7105 Computing Coursework

A Prototyping approach to O-Level Coursework

This example demonstrates how prototyping can be used together with a software package such as a database, to satisfy the requirements for Computing 7105 coursework. It uses *Lotus Approach* but the method is applicable to other databases. Prototyping may also be used with spreadsheets, web sites and programming projects.
The work is presented as though written by a student with the examiner's individual comments clearly outlined throughout. To save on web-page loading time scans of documents have not been reproduced and similar restrictions have been applied to screenshots. A full analysis and explanation of the marks is given at the end of each section.

Section A: Problem Analysis and Specification

My school library has recently been given a grant to update its book stock and introduce a CD-based, multimedia section. This should result in a library with about 3000 books and a network of five computers with access to several encyclopaedias stored on CDs. The library is run by 5th and 6th form pupils, supervised by one of the English teachers.
Until now, the library has been run as a manual system using a card index based on library tickets and with reminders being sent through the internal post when books become overdue.
The borrowing and returning system works fairly well, although tickets have sometimes been lost. The main problems lie with overdue books and with book reservations.
Overdue books are notified to borrowers on pre-printed slips but the librarians still have to fill in details by hand. This is time consuming and mistakes can be made.
The reservation system is even worse since the librarians have to fill in another slip and then do a manual search of all the borrowed tickets to find the right one and place the slip with the ticket. This can be very time consuming.

Examiner's comment:
Examples of the overdue and reserved slips are helpful here. These should be actual documents or scans of them, rather than word processed versions.

The pupil librarians asked whether someone from the O-Level Computing class could make a database for them. I agreed to attempt the task.
To find out exactly what was needed, I interviewed the senior, pupil librarian and spent some time in the library seeing how it worked.

Examiner's comment:
A transcript of the interview and details of observations should be included here.

As a result of my research, I have found that the librarians would like the database to do the following:-

- Record loans. They would like the system to record who has borrowed a book. The database should be able calculate the return date and also indicate if a borrower tries to take out more books than is allowed or tries to take out a book while they have another one that is overdue.
- Automatically produce letters to borrowers who have overdue books.
- Make the reservation system easier for librarians to use.
- Allow searches for books by title, author and subject.

Examiner's comment:
The first task has been specified in detail. This should have been done for each task in the list.

I noticed in my observation that pupils frequently ask the librarians where they can find various books. I therefore suggested that

- the database be made available on the library network so that pupils can make their own searches.

The librarians agreed that this would be a good idea and would save them some work.

Examiner's comment:
Users are not always aware of all the improvements that could result from using a computer. Here the candidate has noticed a likely possibility and has gained the approval of the user to implement it. Even if the suggestion had been rejected by the user it would have been a good idea to mention it in the report.

To make the library system work, I think that I need to break the job down into three main areas, loans, searching and mail merging. I think that reservations and pupil searches can be dealt with as part of the general search routines.
For loans, I will probably need to use calculated fields to work out due dates. I must also deal with the problem of books that are overdue and identify students who try to borrow more books than they are allowed. These could both be done by running searches automatically as each loan is processed, and displaying a message if the loan is not allowed.

Examiner's comment:
By considering the problem from a computing point of view, the candidate has identified some sub tasks and desired outcomes that did not occur to the user. Note that the decision whether or not to run a search, or a detailed explanation of how it would be run should be left to the design stage.

The project will be successful if the librarians can use the database to fulfil the requirements that they specified and also enable pupils to conduct their own book searches.

Examiner's comment:
Such a statement of success criteria is only acceptable if the requirements are well set out. If the candidate has any doubts it would be better to restate the requirements.

<u>Overall marking comments: Problem analysis and Specification</u>

Some relevant details such as the number of pupils, the number of books that a pupil can have at one time, the number of transactions per day are missing. Otherwise, the problem is clearly identified and broken down into sub problems. The candidate has clearly indicated how he or she sees the tasks and there is a full discussion of the problems posed by current methods. The desired outcomes have been stated and the evaluation criteria are acceptable.

Marks: 13/15

Section B: Design and planning of solutions

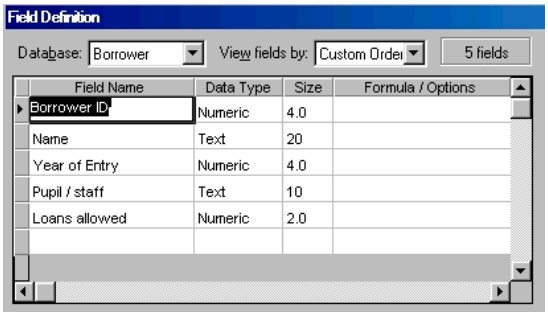
The library is to be equipped with 5 Pentium 4, 1200 MHz computers with 256 Mb RAM, 20Gb Hard drives, DVD drives and network cards. The monitors are 17" SVGA. The I.T. department is donating one of its older printers, an HP 840c, which will be attached to one of the machines. This machine will be the one that the librarians will work with. The printer will be shared so that all of the pupils using the system will be able to print if required.

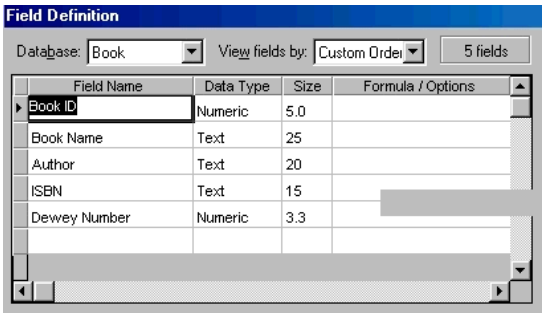
Examiner's comment:
In this project, the candidate has no choice of computers, printers, or operating system and so is unable to justify the choice. The candidate has however gained some credit by describing the equipment that he or she will be working with. There is no need to give a detailed description of hardware that is not relevant to the project. In this case, the make of the monitor and its refresh rate, details of the sound card, and graphics card do not really matter when running a database.

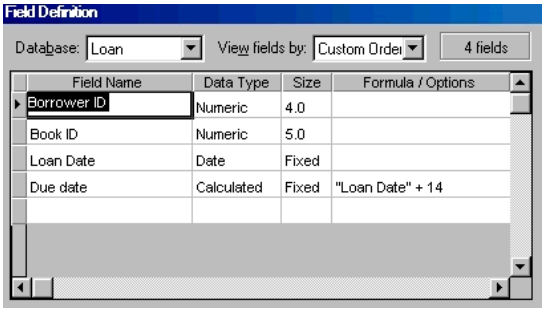
The software used will be Lotus Smart Suite 97, running under the Windows NT4 operating system. Both Lotus and NT4 are capable of running on P100 computers with 16 Mb RAM so the hardware supplied will be more than adequate for the proposed database.
The database could be written using Microsoft Access but the school is not licensed for this and it would therefore incur additional expense. Alternatively I could have written a database using Visual Basic, which the school does have. This might produce a database which conforms more exactly to the librarians' wishes but it would be more difficult to alter if requirements changed in the future. It would also take a lot longer to make.

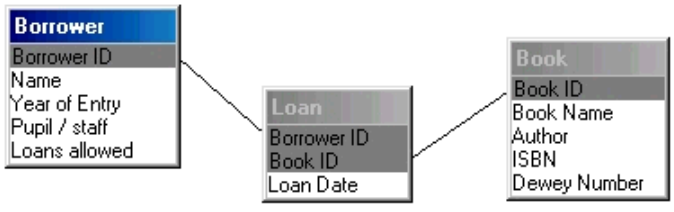
Examiner's comment:
The candidate has clearly justified the choice of software.
Note that the candidate has not copied large portions from software manuals or help files but has confined the justification to what concerns this project. The report does not read like an advertising brochure for the software.

The database will have three tables, Borrower, Book and Loan. The field details are shown below.
Links between the tables will be made using the Borrower and Book ID fields



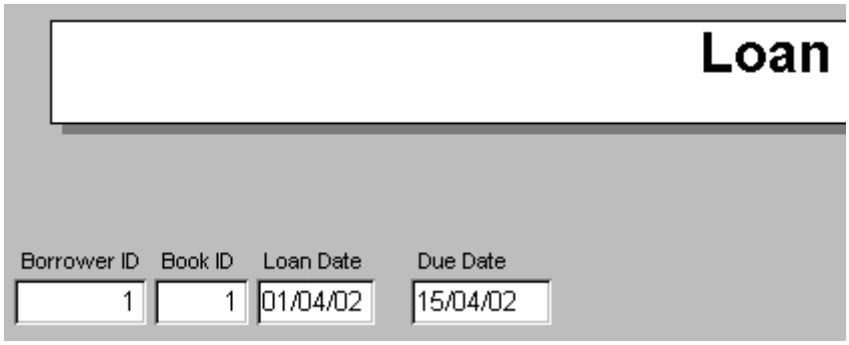


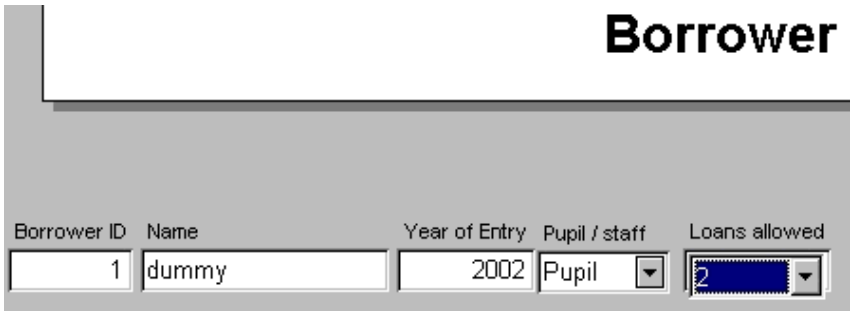


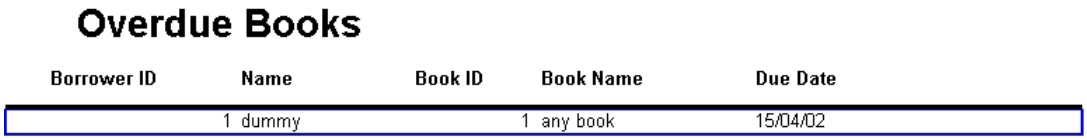


Examiner's comment:
This is a clear and appropriate plan. The tables and links are only a first attempt and might have to be altered later in the project as the prototype is developed.
The candidate should explain the reasons for the choice of fields and their types and sizes at this stage. It could be done by hand annotation of the pictures.

Having decided on the fields that I would need, I made basic layouts for the main screens and reports.





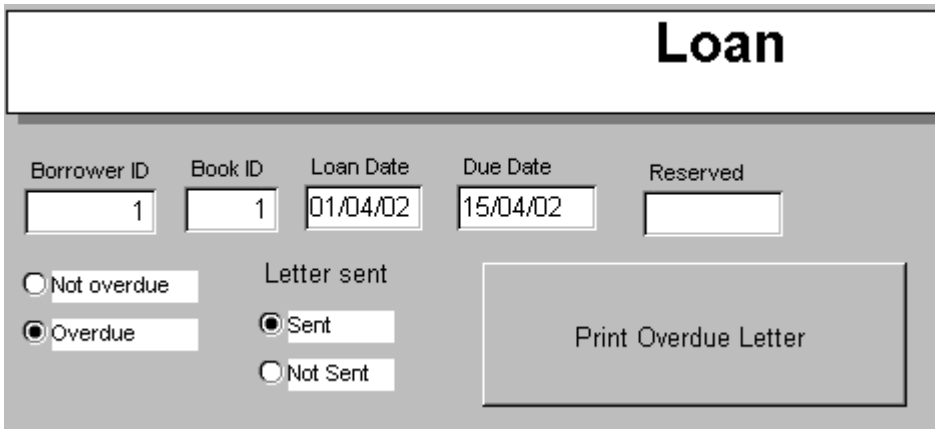


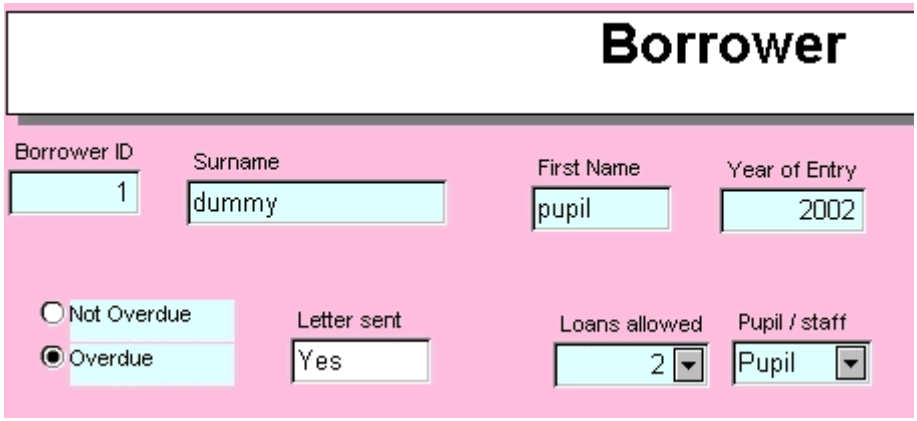
Note:
To save on web-page loading time the editor has allowed space for only three snapshots of parts of screens to be reproduced here.

Having made my initial design, I tested it from the user's point of view by showing it to the librarians to get their reactions to it. They were happy with the overall concept but made some requests for further features.
They asked that the name field be split into two, one field for surname and one field for first name.
They wanted a visual cue to appear on the Borrower form when a book becomes overdue.
They wanted a visual cue on the Loan form to show that a book is reserved.
They asked for the mail merge that prints overdue letters to have some manual control so that letters are not sent to someone every day that they have a book overdue.
Having talked the librarians through the workings of the database, I felt that it would be useful to have some on- screen instructions as well. I suggested that this would help when each year's new librarians are recruited. The librarians agreed that this would be a good idea and also asked that the screen layouts be made a little more colourful.

Examiner's comment:
Some aspects of implementation are discussed here as part of design but they will count towards the mark for the implementation section.

Splitting the Name field was a simple matter of adding a new field, renaming the original one and then repositioning things on the screen.
For the reserved cue, I opted for a simple text box, as there is no way to work out if a book has been reserved. I thought about adding Name and ID fields to show who wants the book but this would get difficult to use if the book is popular and there is a waiting list for it. Instead, the librarians will simply put a borrower ID into the text box and add others if a waiting list grows.
For the overdue cue, I made a calculated field on the Loan table, based on the due date and today's date. This displays as radio buttons on the Loan and Borrower forms.
To prevent letters being sent to all overdue borrowers every day, I made a field on the Loan table which indicates that a letter has been sent. I also placed a button on the Loan form which runs a macro. This macro prints the overdue letter and then sets the "Letter Sent" field to "Yes". This is duplicated on the Borrower form. When dealing with overdue books, the librarians can either search for all overdue books and then decide which letters to print, or they can search for all overdue books where the "Letter Sent" field is set to "No".
The amended forms are shown below.





Examiner's comment:
The candidate has not stated whether a further demonstration was arranged for the librarians at this stage. A demonstration might have resulted in further modifications to the prototype.

To show that the system works, I will have to test all of its components in a logical fashion. I must ensure that it will fulfil all of the user requirements and that the calculated fields and macros produce the correct results. To do this I will produce test data and a detailed test plan. I will also set the field validation values at this stage.

Test Plan.

Test No.	Test	Expected outcome	Comments
1	Borrower ID, insert valid ID 4 figs	Accepted	Validation test
2	Borrower ID, insert duplicate ID	Rejected	Validation test
3	Borrower ID, insert invalid ID 5 figs	Rejected	Validation test
11	Search for overdue books	finds 3 books	title5, title11, title15
12	Search for books due today	finds 2 books	title2, title6
13	Search for borrower name	finds 1 name	name1

Examiner's comment:
The candidate clearly knows how testing should be carried out but has not listed a complete set of test data to be used later in the project when the database has been built.

Note:
Only a sample of a larger list of tests has been reproduced here.

Testing should be split into 2 areas, (i) component tests as shown above, (ii) system tests to show that user requirements have been met.

Overall marking comments: Design and planning of solutions

The candidate has clearly presented and justified details of the hardware and software systems to be used and has mentioned suitable tools and techniques for building this database application. Although, to save on web-page loading time the complete designs for screens and reports have not been reproduced here, the actual screens and reports submitted by the candidate were well designed and were a sensible size. The candidate included some of the early drafts for these designs and explained why they had been modified in later versions. Some indication of the methods of validation and testing had been tabulated but no details of test data or of expected results were submitted. Test data should have include examples to show the normal working of the database. In this case it would require a number of books and borrowers and some loan and return transactions. Some of the loans should be overdue, others should be due today. It should also include examples that would fail the validation tests.

Marks: 20/25

Section C: Implementation of design

Examiner's comment:
As mentioned in an examiner's comment above, some aspects of implementation were included in Section B. The candidate has submitted nothing further under this section. The candidate could have included a user manual or an explanation for a user of how the solution works. Good marks could have been gained for a clear, comprehensive, self contained manual or explanation covering normal operations and simple errors. Examiners cannot award marks for work that has not been submitted.

Overall marking comments: Implementation of the design

The candidate has used appropriate tools and techniques and has understood how to use them. There is evidence to show that a solution has been achieved and there is discussion of some of the difficulties encountered and how they were solved. Additional marks could have been gained for details of how students attempting to borrow too many books are identified, and for details of how calculated fields and macros have been developed. In the following section the candidate reports that a method of identifying students who attempt to borrow too many books has been incorporated but sadly no evidence of this is presented and so no marks can be awarded.

Marks: 18/30

Section D: Testing of the Solution

Now that I have made my database, I need to test it by carrying out my test plan. This is tabulated below, with screen shots in appendix A which show the results of the tests.

Test No.	Test	Expected outcome	Actual outcome
1	Borrower ID, insert valid ID 4 figs	Accepted	Accepted fig1
2	Borrower ID, insert duplicate ID	Rejected	Rejected fig2
3	Borrower ID, insert invalid ID 5 figs	Rejected	Rejected fig3

Examiner's comments:
The candidate has presented real evidence; not just word processed results. The use of an Appendix is a good strategy because it prevents the test section of the documentation becoming too bulky. These tests seem to have been passed successfully but if tests do fail, the errors should be documented and an explanation of how they were overcome or of how they might be overcome should be included. Only three of the tests have been reproduced here.

APPENDIX A

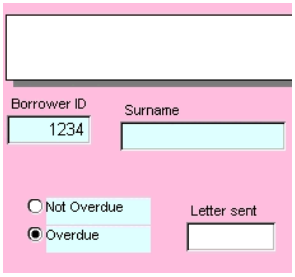


Fig 1.
4 figure borrower ID accepted.
Test 1

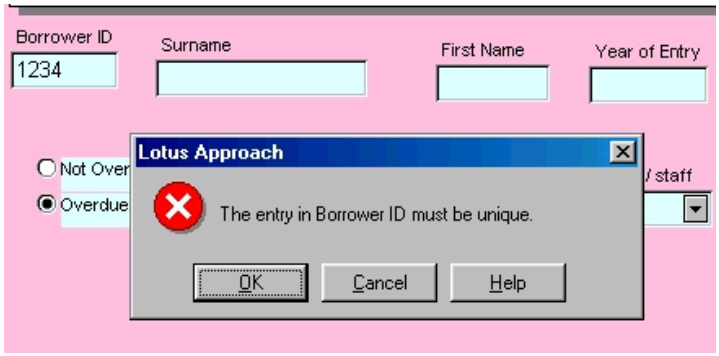


Fig 2.
Duplicate borrower ID rejected.
Test 2

Examiner's comment:
The candidate has displayed the message against its relevant background. A message by itself does not show that it was triggered by the correct conditions.

As a result of my tests I found that where fields occur in more than one table, the validation rules must be set identically for each. I also discovered a problem with the maximum number of books allowed per borrower. This appears on the borrower form but there is no check to see if the number is being exceeded, except by doing a search of Loans.
I solved this problem by writing a macro which operates from a button on the Borrower form. The macro runs a search for that ID in the Loans table and displays the current loans for that borrower.

Examiner's comment:
Credit will be given for identifying this problem, greater credit would be due if there was evidence to show that the solution had been incorporated.

Once I had fixed the problems and was confident that everything was working, I asked the senior, pupil librarian to try it out. After a short tuition period, he was able to carry out all of the operations that he had requested in the original specification. He wrote a report on this which is included in Appendix A.

Examiner's comment:
This is a good strategy. User testing is important. If comments are unfavourable, the candidate can discuss them in the evaluation section and suggest possible developments. The comments may be typed by the candidate and signed by the person evaluating the work.

<u>Overall marking comments: Testing of the solution</u>

The candidate clearly understands the principles of testing and has presented evidence to show that testing has been done thoroughly. Appropriate amendments have been mentioned.

Marks: 6/10

Section E: Evaluation of the solution

Software
The chosen software is user friendly with good help files and wizards which made development of the more complex parts of the project relatively simple. All the page formatting is WYSIWYG and there is a large selection of tools for designing the appearance of screens and reports.
Mail merge to make the overdue letters was fairly simple to set up, using fields from Approach to create a Form Letter.
This was done within the database but there are only a few layout choices and it produces a fairly plain looking letter. It could also be done in conjunction with Word Pro, the Lotus word processor, if the librarians want to make something more elaborate in future.
Data entry was fairly simple as the tab order can be customised for each form to allow fields to be completed in a logical order rather than creation or display order.
Man machine interface.
My end user was happy with the interface and found no real problems in using it. He did comment that some buttons to initiate common operations such as searches would make things a little easier and these could easily be incorporated into the first upgrade. See Appendix A page 8 for the user appraisal.

Examiner's comment:
This is the right approach; the candidate has included both good and bad points, strengths and weaknesses, and has kept the discussion relevant to the project. Where evidence exists elsewhere in the report, the candidate has given page and figure references to it. Missing evidence would probably indicate an incomplete report.

Effectiveness of solution
My original user requirements and success criteria were as follows:-
1. Record loans. They would like the system to record who has borrowed a book. The database should be able calculate the return date and also indicate if a borrower tries to take out more books than they are allowed or tries to take out a book while they have one overdue.
Loans are recorded and referenced to a borrower using the Loans table. The due date is calculated from the loan date. Number of books borrowed by each borrower is found by a macro. Overdue books are found by a comparison between the due date and the system date and the result is displayed on the Loan and Borrower forms.
2. Automatically produce overdue letters.

Overall marking comments: Evaluation of the solution

Not all the original user requirements have been satisfied and the candidate has not been entirely honest in claiming that they have.
Evidence is linked to stated success criteria, weaknesses and possible solutions are considered. The effectiveness of the software and the quality of the man-machine interface are fully discussed.

Marks: 16/20

Total mark: 73/100