I.T. Project : Dental Surgery Filing System

I.T PROJECT 2001-2002

Lewis Marshall 11R

Identification

Identification of Problem

Problem 1 - I have noticed that at my dental surgery, they use a manual filing system to keep record of all the patients' records and when a record is needed by the dentist, the receptionist has to sort through the records in order to find it. I feel that they would benefit from having a computer version running to help save both time and effort for the receptionists.

Problem 2 - When a patient wishes to book an appointment, the receptionist must sort through the booked appointments, which are written in a book, to check whether a certain time or date is available. This problem could also be resolved using computers.

Problem 3 - The surgery types up individual letters to be sent to different patients to inform them of different events and remind them about upcoming appointments. This task would be much easier using certain software on the computer.

Benefits of a Computerised System

It would be very beneficial for the surgery to use a computer-based system for a number of reasons. Searches for patients can be carried out much faster. Communication between each dentist is much better as they can each share dentistry records etc. over a network rather than having a receptionist to go back and forth delivering them, and they will also be able to alter and add new records with not much difficulty and from anywhere in the world considering their network will allow this. The chance of records being lost or damaged by receptionists or dentists is lower unless they crash the system, and even then if the data has been backed up it can be easily restored. Access to records can be restricted on different levels so that very confidential records cannot be accessed by just anyone. Useful queries can be carried out on computer much more quickly and efficiently than if done manually. Data can be printed using a variety of fonts, colours, and sizes to help focus the dentist's attention on the most important data. In addition, images can be combined with textual data to create a more complete "picture" of the patient's condition. Double bookings are a thing of the past as computers can quickly sort through the data to check whether a certain time and date has been allocated to someone else. Software can be used on the computer so that letters can be typed up and names of patients can be entered from the database automatically, rather than typing each letter up manually to each patient. Complex calculations can be done much quicker on the computer using formulae.

Problem 1

Problem Outline

As stated before, my local dentist, Ayres Road Dental Surgery, uses a manual filing system. I have decided that it would be very beneficial if they used a computerised system to handle the patient's records. In order for me to create this however, I have designed a few end user requirements as guidelines to make sure I have an idea of what is required of this system.

End User Requirements

- The system must be simple for non-technical staff to use and contain a variety of different colours and styles to make important things stand out and also be colour blind friendly.
- The system must consist of a table containing relevant information about each patient (e.g. name, address etc.)
- The system must be re-usable so new patients can be added to the system.
- A simple data capture form must be designed for entering data into the system that is easy for both technical and non-technical users to understand.
- A simple form to navigate the system must be designed which is also easy for non-technical staff.

Information To Output From The System

The information to output is an A4 sheet containing a list of all the current patients in the database, with the surnames in alphabetical order, including their name, sex, date of birth, postcode and date of last visit. A list of all the patients who have not been to see the dentist within the past six months must also be produced on a separate A4 sheet, including their name, sex, postcode and date of last visit. This list must be in chronological order (referring to the date of last visit).

Data To Input Into The System

The main input into this system are all the patients' records which are already recorded on paper within the surgery. The following are the different factors which are to be stored about each patient:
- Surname
- Other Names
- Sex
- Date Of Birth
- Address
- Postcode
- Telephone Number
- Condition of Teeth
- Date of Last Visit

Each patient will also be given a unique ID number so that patients can be found easily and to avoid duplicate records. The surgery also commented on how useful a picture of the patient would be if included with their record. I will take this into consideration as one of my inputs but I am not sure given the resources I have access to whether or not I can do this.
To save time for myself, I will only enter a sample of the patients in the dentist's file, just enough to be able to test my system works efficiently without errors.

Performance Criteria

Some performance criteria were mentioned in the end user requirements earlier but I feel there are a few more 'guidelines' for me to follow to ensure a good system.
- As stated, the system must be re-usable but also the system must prevent duplicate records, so that if someone joined the surgery a few years ago and then forgets, their details cannot be entered again as if they were a new patient.
- Each record must be unique so that queries can be easily carried out.

Software Packages

Before I begin to design my system, I must decide on which software package will be most suitable and which I will be using. I have considered using four different types of software to implement this task:

- The first of these is a desktop publishing package such as Microsoft Publisher. This software package would be beneficial as it would allow me to have well designed, colourful forms but has not got the capabilities in recording data and using that data to do different things.
- The next software package I have considered using is a word processing package like Microsoft Word. Similar to a desktop publishing package, it allows me to combine text and graphics on the page to make it look decorative but again fails when coming to the practical side of the problem. Even though tables can be produce to store data, queries cannot be done on the data and forms cannot be produced to link the data together
- The third software package is a spreadsheet, such as Microsoft Excel. Spreadsheets are very capable of holding and sorting through data, much better than word processing and desktop publishing. However, there are some downsides. Spreadsheets are not very capable at producing forms for entering data or for linking different parts of the system together, which is required by the end user.
- The final piece of software considered is a database. From reading through the identification and the end user requirements, it is evident that a database package such as Microsoft Access is very suitable for implementing the system. Tables can be produced to store data on each patient, using a number of different fields to separate each aspect of the patients record. Forms can be designed for entering and navigating the system, and graphics can be used to make it look more appealing to the user. New records can be entered into a database at any time using the forms created so is easily re-usable. Reports can be produced using queries.

Software Comparison

I have chosen to use a database to implement this system for the reasons mentioned above. In this case, I considered using two different database packages. The first is Microsoft Works Database. This is an example of a simple database package. It allows me to organise data into a table and use forms to enter records. It also allows the user to create reports which may come in useful but it does not support macros. Macros, as I will describe later, will be very useful in making a navigating form. I also don't have much experience using Microsoft Works Database.

The second database package I have considered using is Microsoft Access. This package is very suitable for this task. Like Microsoft Works Database, it allows me to organise data in a table, create forms for entering data and produce reports. The advantage of Access is that it supports macros. This is a very important factor, as it will help me design my database so that it is simple for non-technical staff to use, required by the end user. I am very used to using Access as I have been using it for about 5 years now so I am very familiar with all the toolbars and different aspects of it. Access is also good for me to use as it is part of an integrated package, Microsoft Office, meaning I can use the data stored in the database with other software packages (e.g. mail merge with Microsoft Word).

DESIGN

Now I have decided which software I will be using to implement the system, I can begin to design it. To start with, I will create a new database and save it under a suitable name (e.g. Dental Surgery). Throughout the implementation of this task, I will save it frequently on a floppy disk so that a crash in the school system means I will not lose my project.

Table

I will create a new table, again saving it with a suitable name. In design view, I will create each field that the user will be entering data into. The following are the different fields (in bold) I will be creating and their formats (i.e. what kind of data will be entered into them)

Record Number – This will be an Auto Number, meaning that a number will be automatically inputted by the system each time a new record is produced. It will also be a primary key field so that each record has a unique record number. The number will be set to increment by 1 each time by setting 'New Values' to 'Increment'.

Surname – This will be in text format and the field size will be set to 20 characters, as surnames tend to be less than 20 characters so will avoid erroneous data and to also save disk space.

Other Names – This will also be in text format with the field size set to 20 characters to save disk space. If middle names do not fit into the field then initials can be used.

Sex – This will be a lookup column consisting of two options, Male or Female. The field size will be set to 6 characters to save disk space and the 'Required' section will be changed to 'Yes' so that this field cannot be left blank, as it is important for the dentist to know the sex of the patient.

Date Of Birth – This will be of Date format in the form of a short date (e.g. 19/07/01)

Street, Area & City – These 3 fields will be text format, with 'Street' having a field size of 20 and 'Area' & 'City' having a field size of 15. The default values 'Old Trafford' and 'Manchester' will be used for the fields Area and City respectively, as most of the patients will live close to the surgery, which saves time for the user as they won't have to keep typing the same words repeatedly.

Postcode – This field will be text format of size 8 characters as postcodes consist of numbers and letters and are a maximum of 8 characters. The default value 'M16-' will be used as most patients will live in this area and so have this at the start of their postcode.

Tel. Number – This will be a text field as I think phone numbers are more easily interpreted when containing a space or hyphen between the area code and the rest of the number, which are defined as text and not number. The default value '0161' will be used as this is the area code for Manchester and a field size of 12 to minimize use of disk space.

Condition of Teeth – This will be a text field and the field size will be 30 characters so to contain information of the patient's teeth. If this field is left empty in the database, it can be assumed that they have perfect teeth.

Date Of Last Visit – This will be in date format as dates are going to be entered into this field. The field will also be set to 'Short Date' under 'Format' at the bottom (e.g. 19/07/02)

<u>Entry Form</u>

As the dental surgery will have quite a lot of records, I have decided to use an entry form to make it easier for the user to enter data into the database. Rather than having to scroll up and down all the time, the user can use arrows on the screen to go back and forward, or to the first or last record. This form will be created so that the user can see where to enter the correct data into the database. It will include all the fields that are in the Records table. Each field name will have an appropriately sized box next to it for entering the data. This is my standard entry form created using the form wizard (shown at the top of the next page).

On the standard form above, it is not clear what the form is relating to but if I include a header on the form, which acts like a title it will be much clearer to the user what form they are using. I will now think about different colours and patterns I could use to make the form look more appealing to the user and be a bit easier to get around. The following are three different designs I have considered:

Entry Form

Record Number	1		Street	20 Beever Street
Surname	Thomas		Area	Old Trafford
Other Names	Julie		City	Manchester
Sex:	Female		Postcode	M16-9DP
Date Of Birth:	26/10/63		Telephone Number	0161-8777124
Date Of Last Visit	19/06/01			
Condition Of Teeth				

Record: |◄ ◄ 1 ► ►| ►* of 19

Entry Form

Record Number	1	Address:	Street	20 Beever Street
Surname	Thomas		Area	Old Trafford
Other Names	Julie		City	Manchester
Sex:	Female		Postcode	M16-9DP
Date Of Birth:	26/10/63		Telephone Number	0161-8777124
Date Of Last Visit	19/06/01			
Condition Of Teeth				

Record: |◄ ◄ 1 ► ►| ►* of 19

I have decided that I will use the design I created first (the first of the three designs). I decided this as I think it fits the user requirements better than the other two. The text is easy to read and the colours are easy on the eye. The only problem I could spot with this design is some of the fields, like the record number field, have bigger entry boxes than the actual text which will be entered into them so is a waste of space and can also be misleading to the user as the may think a lot of text must be entered. The following is therefore my final design for the entry form:

Entry Form

Record Number	1
Surname	Thomas
Other Names	Julie
Sex:	Female
Date Of Birth:	26/10/63
Date Of Last Visit	19/06/01
Condition Of Teeth	

|	
Street	20 Beever Street
Area	Old Trafford
City	Manchester
Postcode	M16-9DP
Telephone Number	0161-8777124

Record: |◄ ◄ 1 ► ►I ►* of 19

Navigating Form

This is a very important aspect of my system. This is the starting screen for the user to find their way to different parts of the database (e.g. display the Records table, display the entry form etc.). This is where macros come into use. A macro is a sequence of one or more instructions that is executed when some event occurs. I am going to use macros in conjunction with command buttons on a customised form so that when the user clicks on a button or caption, a certain event or group of events occur. I will also design a macro that will load up the starting screen automatically when the database is opened to save time and effort.

To start with, I will consider the layout of the starting screen. I will enter a large, appropriate title in the form header so that the user knows what is going on. I will then need to create the macros that I will be using to carry out different tasks. The first macro I am going to create is one that will open the records table and also maximise it. The following is a screenshot showing how I will do this:
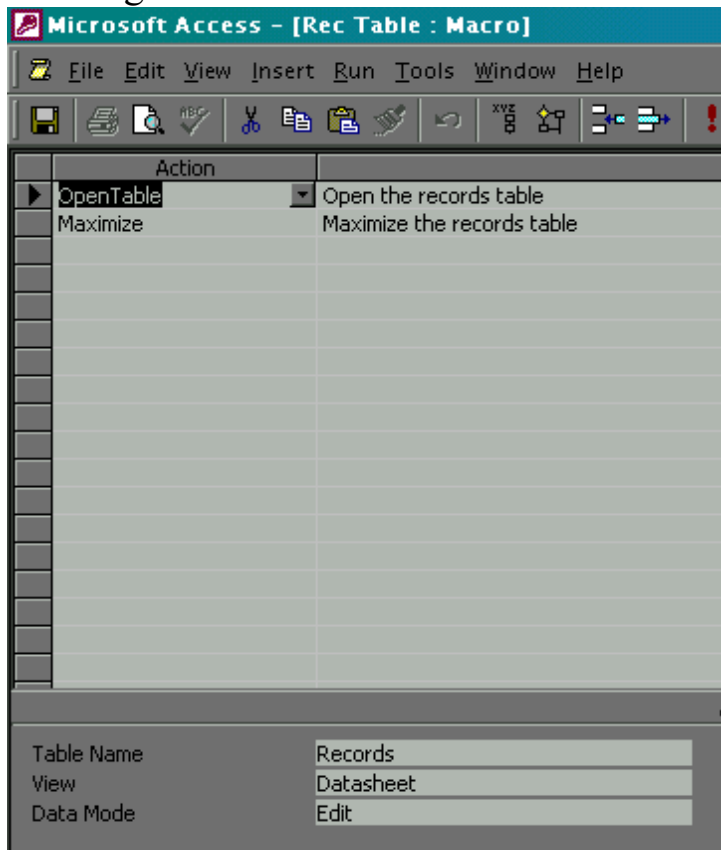
The next macro I will create is one to open the entry form and maximise it. This is done in much the same way as the first one I created but instead of choosing 'OpenTable', I will use 'OpenForm' and put the form name at the bottom. The next macro I will need to create is one to exit the database. This is a bit different to the two created already so I will again use a screenshot to show how I will create it:
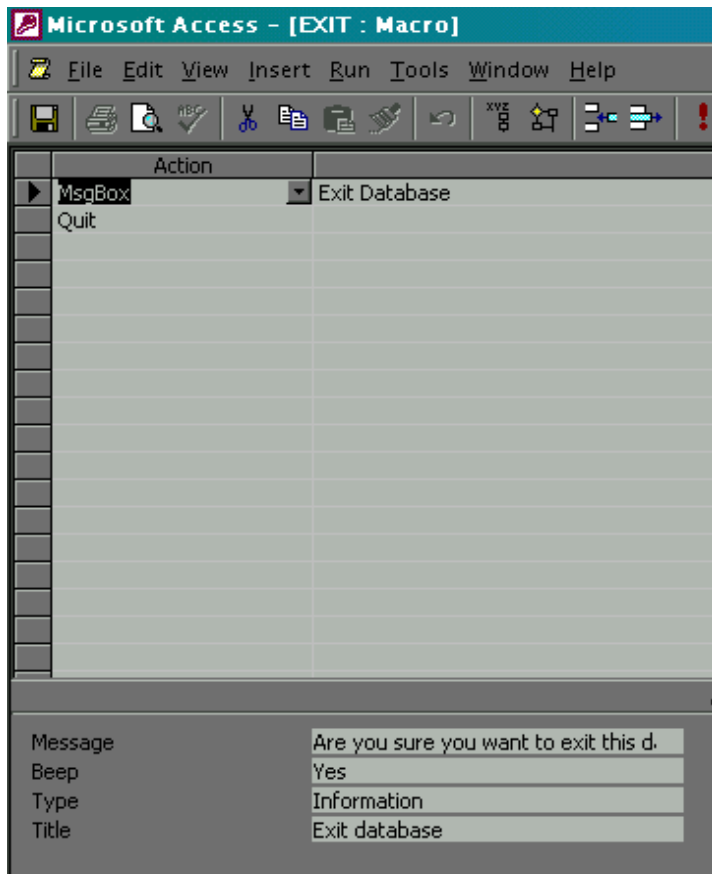
Now I have created all three macros and saved them with appropriate names, I can now add some command buttons to the navigation form I began to make earlier. These are made by clicking on the command button in the toolbox and drawing three boxes on the form. Then I will go to the properties of each button so that I can customise them. On each button, under 'On Click', I will choose one of the three macros and close the properties box. I will then add necessary labels next to each of the boxes containing text telling the user what the button will do once clicked.

Now I have the basis of the form, I can think about the layout of the form. I think I will keep the database consistent by using the same design as I did for the entry form, so here is my final design for the navigating form:

It is notable that I have left a space in the middle of the form. This is because I think I will be able to add more buttons to the form once I have created my other systems so that they are connected.

When the user comes to use the system, I want this screen to appear automatically on opening the database. This can be done again using macros. I will design another macro with its functions being 'OpenForm' - open the starting screen, and 'Maximize' – to maximize the starting screen. To make this macro run automatically, I will name it AutoExec, which stands for 'automatic execution'.