

a. You cannot change the Student ID and the date on which the record was last updated.

b. The **Family** Tab gives you lists for entering the language, religion and parents' marital status of the student. You are however still able to type in a value if it does not exist in the list.

c. For the **Doctor, Medical Aid, House**, **Class, Guardian**, **Mother** and **Father** tabs, you must select the value to record by choosing the **Select** button shown below

d. The **Relatives** tab enables you to add and remove the students' relatives at the school. The **Add** button will start a wizard that will help you locate the relative's details from the infoQuest system. Once added, you can use the **Edit** button to edit the relationship and the **Remove** button to remove the relationship.

e. To deregister a student, choose the **Deregister** icon. You will then be presented with the following window which you should fill. After filling in the required details, choose the **Deregister** button.
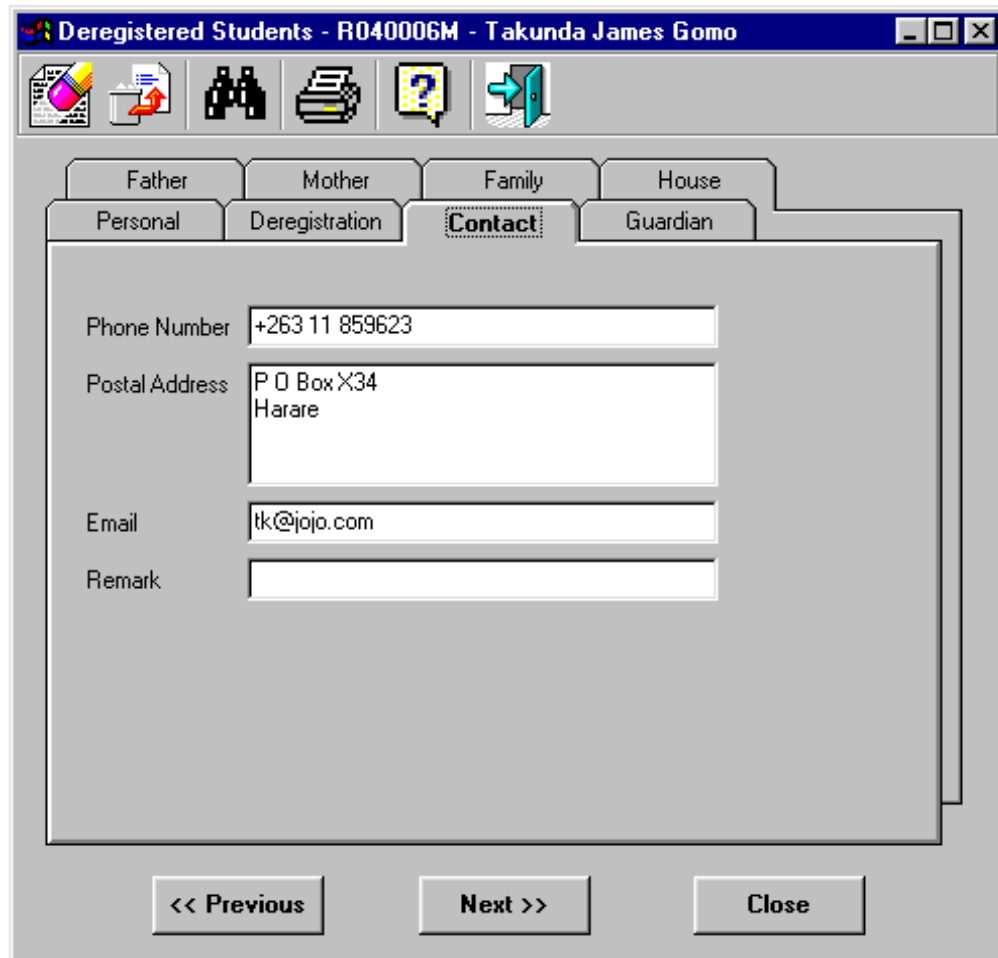
Deregistered Students

The deregistered students facility archives the details of deregistered students. A screen shot of the window is shown below.

With the exception of the details recorded under the **Contact** tab, you cannot edit deregistered students' details. You can however reregister a student by selecting the **Reregister** icon and then filling out the form show below which is similar to the **Registered Students** form

Reregister Student - R040006M - Takunda James Gomo

Tabs: Doctor | Medical Aid | Relatives | Class | House
Personal | Family | Guardian | Father | Mother

Personal

Field	Value
Student ID	R040006M
First Name	Takunda
Middle Name	James
Surname	Gomo
Sex	Male
Date Of Birth	Wednesday, 7 November, 1984
Date Enrolled	Sunday , 12 January , 1997
Last Updated	Friday, 4 June, 2004
Birth Certificate No.	14 5263 8596
Remark	

Reregister   Help   Cancel

Batch Updating Classes

At the end of the year, you may need to move all of the students in a given class to a new class. InfoQuest provides a facility for automating such a process. The facility can be accessed by choosing **Batch Update Classes** from the Database menu.

| Batch Update Classes	⊠
Current Class	00002 - 2A1 ▼
New Class	00001 - 1A1 ▼
Update	Help	Close
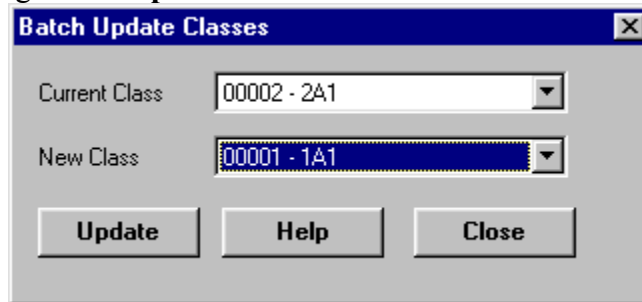
You may find the following strategy useful

    a. Move all the Upper sixes and Form Fours into specials classes e.g. U6 Sc Leavers for all those who were in U6 Sc

    b. Start with the Lower sixes and end with the Form ones.

Chapter 5: Finding Information

The Advanced Search Utility, shown in the diagram below, allows you to execute queries on the database for as long as those queries do not seek to modify data. This utility will allow you to find any information you may want from the database provided you know the SQL required for the query. In addition to knowing the SQL, you will also need to know the database structure. For information on the database table structure, consult your database administrator.
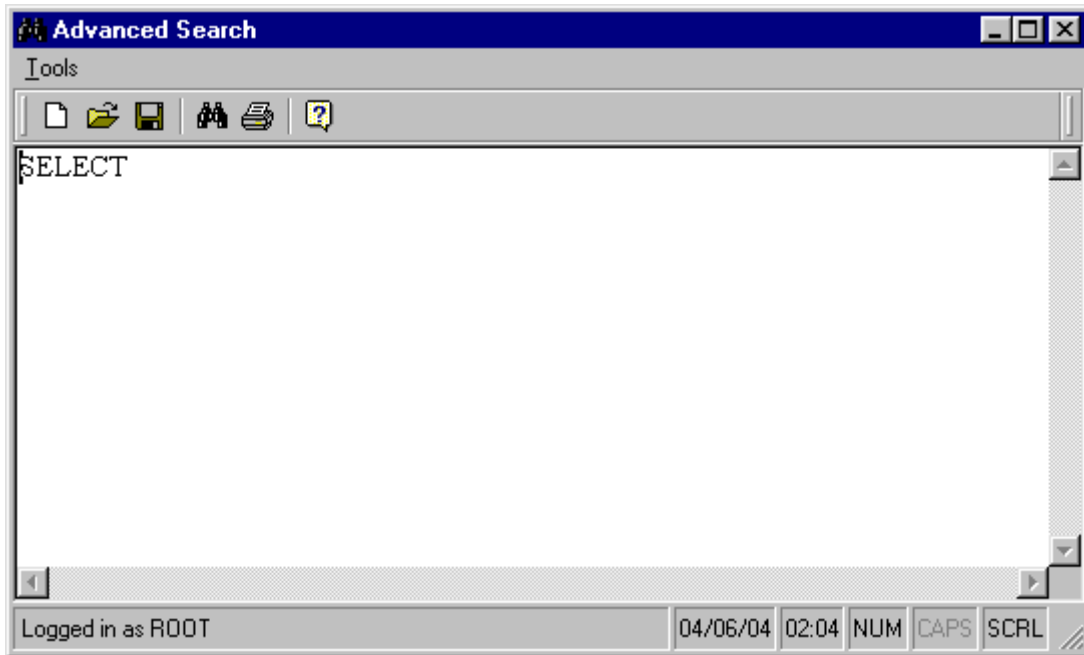
The main search window, pictured above, allows you to manage your SQL files by saving and opening them later. The following is a summary of what the functions provided under the **Tools** menu of the main window do.

Menu Item	Remarks
New Query	Erases the query window allowing you to start working on a new query.
Open SQL File	Opens a previously saved SQL file.
Save SQL File	Saves the currently displayed SQL file under its current filename if it already has one.
Save SQL File As	Saves the displayed SQL file under a different filename.
Execute Query	Presents the search results for the displayed query.
Print SQL	Prints the displayed SQL file text.

Viewing Query Results

After entering your SQL or opening a predefined query, you can view the results it returns by choosing **Execute SQL** from the **Tools** menu. If the query finds any matches you will be presented with the following results window

Search Results

	Staff_ID	Title	Surname	Initials	Dept_ID	Dept_ID	Dept_Title
1	S040001M	Mr	Duche	JD	3	3	Mathematics
2	S040002G	Mr	Knottenbelt	R	3	3	Mathematics
3	S040003C	Mr	Madzima	K	1	1	Computer
4	S040004Y	Mr	Ndlovu	P	1	1	Computer
5	S040005U	Mr	Museba	SA	1	1	Computer
6	S040006Q	Ms	Muchemwa	LF	4	4	English
7	S040007K	Miss	Chauke	D	4	4	English
8	S040008F	Mr	Mahachi	I	2	2	Commercials
9	S040009B	Mr	Maringwa	D	2	2	Commercials
10	S040010J	Mrs	Kondo	V	3	3	Mathematics
11	S040011E	Mrs	Gambara	HM	4	4	English
12							
13							

Page 1 of 1

From the above show results window, you can do the following by choosing the appropriate function from the Tools menu.

➢ Adjust column widths so that they display the full details.
➢ Export the results to Microsoft® Excel
➢ Export the results to a generic format understood by all spread sheets
➢ Adjust the row height. For example you can instruct the system to display 4 lines of text for each record so that you can view the addresses clearly.

❗	Addresses do not export well to the generic format. You will probably get a better result if you export to the Excel Format and then try to import the results from the excel file into your spreadsheet program.

Using Pattern Matching In Your Searches

You use wildcard characters as placeholders for other characters when you are specifying a value you want to find and you:

- ➢ Know only part of the value.
- ➢ Want to find values that start with a specific letter or match a certain pattern.

In the Search and Advanced Search utility you can use the following characters to find such field value.

| Character	Usage	Example
*	Matches any number of characters. It can be used as the first or last character in the character string.	wh* finds what, white, and why
?	Matches any single alphabetic character.	B?ll finds ball, bell, and bill
[ ]	Matches any single character within the brackets.	B[ae]ll finds ball and bell but not bill
!	Matches any character not in the brackets.	b[!ae]ll finds bill and bull but not bell
-	Matches any one of a range of characters. You must specify the range in ascending order (A to Z, not Z to A).	b[a-c]d finds bad, bbd, and bcd
#	Matches any single numeric character.	1#3 finds 103, 113, 123

Notes

- ➢ Wildcard characters are meant to be used with text data types, although you can sometimes use them successfully with other data types, such as dates, if you don't change the Regional Settings properties for these data types.

- ➢ When using wildcard characters to search for an asterisk (*), question mark (?), number sign (#), opening bracket ([), or hyphen (-), you must enclose the item you're searching for in brackets. For example, to search for a question mark, type **[?]**. If you're searching for a hyphen and other characters simultaneously, place the hyphen before or after all the other characters inside the brackets. (However, if you have an exclamation point (!) after the opening bracket, place the hyphen after the exclamation point.) If you're searching for an exclamation point (!) or closing bracket (]), you don't need to enclose it in brackets.

- ➢ You can't search for the opening and closing brackets ([ ]) together because the Microsoft® Jet query system interprets this combination as a zero-length string. You must enclose the opening and closing brackets in brackets ([[ ]]).

Chapter 6: Mail Merge

The Mail Merge facility allows you to write a standard letter from which the system can then generate personalised letters using a query that you specify. When you start Mail Merge, you will be prompted for a **Query Source File**. This is the file that the system uses to obtain the personalised details.

Before supplying a query file, it is often prudent for you to open it using the Advanced Search utility so that you can be assured that it returns the records that you want. Once you have specified the source query file, you will be presented with the following window.
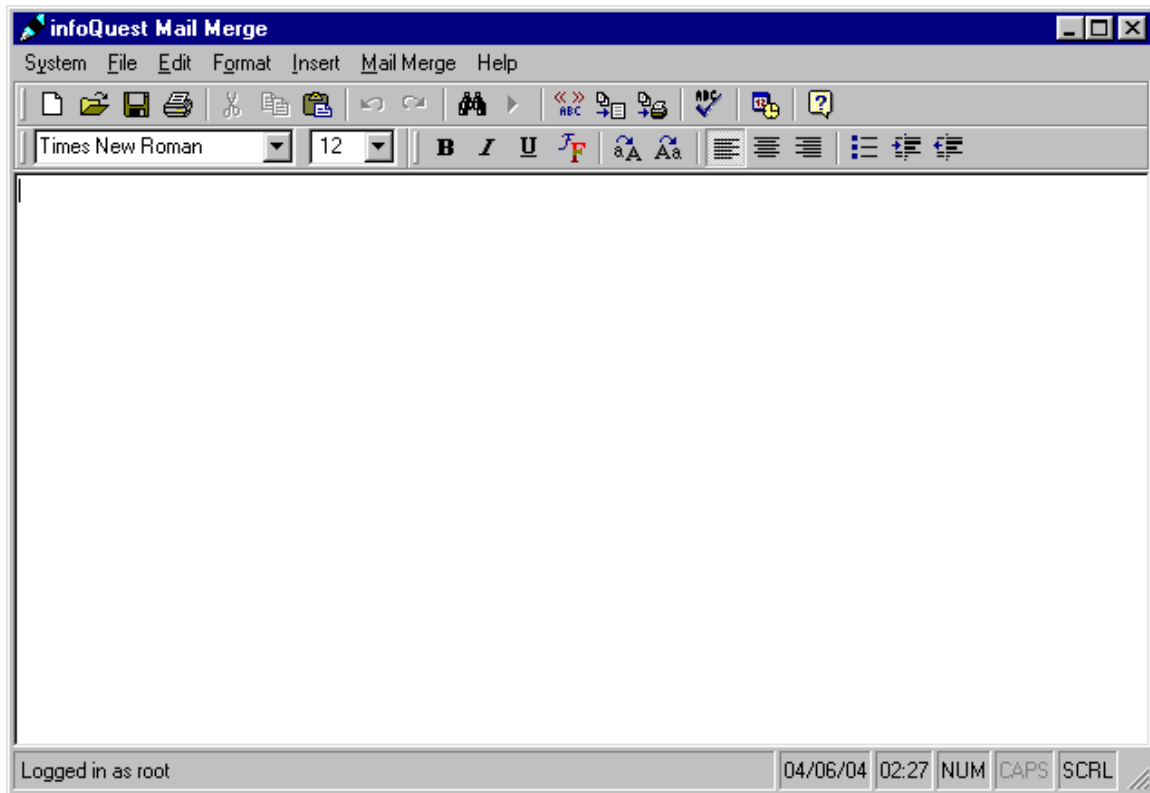

The above window is a typical document editor with added features for mail merge processing. You will find that you will be able to perform most of the functions you will need by selecting the toolbar buttons. To find out what a particular toolbar button does, place the mouse pointer over it for about 3 seconds and you will get a tool tip.

Text Editing Functions

After you have specified the source SQL file, you must then supply a document giving the standard letter. To do this, you can either type a new letter or open an existing one which you can modify, if you wish, before you run the mail merge. The editor supports several file handling and document formatting functions. To get a feel of what a standard

letter looks like, have a look at the sample provided in your **Letters** folder that is also reproduced below.
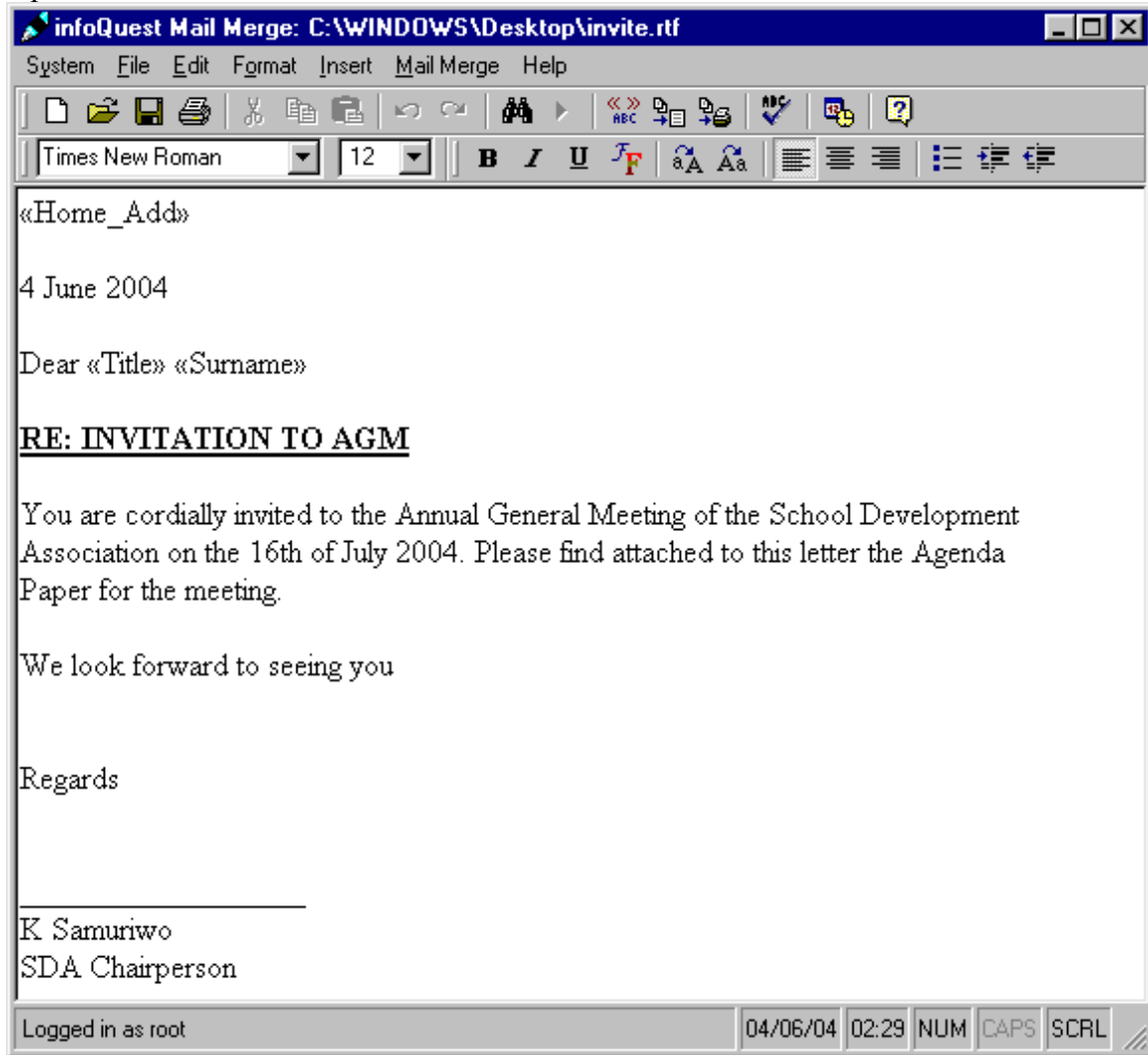
Creating A New Document
When you start the Mail Merge utility, it automatically loads a new document for you. If during your mail merge job you decide to create another document that will **use the same** SQL file, choose the '**New**' icon in the document toolbar.

To create a new document that will use a **different** SQL source file, use the **Start New Session** option outlined below instead.

Opening A File
1. Click the **Open** icon in the toolbar.
2. Locate the file in the window provided and double click it. If you come across anything in this window that you do not understand, you can get additional help by clicking the **What's This** icon shown in the diagram below and then clicking the item you want to find more information about.
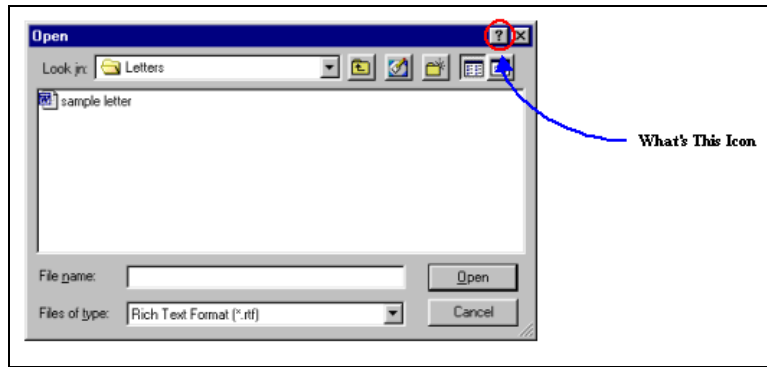
Saving A File
1. Click the **Save** icon. 💾
2. If the file does not have a filename, you will be prompted to specify a filename. If you need more information on any of the items in the windows that will be displayed, click the "**What's This**" help button and then click the item you want additional information on.

Saving A File Under A Different Filename
If you wish to save the changes you will have made to the existing file but still want to keep the old version, you must save the displayed file under a different filename. To do this,
1. From the **File** menu, choose "**Save As**".
2. You will be prompted to specify a filename. If you need more information on any of the items in the windows that will be displayed, click the "**What's This**" help button and then click the item you want additional information on.

Printing the Standard Letter
1. Click the Print icon. 🖨
2. A standard Print window from which you should choose the Print button will then prompt you. If you need more information on any of the items in the windows that will be displayed, click the "**What's This**" help button and then click the item you want additional information on.

Searching For Words or Phrases
There will be times when you will want to locate a particular word or phrase from your standard letter. Steps for doing this are outlined below.
1. Choose the New Search icon 🔍

2. You will be resented with the following window.

3. In the **Find What** box, enter the text you want to locate.
4. Select the **Whole Word** option if you would like to find whole word only.
5. Select the **Match Case** option if you would like the matches to be only returned if they match the word you specified in terms of capitalisation.
6. Select **Find Next** to locate the first match.
7. To find additional matches either
   a. Choose the **Find Next** button.
   Or
   b. Close the Find window by selecting **Cancel** and then press the click the **Find Next** icon ▶

To start a new search, repeat the above steps. If the **Find** window is already open, simply type in the new word you want to look up and then choose the **Find Next** button.

Replacing Words or Phrases
1. Select **Search and Replace** from the **Edit** menu.
2. You will be presented with the following window.
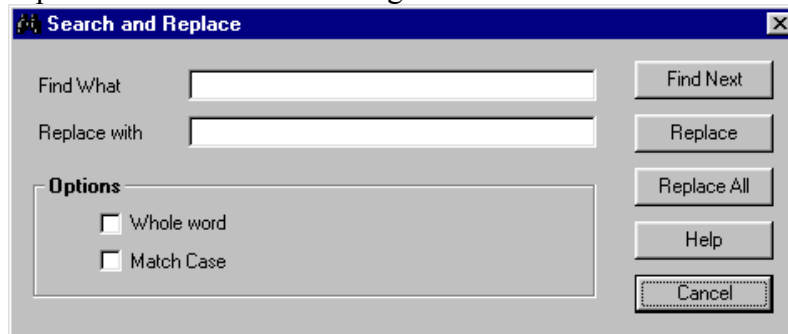
3. In the **Find What** box, enter the text you want to replace.
4. In the **Replace With** box, enter the text you would like to replace the text you will have specified above.
5. Select the **Whole Word** option if you would like to find whole word only.
6. Select the **Match Case** option if you would like the matches to be only returned if they match the word you specified in terms of capitalisation.
7. If you want to replace all the occurrences in just "one go" then select the **Replace All** button. To replace items one by one,
   a. Click the **Replace** button. If a match is currently selected, the system will replace that single occurrence. If there is no match selected, the system will

find the next match, select it and replace it the next time you click the **Replace** button

b. If you come across an occurrence that you would like to skip over without replacing, select **Find Next** when it is selected.

c. You can stop replacements at any time by choosing the **cancel** button.

Spell Checking Your Document

1. Select **Spell Check** from the Edit menu or select the Spell Check icon from the toolbar.
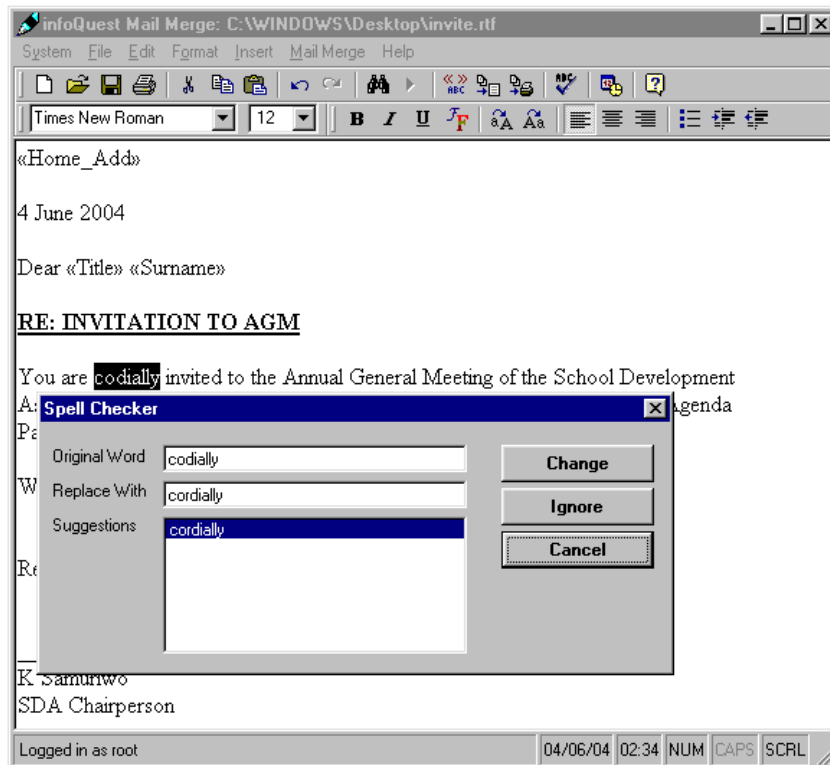
2. If a spelling error is discovered you have three options

    a. Replace the word with one of the suggestions.
        i. Select the suggestion from the list
        ii. Choose **Change**

    b. Replace the word with a word that you type in yourself.
        i. Type in the replacement word in the **Replace With** box
        ii. Choose Change

    c. Ignore the spelling error by choosing **Ignore**

Cutting, Copying and Pasting

The Levy system provided standard cut, copy and paste functions for text.

➢ To copy some text, select the text and then select the copy icon from the toolbar. The keyboard shortcut is **Ctrl⊕ C.**

➢ To cut some text, select the text and then select the cut icon from the toolbar. The keyboard shortcut is **Ctrl⊕ X.**

➢ To paste some text, place the cursor where you want the text to be inserted and then select the paste icon from the toolbar. The keyboard shortcut is **Ctrl⊕V.**

These functions **may** work with pictures but this is not guaranteed. To get reliable results with pictures, use the keyboard shortcuts instead.

Dealing with Fonts

The document formatting used in the mail merge facility supports several fonts, font sizes and text effects like colour and underlining. To use these formatting facilities,

1. Select the **Fonts** icon.
2. You will be presented with the following window.
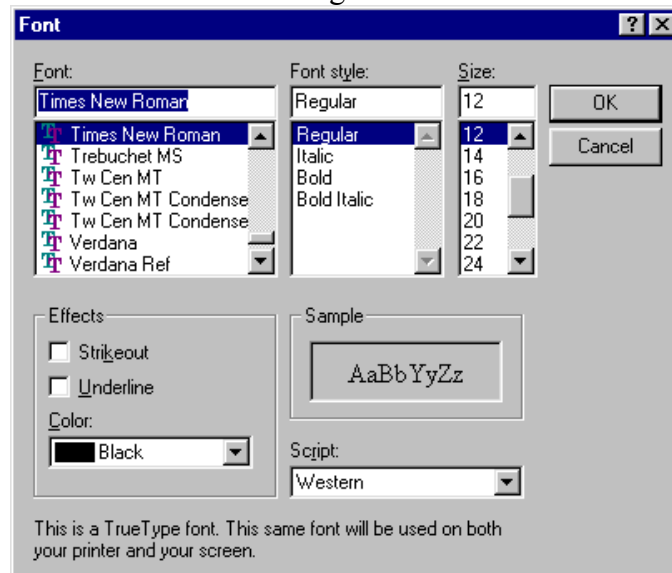
3. Select the options you want. Note that the window your operating system presents to you may look a bit different.
4. Choose **OK**.

Alternatively, you can use the toolbar formatting functions. To find out what each does, place your mouse over its icon for about 3 seconds.

Changing Case

You can change any selection of text to all capitals or small letters. To convert a selection into all capitals, select the capitalise icon. For the reverse, use the "To Lower Case" option.

Inserting a Date Into Your Document

1. Place the cursor where you want the date to be inserted.
2. Select the **Insert Date** icon.
3. You will be presented with following window. Double click the date you wish to include. To insert a date not displayed in the calendar, use the buttons in the top left and right corners to browse to the calendar for the year you want.
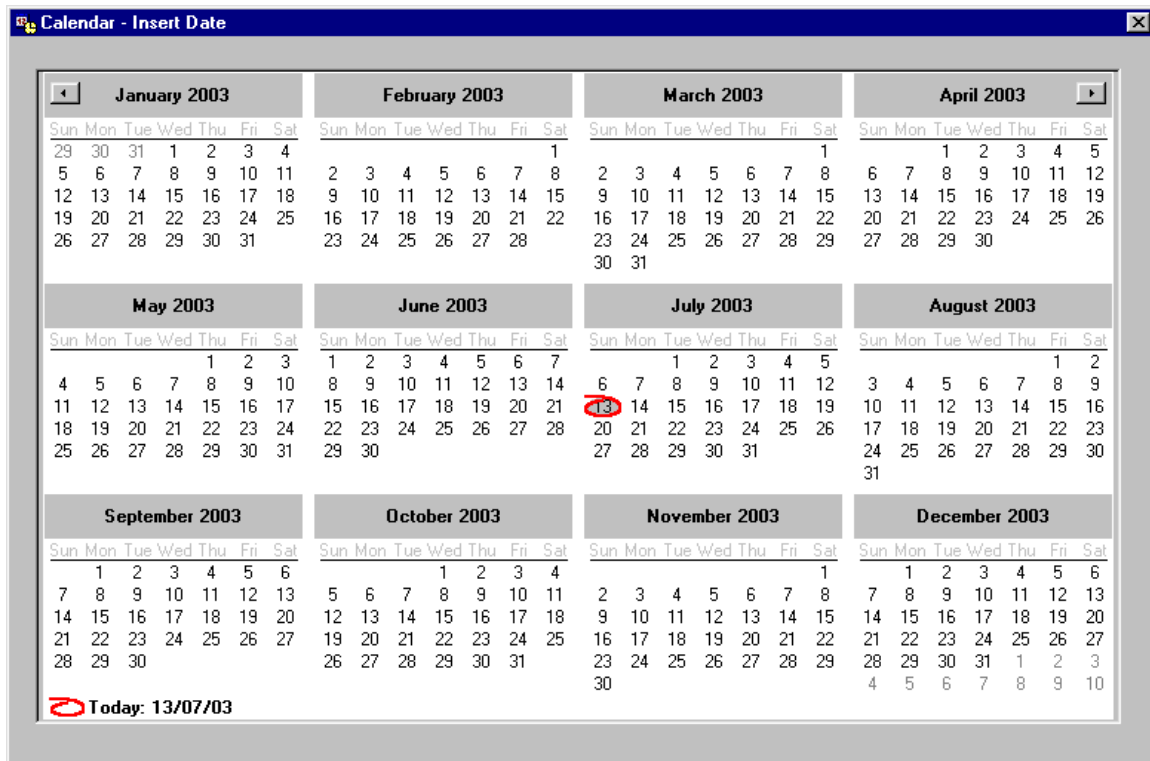
Inserting A Picture Into Your Document
1. Place the cursor where you would like the picture to be inserted.
2. Select **Picture** from the **Insert** menu.
3. Select the picture file you would like to insert from the list of files provided.

Inserting Another Document File Into Your Current Document
1. Place the cursor where you would like the picture to be inserted.
2. Select **File** from the **Insert** menu.
3. Select the file you would like to insert from the list of files provided.

Mail Merge Functions

Merge Fields
A merge field is a label instructing the system to insert a personalised detail at the position in the document at which the merge field is. For example, in the sample letter show at the beginning of this chapter there is a merge field for the address represented by «Guardian Address». This label is in fact an instruction to the system to replace «Guardian Address» with the contents of the "Guardian Address" column in the result of the query specified in the source SQL file that you supply when you are starting a merge session. To insert a merge field,
    1. Place the cursor where you would like the personalized detail to be inserted

2. Do one of the following
    a. Click the insert merge field icon and select the field you want from the list that will be provided from a window looking like the one shown below. You select by either double clicking the field in the list or clicking the field and then the **OK** button.
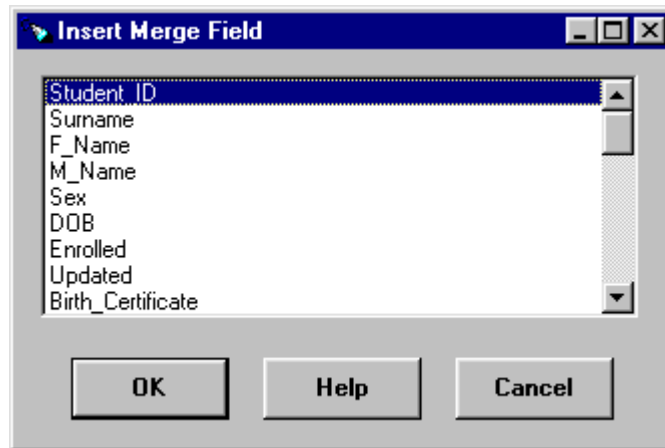
    b. Press the right button of your mouse. You will be presented with a pop up menu from which you can then select the field you want
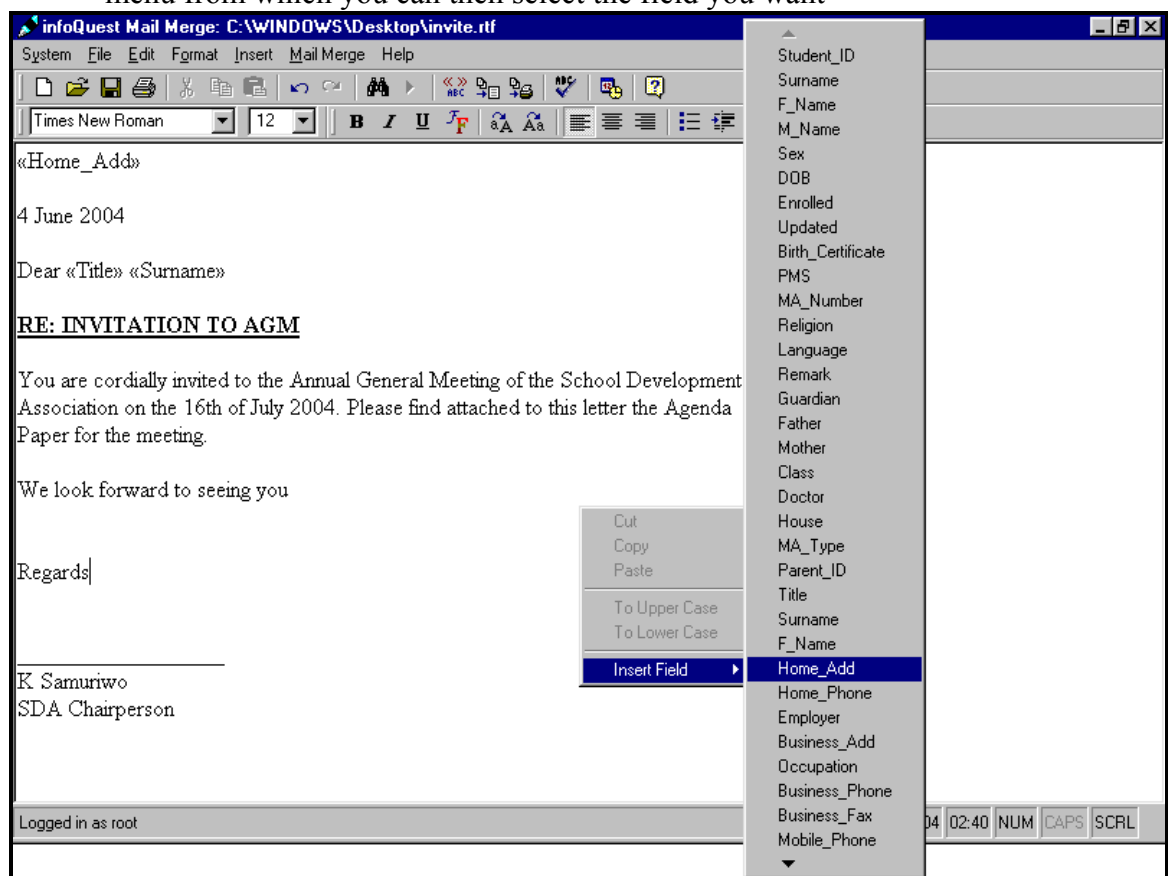
    c. Select **Insert Field** from the **Mail Merge** menu and follow the steps outlined under the first option.

Previewing The Personalised Letters

Before printing the personalised letters, you may want to have a look at how they will appear. To do this, choose the Preview Mail Merge 📑 icon from the toolbar. This will display a window with a list of all the letters with the personalised details inserted. A sample of the window is shown below
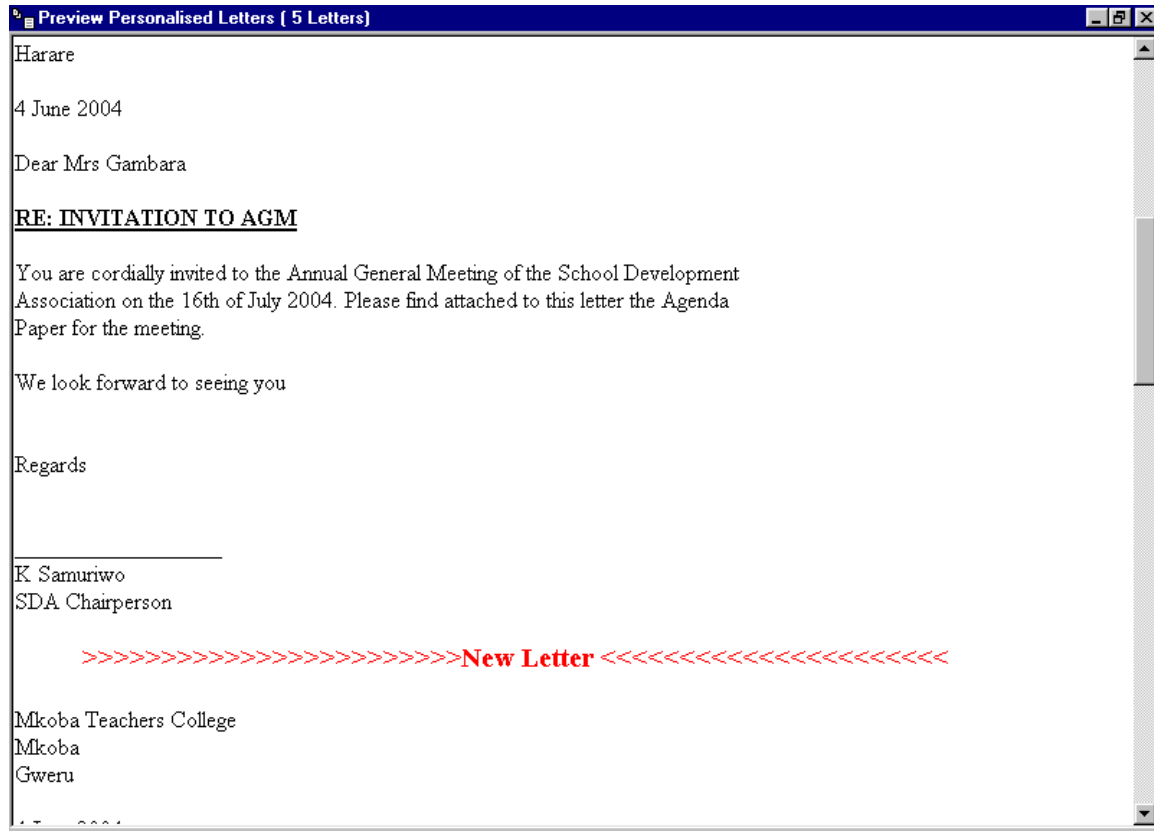
Producing Personalised Letters

To produce your personalised letters, simply choose the **Produce Personalised Letters** 📑 icon. You will be shown a progress bar as the letters are produced. If the source query is complex, the system might take a little while before displaying the progress bar. Please be patient.

Viewing The Query Source

If for some reason you want to see what the source query you just specified contain, choose **View Query Source** from the **Mail Merge** option. This maybe particularly useful when the system presents you with a list of field names that you did not expect.

Starting A New Session
A mail merge session is based on a single SQL file. If at any point you decide that you want to use a different SQL file, you must start a new Session. To do this, choose "**New Session**" from the Mail Merge menu.

APPENDIX A: TRADITIONAL LIBRARY SYSTEMS

The Browne System

The Browne System is a pocket based library circulation management system. In the system, each library patron is issued with pockets equal in number to the number of books that the patron may borrow at any one time. Sometimes, the pockets are colour coded with the patron being issued a number of pockets of a given colour equal in number to the number of books that he/she may borrow from a given section of the library at any one. For example, a patron maybe issued with two purple pockets that maybe used to borrow reference section books and five red pockets that maybe used to borrow books from other sections.

Each book in the library has a card that is kept in a pocket attached to the inside cover of the book. The card records the identification details of the book such as a title, author and accession number. When a loan is made, the card is removed from the book and placed in one of the patron's pockets. The pocket is then placed in a tray marked with the due date of the book. There is usually a tray for each due date applying to books that are not overdue and one for overdue books. The due date is also recorded on a "due date form" that is stuck on the inside cover of the book.

When a book is returned, the librarian retrieves the patron pocket from the "due date" tray and then returns it to the borrower. The book card is then replaced in the book that has been returned.

For an in-depth description of the Browne System please refer to the State Library of Queensland's article on the subject that is hosted at http://www.slq.qld.gov.au/pub/clsman/daytoday.htm

The Dewey Decimal Classification System

Description

The Dewey Decimal Classification System, more commonly known by the abbreviation, DDC, is a library classification system. A library classification system organizes the knowledge found in libraries by assigning a classification number to each book or piece of library material based upon its discipline. A discipline in the DDC is not the same as a subject; in fact, a subject may be found in many disciplines in the DDC system. For example, books about death may be found under many subjects including; biology, religious ethics, art, and sociology among others.

The Dewey Decimal Classification System is used widely in libraries because of its simple structure and because it easily incorporates new areas of knowledge; for example, materials about the Internet. Today 95% of school and public libraries, and 25% of academic libraries in the United States use the DDC system.

Overall View of the DDC

Basically, the DDC divides all knowledge found in the library into ten main classes. These are identified by the numbers 000-900. Generally, each class represents one main subject; for example, works in the 200's will always be about religion.
Here is the basic scheme of the DDC:

000			Generalities
100	Philosophy	&	Psychology
200			Religion
300		Social	Sciences
400			Language
500	Natural	Sciences	& mathematics
600	Technology	(Applied	sciences)
700		The	arts
800	Literature	&	rhetoric
900 Geography & history			

Each of these main classes can be subdivided into ten divisions. The divisions under the 600's include the following:

Technology (Applied Sciences)
610 Medical sciences, Medicine
620 Engineering & allied operations
630 Agriculture & related technologies
640 Home economics & family living
650 Management & auxiliary services
660 Chemical engineering
670 Manufacturing
680 Manufacture for specific uses
690 Buildings

Since the DDC system goes from the general to the specific, books from 600 to 609 will be general books about Technology.

Finally, each of these ten divisions is subdivided into sections. Medical sciences, a subdivision under the class Technology, appears on the DDC schedule like this:

610   Medical   sciences   Medicine
.1-.9   {Standard   subdivisions,   medical   personnel,   nursing}
611   Human   anatomy   (cell   biology),   histology   (tissue   biology)
612   Human   physiology
613   Promotion   of   health
614   Forensic   medicine,   incidence   of   disease,   public   preventive   medicine
615   Pharmacology   and   therapeutics
616   Diseases
617   Miscellaneous   branches   of   Medicine,   Surgery
618   Other   branches   of   medicine   Gynecology   and   obstetrics
619 Experimental medicine

Note that just as in the main divisions, the DDC goes from the general to the specific. Section 610, Medical sciences, is more general than the numbers 611 through 619, which cover aspects of the medical sciences.

How to Use the Dewey Decimal Classification System
1. A Dewey Decimal number will always have at least three digits. The first digit always indicates the main class. Therefore, any book having a number, which begins with a 7, will be a work about the arts. Likewise, any book with a number beginning with a 3 will be about the Social Sciences.
2. In the DDC, after the first three numbers, there will always be a decimal point. To find the book 811.54 C32 on a shelf, first find the first three numbers 811. The numbers after the decimal point, will be treated like decimals, meaning books with the number 811.5 will precede books with the call number 811.52, which will precede books with the call number 811.54. Once you have located the 811.54's (in a large library there will be many of these) then search alphabetically for the letters; e.g. 811.54 A22 will precede 811.54 C32.

3. In the DDC, many books will share the same call number. Therefore, if you searched the catalog for books about diseases, you will find many under the call number 616. If you went to that shelve in the library, you would find many books about various diseases there, grouped together by specific disease. For instance books about breast cancer will be under the call number 616.994. Likewise, books about heart disease will be found under 616.12. Remember that not all books about breast cancer or heart disease will be found here; others about diet or prevention or heart disease in children will be found under other call numbers in other disciplines. But books with the exact same call number should be about similar topics. Therefore, when you find a book that is right on your topic, look for books on the shelf near it with the same call number.

4. When several books have the same call number; for example, 575, you can distinguish between them by looking at their author numbers. The author number comes from a table known as the Cutter-Sanborn tables which gives a code of letters and numbers derived from the spelling of the author's last name, along with his or her first initial; or in some cases, the title of the book. The book and the author number give a unique address to the book so that you can find it among very similar numbers on the shelf.

5. Don't forget the Relative Index, which is the second main part of the Dewey Decimal Classification System. It is a good starting point for finding materials on various topics. It is alphabetical and very detailed and gives numbers for the same subject from the viewpoints of various disciplines. For example, look at the headings under Aerodynamics:

    Aerodynamics
    aeronautics 629.1323
    astronautics 629.4151
    meteorology 551.5153
    physics   533.62

How to Find A Book on the Shelf When Using the DDC

First find the main class where the book is shelved; for example, the 000's or the 600's. Then find the place on the shelf where the first three numbers match. If you are looking for 741.45 S32 try to find the 741's. Now search for the 741's which also include the first decimal number, 741.4; then scan the shelves until you find 741.45. Treat the numbers after the period as decimals, so 741.45 will go before 741.5. Now search the books with call numbers which match 741.45 S until you reach 741.45 S32.

Sometimes only a few books will almost match your number, sometimes there will be hundreds but if you follow these steps you will find the right book.

Frequently Asked Questions

1. What are the main differences between the DDC and the Library of Congress Classification Systems?
   - A. The DDC system was developed on a philosophical ordering of the world of knowledge. Subject specialists developed the LC Classification System to meet the demands of an actual, working, world-class library.
   - B. DDC uses a numerical based call number. Library of Congress Call Number:s begin with either one letter or two followed by at least four digits between the numbers 0000 and 9999.
   - C. The LC system can be expanded readily to catalog many books on the same subjects; whereas the Dewey Decimal Classification System works better in libraries which have less depth.

2. Is the DDC really a decimal system? If not, why is there a dot after the third number if more than three numbers are used? The dot in DDC is actually not a decimal. This dot has no real meaning, but it does make remembering long numbers much easier. The DDC is called a decimal system because the entire structure is a base-ten structure, which means that each number to the left of the point is ten times greater that the number to the right of the point. This allows numbers to show a hierarchical relationship. By that I mean that books about engineering in the 620 will be also under the subject Technology, which is a broader subject represented by all the books in the 600's.

3. What do the letters which are combined with numbers at the end of Dewey numbers represent? I thought DDC was a numerical system only. The letters are called Cutter numbers and they allow librarians to give each book an individual number. For example, many libraries will have many books on the same subject. To distinguish between these books, a Cutter number is added. The Cutter number is derived from the author's name. The first letter is the first initial of his last name, the numbers in between are a coded transcription of the spelling of his name, and the final letter, is the author's first initial. Books with no author entry will have a Cutter based on a title entry.

APPENDIX B: PROJECT APPROVAL DOCUMENTATION

Submissions to the Dept of Computing Sc

Application Letter

P. O Box 5526
Harare

19 October 2003

The Chairperson
Department of Computer Science
University of Zimbabwe
P. O. Box 5526
Harare

Dear Madam

RE: Application For Project Approval

I would like to apply for permission to undertake and submit the project detailed in the attached project appraisal in partial fulfilment of the requirements of the Bachelor of Business Studies and Computing Science degree program under course code CT360.

I sincerely hope that my application will be successful.

Yours Sincerely

Motsi Tinovimba G.
BBSCT III

Project Proposal

Name   : Tinovimba Gerald Motsi
Reg. Number   : R014626H
Program   : BBSCT III
Title   : Project Proposal

Problem Definition
It has been noted that although several schools have invested heavily in information technology, the focus of this investment has been limited to fields of academic study. In particular, although the technological infrastructure for an integrated administrative system exists in such schools, no efforts have been made to come up with the software solution capable of implementing such a system.

Proposed Solution
The solution to the above problem can only be determined to a reasonable level of accuracy after a comprehensive study of the information managed by high schools. It is envisioned that the solution will be generic with customisable components and thus applicable to all high schools. The system will be composed of a core that will include the components that will be common to all the schools. The services provided by the system will be enhanced by optional modules that will plug into the core. These will include, but not be limited to, a library, courses management and fees management subsystem. Each of the components will be client-server based and provide a platform independent web interface to allow authorized users to obtain information from or to provide input into the system. To ensure rapid application development, the fees management and library systems will rely heavily on code reuse although substantial modifications will be required to upgrade the available components into client-server applications.

Proposed Supervisor
Dr Hapanyengwi who has since agreed to supervise the project.

Compiled by

_____

Motsi Tinovimba G

Submissions to MOESC

Application to the Permanent Secretary

Department of Computing Science
University of Zimbabwe
P O Box MP167
Mount Pleasant
Harare

20 October 2003

The Regional Director
Harare Region
Ministry of Education, Sports and Culture
Chester House
Harare

Dear Sir

RE: RESEARCH CLEARANCE

I am a final year Bachelor of Business Studies and Computing Science (Honours) student at the University of Zimbabwe. As part of my approved program of study, I have been tasked with the carrying out of a research into the specification, design and implementation of an Information Management System for Zimbabwean high schools. I am therefore hereby applying for clearance to carry out the necessary background research.

The research, should you approve it, would focus on the establishment of the information managed by high schools with particular focus on

- The collection and storage of data during enrolment
- Registration of courses, examination scheduling and results publication.
- Collection, processing and storage of data used in the fees subsystems.
- Operational procedures for libraries and their associated data requirements.
- Event scheduling.
- Boarding facilities management with particular focus on the allocation of students to different halls of residents.

It is anticipated that the information would be solicited through interviews with the School head or any designate he/she may appoint at their earliest possible convenience. Authorisation from the Permanent Secretary to seek approval from the Regional Director has already been sought and obtained. Please refer to the attached letter of authorisation.

I sincerely hope that my application will be successful.

Yours faithfully

_____

Motsi Tinovimba G.

Project Proposal

Title
High School Information Management System

Terms of Reference
Authorised by the Department of Computing Science at the University of Zimbabwe for submission in partial fulfilment of the requirements of the Bachelor of Business Studies and Computing Science (Honours) degree programme under course code is CT360.

Project Supervisor
Dr G. T. Hapanyengwi
Department of Computing Science
University of Zimbabwe
P. O. Box MP167
Mount Pleasant
Harare

Objective of Research
To establish an understanding of the processes involved in the generation, storage and consumption of information in a typical Zimbabwean high school deep enough to facilitate the specification, design and implementation of an integrated and fully computerised Information Management System.

Methodology
The research will be by means of interviews with key personal as identified by the head of the participating school. The target group of schools will be drawn from Harare Region and will be chosen so as to be inclusive of the 3 main categories of high schools i.e.
- Government
- Private (Church Affiliated)
- Private (Not Church Affiliated)
The interviews would be held at the earliest possible convenience of the personnel involved after meeting any other formalities required by the institution before carrying out the research.

Interview Focus
The interviews will focus on the following key areas
- The collection and storage of data during enrolment
- Registration of courses, examination scheduling and results publication.
- Collection, processing and storage of data used in the fees subsystems.
- Operational procedures for libraries and their associated data requirements.
- Event scheduling.
- Boarding facilities management with particular focus on the allocation of students to different halls of residents.

Interview Questions

7. **Enrolment of Students**
   a. May you please describe the enrolment process with particular focus on
      i. Information collected.
      ii. How this information is processed and stored.
      iii. The additions and modifications made to the information stored in 1.a.ii over the course of the student's enrolment at the school.
      iv. Documentation and/or computer facilities used in facilitating 1.a.i & 1.a.ii.
   b. What problems have you experienced in managing the process detailed in 1.a
   c. If you were to specify the requirements for a computer system to aid the solving of the problems identified in 1.b, what would you include?
   d. Of the information stored by your schools with respect to each enrolled student, what would you want to be directly available to the student's guardian through a web interface?
   e. Of the information referred to in 1.d, what would you wish a parent to be able to modify directly through a web interface.

8. **Academic Registry**
   a. What courses do you offer?
   b. What constraints do you place on the courses that any given student can enrol for?
   c. What information do you store with regards to the courses on offer and the students enrolled for such courses?
   d. What is the assessment criterion for each group of courses?
   e. What aspects of the results of the assessment referred to in 2.d are made available to guardians? What documentation is used in facilitating this?
   f. What problems have you experienced in
      i. Timetable preparation
      ii. Assessment
      iii. Communication of assessment results
      iv. Storage of information
      v. Any other aspects of the registry
   g. How would you employ a computer system to aid the problems identified in 2.f

9. **Fees Management**
   a. Types of fees charged
      i. What fees do you charge?
      ii. To whom do these fees apply?
      iii. Who is responsible for which fees?
      iv. Who gives the directive to charge any given student a given fee?
      v. What documentation if any is used in (3.a.iii) above?
      vi. What documentation used in communicating the fees charged against any given student to that student?

b. Storage of information
    i. What specific transaction records do you keep?
    ii. What documentation is used in this storage?
    iii. For how long are these records kept?
    iv. What happens to the records when their period of use has expired?

c. Uses of information
    i. What specific uses do you put the above specified information?
    ii. Who are the consumers of this information and what specific aspects of it do they require?
    iii. What problems, if any, have you experienced in storing or accessing this information?
    iv. What opportunities for further utilization of the information would you want explored?

10. **Libraries**
    a. What library facilities does your institution provide?
    b. What classification for books have you adopted?
    c. What classification have you adopted for borrowers?
    d. May you please describe the circulation process?
    e. What restrictions are placed upon the operations that can be carried out by the personnel in the library?
    f. What links exist between the library and the other subsystems within the school?
    g. What problems have you experienced in managing your libraries?

11. **Event Scheduling**
    a. How do you schedule extra-curriculum activities?
    b. Who has access to the schedules referred to 5.a?
    c. What problems have you experienced in the scheduling of events?

12. **Boarding Facilities**
    a. What boarding facilities do you provide?
    b. May you please describe the process used in allocating students to the different halls of residence.
    c. What information is generated and consumed by the process detailed in 6.b?
    d. What documentation is used in 6.b?
    e. What problems have you experienced in managing the information generated by the accommodation system.

Publication of Findings
The research findings will form part of the report to be submitted as to the academic regulations of the University of Zimbabwe. It will therefore be left within the public domain of the Department of Computing Science. A copy will also be published at the author's website at www.geocities.com/tgmotsi. The website is within the international public domain and already contains other projects carried out by the researcher.

Researcher Contact Details
Name   : Motsi Tinovimba G.

Postal Address : Department of Computing Science
                  University of Zimbabwe
                  P O Box MP167
                  Mount Pleasant
                  Harare

E-mail   : tgmotsi@yahoo.com

Website   : www.geocities.com/tgmotsi

Telephone   : +263 11 754102

Compiled by

_____
Motsi Tinovimba G

Letter of Approval

Ref: C/426/3

all communications should be addressed to
"The Secretary for Education Sport and Culture"
Telephone: 734051/59 and 734071
Telegraphic address : "EDUCATION"
Fax: 794505

ZIMBABWE

Ministry of Education Sport and Cultu
P.O Box CY 121
Causeway
Zimbabwe

22 October 2003

Mr Tinovimba Motsi
University of Zimbabwe
Computer Science Department
P. O. Box MP 167
Mount Pleasant
Harare

PERMISSION TO CARRY OUT RESEARCH ON "THE SPECIFICATION, DESIGN AND IMPLEMENTATION OF AN INFORMATION MANAGEMENT SYSTEM FOR ZIMBABWEAN HIGH SCHOOLS: HARARE PROVINCE"

With reference to your application to carry out research on the above mentioned topic in the Ministry of Education's institutions, permission is hereby granted. You are, however, required to liaise with the Provincial Education Director of Harare Province, for clearance before carrying out your research.

You are also required to supply the Ministry of Education, Sport and Culture with a copy of your research that may contain information instrumental to the development of Education in Zimbabwe.

J. Chinamasa
for: **SECRETARY FOR EDUCATION, SPORT AND CULTURE**

MINISTRY OF EDUCATION
SPORT AND CULTURE
SCHOOLS DIVISION

2 3 OCT 2003

P.O. BOX CY 121, CAUSEWAY
ZIMBABWE

MINISTRY OF EDUCATION
SPORT AND CULTURE

2 3 OCT 2003

HARARE REGION
P.O. BOX CY 1343, CAUSEWAY

Figure 27: Approval from the Ministry of Education, Sport and Culture

291

Application to the Regional Director – Harare

Department of Computing Science
University of Zimbabwe
P O Box MP167
Mount Pleasant
Harare

20 October 2003

The Regional Director
Harare Region
Ministry of Education, Sports and Culture
Chester House
Harare

Dear Sir

RE: RESEARCH CLEARANCE

I am a final year Bachelor of Business Studies and Computing Science (Honours) student at the University of Zimbabwe. As part of my approved program of study, I have been tasked with the carrying out of a research into the specification, design and implementation of an Information Management System for Zimbabwean high schools. I am therefore hereby applying for clearance to carry out the necessary background research.

The research, should you approve it, would focus on the establishment of the information managed by high schools with particular focus on
- The collection and storage of data during enrolment
- Registration of courses, examination scheduling and results publication.
- Collection, processing and storage of data used in the fees subsystems.
- Operational procedures for libraries and their associated data requirements.
- Event scheduling.
- Boarding facilities management with particular focus on the allocation of students to different halls of residents.

It is anticipated that the information would be solicited through interviews with the School head or any designate he/she may appoint at their earliest possible convenience. Authorisation from the Permanent Secretary to seek approval from the Regional Director has already been sought and obtained. Please refer to the attached letter of authorisation.

I sincerely hope that my application will be successful.

Yours faithfully

_____

Motsi Tinovimba G.

APPENDIX C: MOUNT PLEASANT HIGH SCHOOL

Introduction

The fees charged by Mount Pleasant High School fall into two categories. The first is made up of so called "government fees" and the Bursar administers this group of fees. The second is made up of levy fees and the Levy Office administers it. The two offices are fully autonomous of each other.

The Bursar's Office

Four fee categories fall under the Bursar's office. These are
  a) The tuition fee
  b) The General Purpose fee
  c) Industrial fees.
  d) Examination fees

The tuition fee is gazetted by government and communicated to the school head by means of a circular from the Ministry Of Education, Sport And Culture. All students in former "Group A" schools pay the same tuition fee. Government also gazettes the General Purpose fee. The government gazettes a range and then invites school heads to select and apply for a specific figure within this range. Industrial fees are charged for all practical subjects with variations being allowed for across the various academic levels. Examination fees are also gazetted and accepted by the Bursar on behalf of the Zimbabwe Schools Examination Council.

All fees are payable in advance. Students are therefore invoiced upon registration and during the last week of each term if the student wishes to continue with his/her studies in the subsequent term. The invoices used enjoy statutory instrument status and must be obtained from the Government Printers. The invoice is the primary document used when paying fees into the school's "School Services Account" and banks are required to only accept payments accompanied by invoices and to note the invoice number when recording the deposit.

Upon payment, the paying agent is issued with two deposit slips one of which must be presented to the school as proof of payment. The school will, upon deposit slip receipt, issue an admission form from the "School Services Fund" receipt book that is also obtained from the Government Printers. Before issuing the admission form, the Bursar is required to ensure that the student actually paid the amount invoiced. If there is a shortfall, she is not allowed to issue the admission form and the bursar must refund any surplus. These refunds are usually in the form of cheques. No balances maybe carried forward or transferred to other accounts including those administered by the Levy Office. For those students who lose their deposit slips, an admission form maybe issued upon receipt of the school's bank statement if it contains a record of a deposit made against the invoice number issued to the student.

The Bursar has an iMac computer with an Epson bubble jet printer. The computer system is however not used in the processing of fees as the accounting system used also enjoys statutory instrument status and does not allow for computer-aided fees processing.
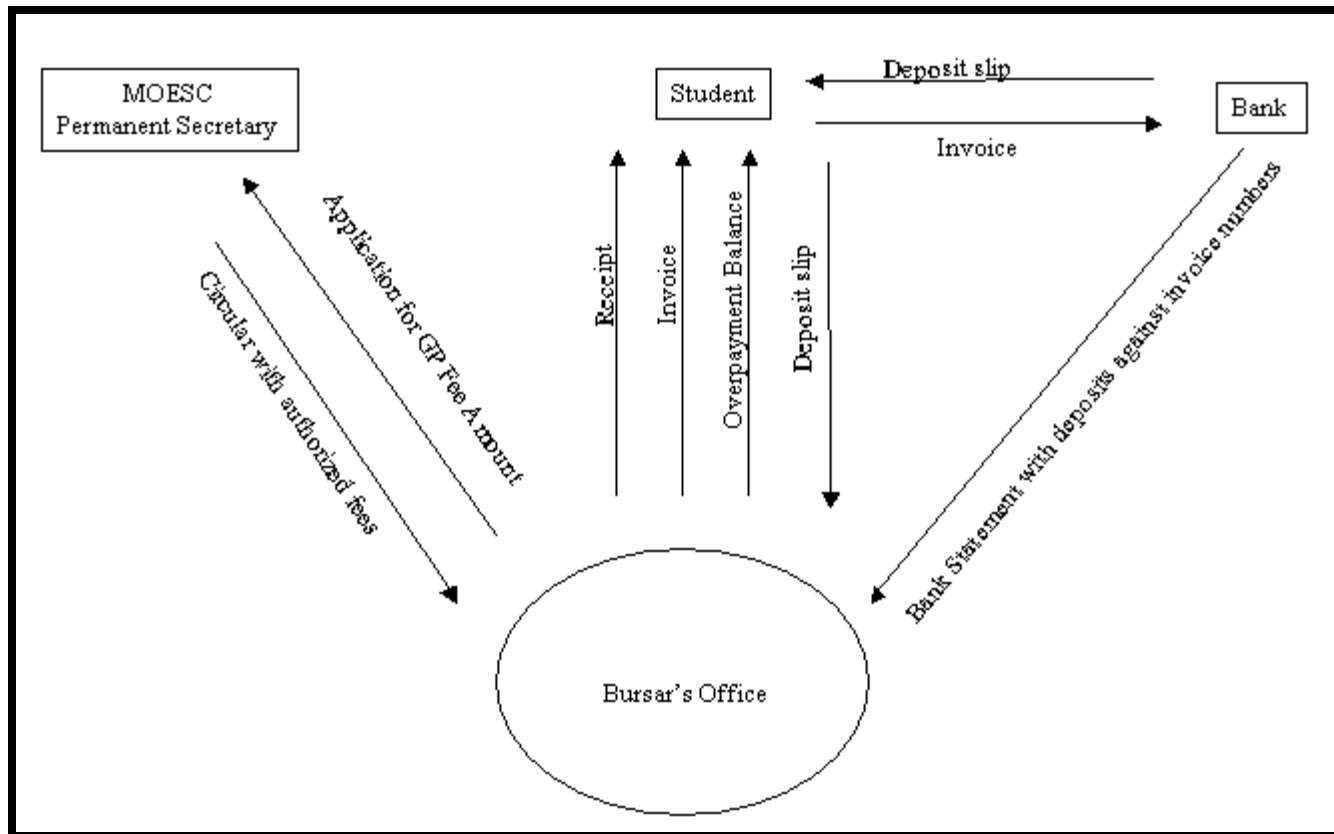
Figure 28: Mt Pleasant High School Bursar's Office Context

The Levy Office

The Levy Office handles the following fees
    a) Registration fee
    b) General levy
    c) Project fund
    d) Caution fee deposit
    e) School Bus Fund
    f) Building Fund
    g) Dinner dance levy
    h) Science levy
    i) Magazine levy
    j) Report book levy

The amounts charged under each fee vary with the academic year of the student. The following categories are currently in use

- Forms 1 and 2
- Forms 3 and 4
- Advanced Level Sciences
- Advanced Level Non-Sciences

The School Development Association, subject to ministerial approval, sets the above fees.

A student returning to the school for Advanced Level studies is required to reregister and thus pay a registration fee. The report book fee on the other hand is paid only on initial registration. The caution fee is also payable upon registration and must be maintained at the prescribed minimum level. A student, who incurs any financial liability, for example by loosing a textbook or damaging laboratory equipment, has his/her account debited. Upon clearance for deregistration, the balance on the account is refunded.

For the purpose of accounting, the Levy Office maintains a Total Debtors account with the individual student accounts being kept as memoranda accounts. It also maintains total fees accounts for each fee class. Entries in the Total Debtor accounts and Total fees accounts are based on figures extracted from these memoranda accounts. The individual student accounts are maintained using an Excel 7.0 spreadsheet. For audit purposes, a break down of the total debtors figure into credit balances for those who have prepaid and debit balances for those still owing is required.

Upon registration, a student is added to the corresponding worksheet in the spreadsheet for the given calendar year. His/her account is then automatically debited. Upon payment, the account is credited and a receipt issued. These transactions often occur one after the other. The invoicing of fees at the end of the term is however procedurally different. Before invoices are prepared, class lists are updated using the class attendance registers. In addition, the student's current "levy fees owing" balance also has to be calculated. This is necessitated by the fact that a student who has failed to pay the required levies cannot be denied education and may thus carry a negative balance until legal recourse remedies the situation. Also, the Levy Office does not refund those who pay more than what they will have been invoiced unless they specifically make a request for this to be

done. The prepared invoices are then used to make the payments through the School Development Association's bank account using the same procedure as that used when paying "government" fees.

The Levy Office has an IBM compatible personal computer and an HP LaserJet 5L printer. The system runs on the Microsoft Windows 95 platform. The principle officer in the Levy Office is quite proficient with regards to using the system as evidenced by the modifications he has made to the Excel 7.0 spreadsheet he has been using.

Statement of Preliminary Feasibility

After carrying out the above research, the project developer has ascertained that the statutory requirements for the Bursar's office make the computerisation of that office's fees processing system unfeasible. As a result, the project scope has been revised to cover the Levy Office only. In light of this, all references in the remainder of this document to the "Fees Processing System" will be with respect to the "Levy Fees Processing System".
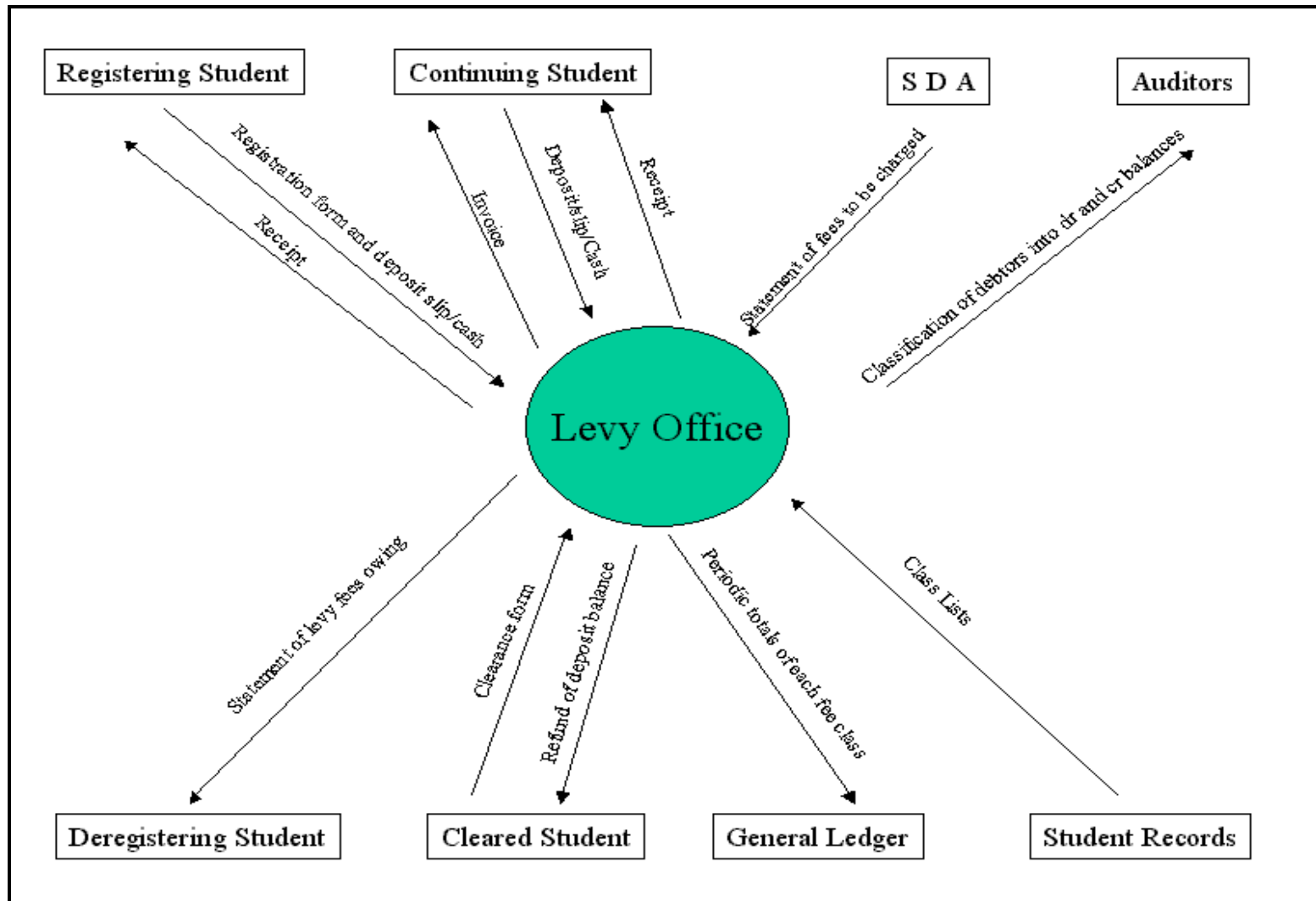
Figure 29: Mt Pleasant High School Levy Office Context

APPENDIX D:
GNU GENERAL PUBLIC LICENSE

Version Information

- ❖ Version 2, June 1991
- ❖ Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
- ❖ Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all. The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

  a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

  b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

  3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

  a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

  b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

  c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for non-commercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source

code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

  4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

  5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

  6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.You are not responsible for enforcing compliance by third parties to this License.

  7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

  8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

  9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

  10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

No Warranty

  11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE; THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

  12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.