Summary

The main aim of this project was to implement a client and sales record management system for a medium-sized accountancy firm named P & Co Chartered Accountant, based in England. This is achieved by developing a PC-based database using Microsoft Access 2000. Below is a list of revised minimum requirements for the project:

1. To replace the paper-based system in P & Co Chartered Accountant to a computerised system.
2. To gather user requirements from the administrator and accountant in P & Co; in short' to obtain a clear view of the problem on the existing paper-based system.
3. To meet the user requirements and to recommend a new system by introducing database to store client and sales records.
4. To design and implement a database holding information about P & Co customers and their account details.
5. To evaluate the system.

My personal objectives for this project are:

- To review much of the work done during my studies at University.
- To gain experience of managing project, especially to apply a software development methodology to a 'real-life' development situation.
- To gain a practical understanding of how to design and implement a database and its interface.

This report will document the methodology used for this project as well as the design, implementation, testing and evaluation stages of the database. It will then be followed by a conclusion of how this project turned out, and possible recommendations for the future.

Archievements

I have met all the above requirements with additional features implemented into the database, the details of which may be found in this report. The database is now working, which allows the administrators and accountants in P & Co to store information in the database in a timely and suitable manner. The end users are very satisfied with my finished working system.

Acknowledgements

I would like to thank the following people for their help and support in the completion of this project:

Haiko Muller – my project supervisor for all his advice and encouragement in guiding me throughout this project.

Stuart Roberts – my assessor for providing me valuable suggestions after the mid-project report and during the progress meeting.

Jonathan Pitayanukul – the owner of P & Co Chartered Accountant who has been always ready to answer questions throughout the development of the database.

Owen Johnson – for his advice on software development methodology.

Mark Chan – for his technical advice in the database design and implementation.

Jacky Chan and Amy Williams – for their love, support and understanding throughout my project.

TABLE OF CONTENTS

		Page
SUMMARY		**i**
ACKNOWLEDGEMENT		**ii**
CHAPTER 1 INTRODUCTION		
1.1	Introduction	**1**
1.2	Minimum Requirements	1
1.3	Understanding the Business Background	1
1.4	Problems	2
1.5	Feasibility Study	3
1.6	Scope of the Project	4
CHAPTER 2 BACKGROUND RESEARCH		
2.1	Introduction	**5**
2.2	Approach and Project Plan	5
2.2.1	Methodology	5
2.2.2	Schedule and Project Management	6
CHAPTER 3 REQUIREMENTS AND ANALYSIS		
3.1	Introduction	**8**
3.2	Current System Procedures	8
3.3	Flow Diagram	10
3.4	Requirement Analysis	10
3.4.1	Functional Requirements	11
3.4.2	Non-functional Requirements	12
3.5	Selection of Implementation Tools	12
CHAPTER 4 DATABASE DESIGN		
4.1	Introduction	**14**
4.2	Entity Relationship Modelling	14

4.2.1 Mapping the E-R Model into Relational Tables 16
4.3 Integrity Constraints 18
4.3.1 Entity Integrity 18
4.3.2 Referential Integrity 18
4.4 Normalisation 18
4.5 User Interface Design 20
4.5.1 User Interface Architecture 20
4.5.2 Forms Design 21

CHAPTER 5 IMPLEMENTATION

5.1 Introduction **23**
5.2 Microsoft Access 23
5.3 Table Implementation 23
5.4 User Interface Implementation 24

CHAPTER 6 TESTING, INSTALLATION AND EVALUATION

6.1 Introduction **29**
6.2 System Testing 29
6.2.1 Function Testing 29
6.2.2 Validation Testing 29
6.3 User Acceptance Testing 30
6.4 Installation of the Software and Maintenance 30
6.5 Users Training 30
6.6 Evaluation 31

CHAPTER 7 EVALUATION

7.1 Introduction **33**
7.2 Project Evaluation 33
7.3 Further Enhancements 34
7.4 Project Conclusion 35

REFERENCES **36**

APPENDICES **39**

Appendix A – Personal Reflection 39
Appendix B – Project Schedule (Outdated) 40
Appendix C – Project Schedule (Revised) 41
Appendix D – Minutes of Meeting 42
Appendix E – Requirement Analysis 46
Appendix F – User Requirements 58
Appendix G – User Interface Form 59
Appendix H – Implementation 61
Appendix I – Design View 62
Appendix J – Database Report 66
Appendix K – Testing 68
Appendix L – User Manual 74

Chapter 1 – INTRODUCTION

1.1 – Introduction

This chapter examines the minimum requirements of this project and outlines the business background and associated problems of P & Co Chartered Accountant. The scope of the project will also be clarified in due course.

1.2 – Minimum Requirements

After suggestion from the project assessor, the minimum requirements were revised since the previous version was not made clear. The minimum requirements were agreed to be:

6. To replace the paper-based system in P & Co Chartered Accountant to a computerised system.
7. To gather user requirements from the owner in P & Co; in short, to obtain a clear view of the problem on the existing paper-based system.
8. To meet the user requirements and to recommend a new system by introducing database to store client and sales records.
9. To design and implement a database holding information about P & Co customers and their accounts details.
10. To evaluate the system.

1.3 – Understanding the business background: P & Co Chartered Accountant

P & Co Chartered Accountant is a medium-sized accountancy firm located in London, Leeds and Manchester. There are five employees working in London, three employees working in Manchester and two employees working in Leeds. The account service that P & Co provides includes VAT returns, Bookkeeping, PAYE, Self Assessment Tax Return and general administration service such as writing letters for clients (detailed description of these services can be found in Chapter 3, ‘Requirements and Analysis’).

Inside the three P & Co branches, computers are only used for simple administration (e.g. typing letters). Since the current system is entirely paper-based, the working environment is full of papers, files and filing cabinets. Clients’ details are stored in files in different filing cabinets according to their business identification number. There are about two hundred individual clients’ files at each branch and so, it is very time-consuming to retrieve the files when clients and other branches’ staff request information. There were occasions when client files went missing and delayed the time taken for information retrieval.

There is a computerised accounting system in Manchester and London branches to record clients' sales and payment details. However, the Leeds clients' accounting system is being recorded in Manchester which caused a lot of problems when Leeds clients want to view their account details. The staff in the Leeds branch would have to telephone the Manchester branch to get the required information. As a result of this, the clients had to wait in the Leeds office for a reply, which is not very efficient in terms of customer service. Also, the accounting software only record the clients' sales and payments details but does not produce sales invoice and payments receipt where the accountant need to produce them separately in Microsoft Word. Again this is very time-consuming. However, since the original owner of P & Co was used to the way the company operated and did not want to invest money to change to a computerised system, the owner and the staff just bear with the problems caused by the paper-based system.

Six months ago, a new owner took over the business. The business was run as three autonomous branches but is now being reorganised so that works are centralised. Specialist work (e.g. VAT, PAYE etc) is increasingly transferred from one branch to another leading to the need to share information. When the London branches' staff request information, the Manchester or Leeds branches' staff will need to telephone or fax the required information back as a reply. The owner therefore realised the problems associated with the current paper based system. He intended to update the three branches with a computerised system in July 2003 using document scanning storage system 'ScanSoft PaperPort Deluxe 8.0' *[1]* to reduce the piles of paper in the office. However, this system only scans the document into the file directories on the network. Since the documents will be indexed by client_id and date, therefore, when data needs to be viewed, the staff will still have to search through the records in the file directories to look for clients' information, and the problem of time-consumption still exists.

As a result, the owner saw the need to build a computerised system that can be used in all three branches to centre all the clients' information in one place. The owner of P & Co would also like to change the accounting system since both Manchester and London offices' accounting systems are difficult to use and cannot provide standard sales invoice and payment receipt.

1.4 – Problems

There are two types of problems in P & Co Chartered Accountant and they are shown below:

Existing problems in a branch

- **Difficulties in searching:** Searching for specific clients' information is very difficult in all three branches as the staff will need to look in the filing cabinet to retrieve files. For example, when a letter is being sent to P & Co office with only the client's name, the staff will not know what

business it should belong to. The problems get worse if clients' files are missing.

- **Time consumption:** Looking for a specific details in clients' folder can be very time consuming.
- **No standard accounting system across all branches:** The accounting system to record clients' sales and payment are only available in the Manchester and London branches, but not in the Leeds branch which is very inconvenient. Also, the present accounting system does not produce sales invoices and payment receipts.

Problems associated with shared information access across the branches

- **Time consuming and geographical problem:** As works are now transferred across branches, the ease of information retrieval becomes a very important factor to completing a job efficiently. The staff in one branch would often need to telephone other branches' staff for clients' information, which is inconvenient, inefficient, cost ineffective and time-consuming.

The above problems prompted P & Co to consider buying an off-the-shelf database-based package, but there are no such tailored packages that can be found on the market. On the other hand, there are packages that can partially meet the P & Co sales record management system requirements. The most suitable accounting package is called 'Sage' *[2]*, which costs almost £2000, but it does not meet the full user requirements of the client record management system. Also, the software package is over the budget and there may be problems amending it that can cost even more. Therefore, the author of this report has been asked to produce a system that would solve the problem whilst reducing the cost. The current system procedures and user requirements will be discussed fully in Chapter 3.

1.5 – Feasibility Study

Feasibility study is used widely in system development projects because it provides a sense of direction and will save time and money by identifying the feasibility of the project. In order to ensure the project is valuable and feasible, a brief feasibility study is examined in three aspects – technical, financial and social.

- **Technical Feasibility:** The implementation will involve a cheap and easy to use relational database using Microsoft Access, and the staff have a basic knowledge in using Access. The design and internal structure of the database is relatively straightforward and can be implemented, therefore it is technically feasible.
- **Financial Feasibility:** The project is considerably low cost because P & Co has Microsoft Access 2000 installed in PCs already.
- **Social Feasibility:** By ensuring user involvement in the development of the project, the social aspect is feasible to achieve. Users will be involved in setting the minimum requirements of the new system and the evaluation stages, allowing the system to be focused closely to the user needs.

the system created has been designed to be as straightforward as possible since most staff lack strong computer skills.

The project was considered feasible by the client sponsored.

1.6 - Scope of the Project

The aim of this project is to implement a client and sales record management system for P & Co Chartered Accountant. In order to meet the project aim, the solution will be to design and implement a database that can add, update, delete, and find clients' records as well as to record clients' sales and payments for the company. Each member of the branch staff can therefore use the database to keep a client's information and to retrieve the information they require. The architecture of the database to be networked will also be introduced, provided there is sufficient time available, so that other branch staff can also obtain access to the updated database (e.g. Manchester and London branch staff can also get access to Leeds' client and sales record database using Active Server Page).

Chapter 2 – BACKGROUND RESEARCH

2.1 – Introduction

This chapter is to outline the methodology the author has chosen for this project as well as the schedule of the project.

2.2 - APPROACH AND PROJECT PLAN

In order to ensure this project is successful and would be completed on time, employing a suitable methodology is one of the most critical factors needed to be considered. The methodology acts as a guideline to the procedures of the project and will help to indicate how the system should be produced. A methodology (described in 2.1.1) has been chosen and an updated project schedule (shown in Appendix C) has been created, which identifies the main phases of the project.

2.2.1 – Methodology

There are many different types of methodologies related to project management, but because of the word limit of this project, only two popular methodologies, the 'SSADM' and the 'Waterfall model' have been chosen to compare in order to ascertain the methodology most appropriate for this project.

SSADM

SSADM (Structured Systems Analysis and Design Methodology) is a methodology (a system of ways of doing things, particularly suitable and orderly procedures), used in the analysis and design stages of systems development.

"It is said to be a data-driven methodology because of its history and emphasis on data modelling and the database, but in its later versions has become more balanced, with, for example, importance attached to the role of users." *[3], p167*

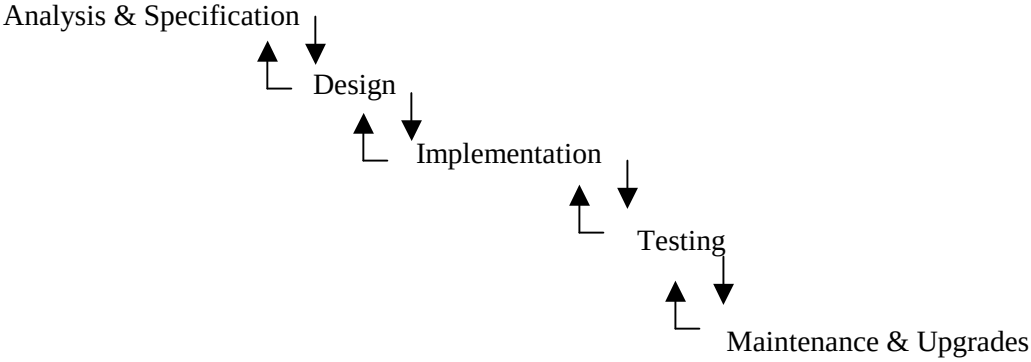
This methodology has primarily, emphasis on documentation, data modelling and the involvement of the user, but not with later project stages including implementation and testing. This breaks down into a hierarchy of stages and they are as follows:

- Stage 0: Feasibility
- Stage 1: Investigation of Current Environment
- Stage 2: Business System Options
- Stage 3: Definition of Requirements
- Stage 4: Technical System Options
- Stage 5: Logical Design System
- Stage 6: Physical Design System

Waterfall Model

The Waterfall model is a very old methodology that originates from the manufacturing industry. A project is split up into five stages as shown in the diagram below:

"The waterfall model – this is the 'classic' model of system development." *[4], p36*



By comparing SSADM to Waterfall model, the author decided to choose Waterfall model to guide me through the duration of this project in order to fulfil the minimum requirements. Although SSADM has the feasibility stages and provides a more detailed analysis and design stages of systems development, it does not allow iteration and does not cover issues of the implementation, testing and evaluation of the software. Therefore, it will not suitable to use towards the latter stage of the project. However, Waterfall model *[5], p15* considers all the stages, it considers the process from requirements analysis through specification, design, coding, testing to documentation and maintenance and upgrades. In addition, iteration is allowed should any modifications to the work already carried out be required. Thus, the waterfall model allows the project to be complete in a logical order, and permits stages to be revisited if new requirements discovered or there are any changes need to be made. This is a form of Rapid Application Development following the waterfall model as an outline framework.

Also, since it already has been agreed in Section 1.6 that the project is feasible and should be continued to develop into further stages, there is no need to carry out another feasibility study as stated in SSADM. Moreover, The emphasis of the waterfall model is on moving down the model; the iteration through steps is limited. Therefore, the process of the project would benefit from it because through testing is completed on each section as the project proceeds, the risk of having to rework tasks that were previously thought of as completed will be minimised.

2.2.2 – Schedule and Project Management

"Project Management ensures that a project is completed on time, within budget and with the required functionality and quality" *[3], p205*

Project management is extremely important to a project like this, which has a very limited time-scale and has to be completed on time. In order to create a framework for managing the project, a project schedule is constructed and represented as a Gantt-chart *(*in *Appendix B, Page 40)* illustrating the duration of different activities. Following the discussion with the project assessor, a more realistic and up-to-date schedule has been realised. Therefore, the revised version is being represented in *Appendix C, Page 41*. Six main phases of the project have been identified and they are as follows:

Phase 1: This is the first stage of the waterfall model. It will involve carrying out some initial research, studying relevant books and web sites to obtain the basic knowledge about underlying technologies in order to do the rest of the project. It will also include gathering requirements from the owners of the problem and analysing them.

Phase 2 and Phase 3: These are derived by the waterfall model stage two, three and four, naturally the design, coding and testing. Phase 2 will consist of implementing a functional dependency analysis, normalisation and design on the database. Implementing the new database and producing standard clients' sales invoice and payment receipt will also be incorporated in this phase. Designing, implementing and testing the data entry interface will make up phase 3.

Phase 4: This stage will be to deal with further enhancements if there is sufficient time.

Phase 5: This is the fifth stage of the waterfall model. It will require an evaluation of the finished product.

Phase 6: Writing up the Project Report is involved in this phase. This is carried out simultaneously with other phases.

Chapter 3 – REQUIREMENTS AND ANALYSIS

3.1 – Introduction

This chapter is the first stage of the 'Waterfall model', which is 'analysis and specification'. To deliver the right system as required by the users, the development of a system must be user-oriented. Thus, it is vital to involve users in the development of the system so as to identify the full requirements.

An overview of the background to the problem of P & Co is given in Section 1.4 and 1.5. Section 3.2 will provide a more detailed analysis of the current system procedures and associated problems. The flow diagram of the activities performed in P & Co and the research on the selection of the most suitable software tool for implementation will also be included in this chapter.

3.2 – Current System Procedures

In order to understand the current paper-based system procedures in P & Co and identify the requirements for the new system, two meetings were arranged with Jonathan Pitayanukul, the owner of P & Co. He is now working in the Leeds, Manchester and London offices on different days of the week. Thus, he should have the knowledge of what is needed in the new system. The detailed minutes of the meeting with Jonathan can be founded in *Appendix D, Pages 42-45*. The following is a summary description of the procedures and problems in the current paper-based system of each branch.

Client and Business Details

- A new paper-based file is created to record the new client and business details. Additional correspondence is also filed into it; for instance, their bookkeeping details and tax details etc. The file is then placed in one of the filing cabinets according to the business identification number.
- If the staff wish to find specific client or business details, they will search through the specific cabinet which contains the file.
- Updates and deletions of clients or business details are always written on paper first until the staff have time to find the specific client files to make the correction. However, the staff sometimes forget to do this, or lose the paper which cannot make the amendment.

Bookkeeping Details

- Clients who make less than £55,000 profit a year in their businesses need a bookkeeping service every three months. The staff who collects the thirteen weeks' bookkeeping takings sheets (the sheets which record each weeks' profits and expenses) and purchase invoices from a client, will record the date the bookkeeping is brought in and completed, as well as who received it on a piece

of paper as a record. The staff may occasionally forget to do this or lose the paper which cannot keep track of the record. In the worst case, the bookkeeping may be left incomplete until the client rings up and checks the process.

VAT Details

- Clients who make more than or equal to £55,000 a year in their businesses need a VAT service every three months. A paper is prepared every month by the staff to record which client needs to complete the VAT and when the client has brought the VAT taking sheets in (but sometimes the paper prepared may be out-of-date). The staff must make sure all the clients have brought the VAT in by the fifteenth of every month for the accountant to work on since there will be a penalty from HMCE if the payment is not posted in by the end of the month. Again, the staff may sometimes forget to or lose the paper which cause problems (e.g. penalty).
- As mentioned in Section 1.4 and 1.5, works are now transferred across three branches, when the client telephones to check the process of the VAT workings, the staff will have to ring the responsible branch for the process and then reply back to the client. This process is very time-consuming, which will delay the job on which the staff is working.

PAYE Details

- PAYE means 'pay as you earn', the employees and employers have to pay national insurance contributions (NIC) and tax on what they earn. Manchester office is now responsible for all three branches' PAYE services, the accountant who deals with it always needs to request other branches' staff to look for information in order to complete the work. Details of PAYE information is placed in another block of filing cabinets which the staff will need to search through. This is again very time-consuming.

Tax Details

- Each person who is self-employed needs to review his/her tax situation every year. The London office is now responsible for completing self-assessment tax returns as the specialist accountant is situated in London. Thus, the Manchester and Leeds offices' staff always need to find the required information for the London accountant to complete the work.

Client Sales Ledger Details (same as Clients' Sales and Payment Record Details)

- Only London and Manchester offices have the system to record clients' sales and payments while Leeds clients' accounting system is being kept in Manchester office. Therefore, once a service is provided to the client, or the client has made a payment, the staff in Leeds office will need to inform Manchester either by phone or by fax for updating the clients' sales and payment records.

- If a client rings up the Leeds office to check how much he/she has owed P & Co, the Leeds' branch staff will then ring up the Manchester branch to get the required information which may cause low workload when there are many requests from clients.
- There are no standard sales invoices and payment receipts in the current system, which means that the staff will need to produce invoices every time they charge clients or receive payment from clients.

3.3 – Flow Diagram

A summarised flow of activities performed in P & Co Chartered Accountant is shown below in Figure 3.1. This diagram will assist the design and implementation phases in the latter chapter.

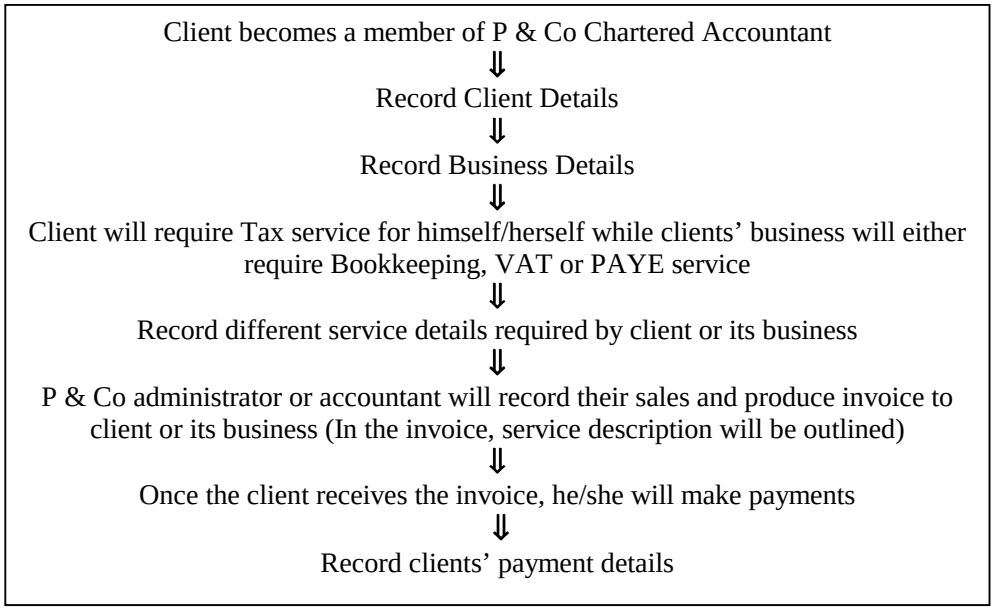


Figure 3.1: Flow of activities in P & Co Chartered Accountant

3.4 – Requirement Analysis

As the current system procedures and problems are known, the requirements will be analysed using the Use Case Model. The Use Case Model is comprised of Use Case Diagrams and Use Case Descriptions, both of which can be found in *Appendix E, Pages 46-57* . Use Case Diagrams are used primarily to capture the high level user-functional requirements of a system *[6], p24*. It aids the developer to model the database requirements and to communicate with the end users on what the database will do. Use Case Descriptions describe the process behind each use case within the diagrams. Moreover, due to the time limit on this project, only some of the client's use case descriptions are included as an example.

A list of user requirements (agreed by the client) devised from the use case models can be found in *Appendix F, Page 58*. They are not in order of priority. The requirements are then translated into the

functional requirements for the purposed database which is shown in Section 3.3.1. There are also some non-functional requirements of the system suggested from the client below in Section 3.3.2.

3.4.1 – Functional requirements

Functional requirements are activities that the new system is required to perform, i.e. what functions the system supports. The activities that they cover consist of storing and retrieving data or producing reports. Below is the list that the system must enable the user to do:

Details	Function
Client	Adding, finding, editing and deleting **Client details** must be possible.
Business	Adding, finding, editing and deleting **Clients' Business details** must be possible.
Bookkeeping	Adding, finding and editing **Bookeeping details** of the client business must be possible.
VAT	Adding, finding and editing **VAT details** of the client business must be possible.
PAYE	Adding, finding and editing **PAYE details** of the client business must be possible.
Tax	Adding, finding and editing the **Tax details** of the client must be possible.
Sales	Adding, finding and editing **Sales details** of the client business must be possible.
	Ability to print the **Sales Invoice** and the **Clients' Sales Summary Report**.
Payment	Adding, finding and editing Payment **details** of the client business must be possible
	Ability to print the **Payment Invoice.**
Client Sales Ledger	Ability to find the client sales ledger record and print the **Client Sales Ledger**.

For *'Bookkeeping, VAT, PAYE, Tax, Sales and Payment Details'*, there is no need to have a 'delete' function because the services and invoices that P & Co offered to client or his business will need to keep track of the record for future practice. If the records are deleted, the information will be lost. However, for *'Client or Business Details'*, the record can be deleted if the client or his business has not needed any service from P & Co. The *'Client Sales Ledger Details'* is retrieved from the sales and payment details and therefore, records should be added and updated in those two tables.

There are some enhancements that the client would like to have on the database which are:

- Produce an up-to-date printout of monthly VAT report;
- Design the architecture of the networking of the database, but this is not important since there will be a IT technician coming from Taiwan to work in P & Co in May 2003 and he would be able to carry on the unfinished work.

These will only be implemented provided there is sufficient time.

3.4.2 – Non-functional Requirements

Non-functional requirements of the system are those not directly linked to the functions or data that the system supports. The system will need to:

- Be restricted to authorised users only.
- Provide a simple, easy to use data entry interface which will operate in real time to enable the client to enter details whilst on the phone.

3.5 – Selection of Implementation Tools

This stage of analysis is to select the most appropriate method of implementing the solution. It has been decided that creating a *database* is the most suitable solution for this development because the development will involve a large amount of related data *[7], p4*.

So as to select the most suitable DBMS to be implemented in this development, the four most popular DBMS used by the accounting industries have been chosen to be compared before making a decision. They are Oracle, Microsoft SQL, DB2 and Microsoft Access. The table drawn below is the summary of features and performance of these four different DBMS devised from *[8], [9], [10] and [11]*. They are in the order of importance relating to the development of this project:

Features and performance	Oracle	Microsoft SQL Server 7	DB2	Microsoft Access 2000
Ease of use	3rd Ease of use	2nd Ease of use	3rd Ease of use	1st Ease of use
Cost	Most Expensive	Third Expensive	Second Expensive	Cheapest
Security	Best	Good	Good	Reasonable
Concurrent access	Available	Available	Available	Available
High transactions data entry per day	Suitable	Suitable	Suitable	Not Suitable
Functionality	Best	Best	Good	Good
Platform	All platforms	Windows only	All platforms	Windows only

Figure 3.1 Comparison of four different DBMS

The factors that are most vital in deciding which DBMS to use for this project are *Cost, Ease of use and Security*. *Ease of use* is the most important issue in this development because the end users only have limited knowledge in using computers. Therefore, the DBMS used must be sufficiently easy-to-use. *Cost* is also another important factor since P & Co will need to invest money in updating new PCs, monitors and in purchasing the ScanSoft PaperPort software. The company budget for the development of this project is £200. In terms of security, as the data holding in the database is confidential, the DBMS chosen must be able to provide reasonable security such as a login password. Concurrent access and transactions entry per day are not so vital as the maximum concurrent users and the average transactions data entry per day are low (as shown in *Appendix D, Page 42-45*).

From figure 3.1, it can be seen that Oracle and DB2 support all known platforms, provide concurrent access, good functionality, good security and are suitable for high transactions data entry per day. However, they are difficult to use and very expensive which disregard the two most important factors that needs to be considered. Oracle costs £7616 with licenses *[12]* which has far exceeded P & Co's budget.

By comparing Microsoft SQL Server 7 to Microsoft Access 2000, they both support a window-based platform. Microsoft SQL Server is cheaper to buy than Oracle and DB2 for most configurations *[13], p354*. However, it costs approximately £910 *[14]* which exceeds the budget set. Microsoft Access is the cheapest amongst the four DBMS compared and the easiest to use although Microsoft SQL scores better in terms of security and functionality.

As a result, the selected database software tool is Microsoft Access 2000 because it scores better in the two most important factors, *ease of use and cost*. This is essential as P & Co only has Access 2000 available on its PC and therefore, does not need to purchase another piece of software. Also, maintenance of the database will be maintained by the technician in P & Co, who is only familiar with Access.

In terms of ease of use and functionality of Access, Access provides tools which can be mastered by even the most inexperienced database users. Querying, forms and report creation are fairly simple tasks, especially when using the main toolbars, wizards, and graphical interfaces provided with Microsoft Access *[11]*. Therefore, Access will be more suitable for the end users given the computer knowledge they have. Login password of the database is also available in Access to restrict admission.

It is suggested that if the number of concurrent users is more than ten, Microsoft SQL Server is more appropriate *[15]*, and if the company is in high production data entry environment, Microsoft Access is definitely not a good choice *[11]*. As stated by the client, the maximum number of concurrent users will be eight and the maximum transactions data entry per day will not exceed twenty. It has also been said by the client that there is no plan in expanding the company in the future two years and thus, the concurrent users and transactions rate per day will not vary much in the coming years.

Finally, Access has been investigated to ensure that it has enough memory capabilities to support this database development. It has been discovered that the maximum database size is 1Gb *[10]*. As the data entered in the database will be a small volume, it is only estimated to be 0.1 GB. This is still far less than the 1GB that an Access database application can be. Therefore, Microsoft Access 2000 is the most suitable DBMS to use in this database development.

Chapter 4 – DATABASE DESIGN

4.1 – Introduction

'Design' is the second stage in the 'Waterfall model'. The design of the database is very important and cannot be overlooked. The P & Co database was first designed incorrectly, which cost an amount of time to correct the design and fix the problems in the next stage, the implementation phase. Therefore, it is vital to ensure the design is accurate and complete, all the requirements in the analysis phase are captured, and that redundancy and inconsistencies in the data are kept to a minimum.

This chapter is intended to identify the entities and attributes required in the P & Co Chartered Accountant database. It will also show the relationships between the tables by means of an entity-relationship modelling diagram, and to look at the functional dependencies of each table. The design of the database is based on the requirements specified by the users, the flow diagram in Section 3.3 and the use cases diagrams drawn in *Appendix E, Page 46-57* . The user data-entry interface design will also be examined after the database design.

4.2 – Entity Relationship Modelling

E-R modelling was first introduced by Chen in 1976 and is now widely used. It is an important method for modelling all the information needed in the database ready for database implementation. This model can be used to assist in the communication between the programmer and the user. The structure of the information that is to be represented in a database is thought of in three types of component:

- Entities, which are distinct things in the enterprise;
- Relationships, which are meaningful interactions between the objects;
- Attributes, which are the properties of the entities and relationships; *[16], p177*

Entities are represented as rectangular boxes in E-R modelling. Nine entities are defined in this database application according to the user requirements, They are: *Client, Business, Bookkeeping, VAT, PAYE, Tax, Sales, Payment and Service.* Note that Bookkeeping, VAT, PAYE, Tax are the four types of services provided by P & Co but they are named as the different entities. Entity 'Service' is identified to assist in recording sales and producing invoices.

As the nine entities are identified, the next step is to identify the relationship between entities. A relationship can be classified by differing degrees. They can be one-to-one (1:1), one-to-many (1:M) or many-to-many (M:N) relationships *[17], p169.* For example, there is a relationship between entities

Client and Business – 'owns' and the relationship between them is many-to-many. A client may own one or more businesses and a business may be owned by more than one client. The E-R Diagram and the explanations of the relationships between the entities of proposed database are set out below:

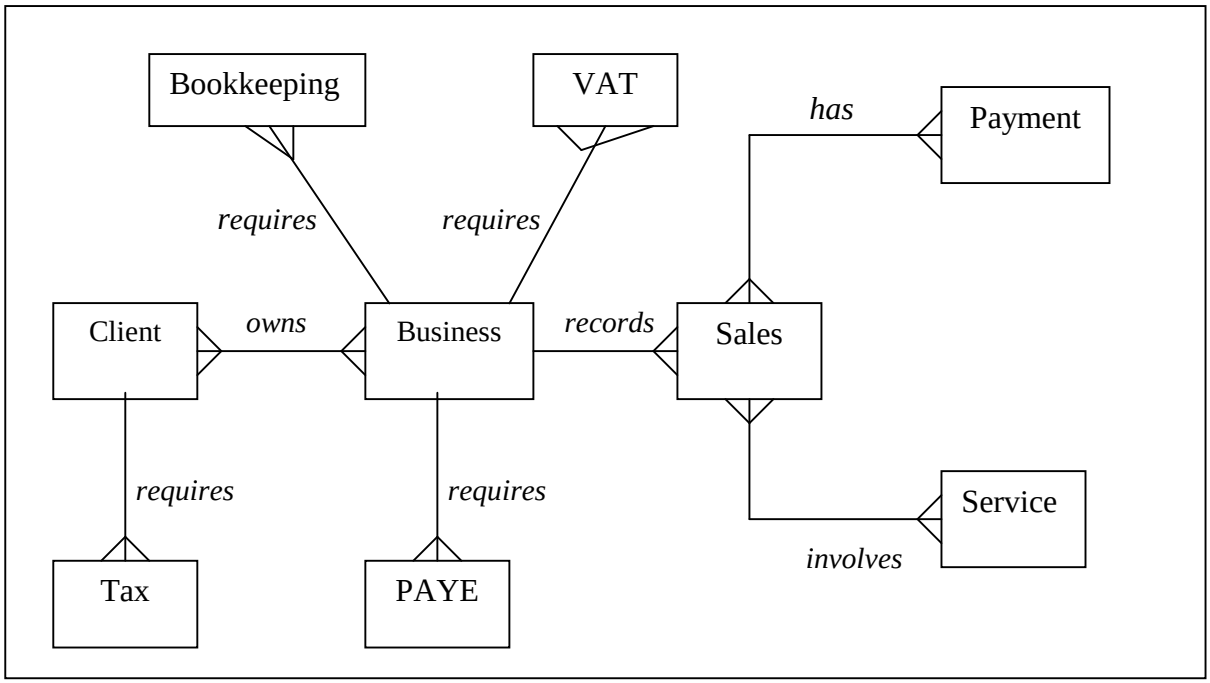


Figure 4.1 : E-R Diagram of the Proposed Database

Entity	Relationship type	Entity	Relationship Degree	Relationship
Client	Owns	Business	M:N	A client owns one or more businesses and a business may be owned by one or more clients.
Client	Requires	Tax	1:M	Each Tax provided for the client has its own Tax_sheet_Id. A client requires one or more Tax but a Tax is required by one client.
Business	Requires	Bookkeeping	1:M	Each bookkeeping provided for the business has its own bookkeeping_ledger_Id. A business requires one or more bookkeeping but a bookkeeping is required by one business.
Business	Requires	VAT	1:M	Each VAT provided for the business has its own VAT_ledger_Id. A business requires one or more VAT but a VAT is required by one business.
Business	Requires	PAYE	1:M	Each PAYE provided for the business has its own PAYE_sheet_Id. A business requires one or more PAYE but a PAYE is required by one business.
Business	Has	Sales	1:M	Each clients' business has one or more sales in P & Co but a sales is from one business (since each sales invoice has its own unique id).
Sales	Has	Payment	M:N	A sales that is invoiced has one or more payment (i.e. clients might not have paid the full amount) and a payment made has one or more sales invoice.
Sales	Involves	Service	M:N	A sales from P & Co to client involves one or more service and a service involves in one or more sales invoice.

Figure 4.2 – The explanations of the relationships between the entities

4.2.1 – Mapping the E-R Model into Relational Tables

The E-R diagram created in Figure 4.1 aids the design of the tables and implementation in Microsoft Access. The main aim of tables in a database is for the storage and retrieval of data required to generate the information necessary for the user. The tables were designed according to the user requirements identified in Section 3.3.1 and the E-R diagram shown in Figure 4.1. The relationships between the entities will then be mapped onto the tables. Data related to each entity are stored as attributes in the tables. The nine entity names will be kept the same as the table names so as to eliminate confusion. Six rules should be applied when undertaking the mapping of the E-R Model into the actual tables (schema) within the database, but due to the fact that only three rules apply to my database application, I will only outline those three rules.

1. Mapping Entity-types

Entities are classified into entity types *[18]*. Each entity-type maps onto a table scheme with associated fields comprising the attributes of each entity-type *[19], p48.* This rule implies that a primary key must be chosen to uniquely identify each record in the table in order to ensure that data redundancy and record duplication do not occur, which would result in errors and space wastage. The nine entities in the tables have appointed a primary key to uniquely identify each record. Therefore, it would prevent data redundancy and record duplication occurring in the database.

2. Mapping 1:M Relationship-types

Relationships are classified into relationship-types *[18]*. 1:M relationship in the E-R Model is represented in the database by posting the primary key of the master entity-type into the scheme of the detail entity-type. For example, scheme 'Business' is the master entity and schemes 'Bookkeeping, VAT, PAYE and Sales' are the detail entities. The relationships between them are one to many; the primary key of *Business* is therefore posted into the schemes of *Bookkeeping, VAT, PAYE and Sales*.

3. Mapping M:N Relationship-types

M:N relationship-types are mapped onto new relationship schemes. In other words, a new table must be created to represent the relationship-type, which will contain the two primary keys of the entity-types participating in the relationship *[19], p50.* There are three M:N relationships in this E-R Model and they are shown in Figure 4.1 and 4.2. Three new tables containing the two primary keys of the entity-types participating in the relationship have been created, they are named '*Client_Business*' (for the M:N relationships between Client and Business), '*Sales_Payment*' (for the M:N relationships between Sales and Payment) and '*Sales_Service*' (for the M:N relationships between Sales and Service).

The list of tables accompanied with all the attributes are given in the table below:

Table	Attributes
Client	Client_Id, Title, First_Name, Surname, Home_Tel, Mobile, Home_Add_Line1, Home_Add_Line2, Postcode, Email, DOB, National_Ins_No *The primary key in this table is Client_Id, a unique identifier to determine different clients.*
Business	Business_Id, Business_Name, Type_Of_Business, Date_Business_Start, Business_Tel, Business_Add_Line1, Business_Add_Line2, Postcode, Status, A/C_YE *The primary key in this table is Business_Id, a unique identifier to determine different businesses.*
Client_Business	Client_Id, Business_Id *This table is used to create a M:N relationships between the tables 'Client' and 'Business'. The primary key is a composite, this allows both 'Client_Id' and 'Business_Id' to be duplicated, but no combination of the two can be duplicated.*
Tax	Tax_Sheet_Id, Client_Id, Tax_Ref, Tax_Office, Date_Finished, Tax_Service_Charge *The primary key in this table is Tax_Sheet_Id, a unique identifier to determine different Tax services completed by P & Co for the client. Client_id is the foreign key in this table as a cross-reference to client.*
Bookkeeping	BKKeeping_Ledger_Id, Business_Id, Date_Received, Date_Completed, Record_By, Service_Charge *The primary key in this table is BKKeeping_Ledger_Id, a unique identifier to determine different bookkeeping services completed by P & Co for the clients' business. Business_id is the foreign key in this table as a cross-reference to clients' business.*
VAT	VAT_Ledger_Id, Business_Id, VAT_Reg_No, VAT_Period, Date_Received, Date_Finished, Received_By, VAT_Service_Charge *The primary key in this table is VAT_Ledger_Id, a unique identifier to determine different VAT services completed by P & Co for the clients' business. Business_id is the foreign key in this table as a cross-reference to clients' business.*
PAYE	PAYE_Sheet_Id, Business_Id, PAYE_Ref, PAYE_Office, Date_Finished, PAYE_Service_Charge *The primary key in this table is PAYE_Sheet_Id, a unique identifier to determine different PAYE service completed by P & Co for the clients' business. Business_id is the foreign key in this table as a cross-reference to clients' business.*
Sales	Invoice_No, Business_Id, Date, Subtotal, VAT, Total *The primary key in this table is Invoice_No, a unique identifier to determine different Invoices issued by P & Co either to the client or its business. Business_id is the foreign key in this table as a cross-reference to clients' business.*
Payment	Payment_Id, Payment_Date, Collected_By, Amount_Owed, Amount_Outstanding, Amount_Paid, Balance *The primary key in this table is Payment_Id, a unique identifier to determine different payments made by the client to P & Co.*
Sales_Payment	Payment_Id, Invoice_No *This table is used to create a M:N relationships between the tables 'Sales' and 'Payment'. The primary key is a composite, this allows both 'Payment_Id' and 'Invoice_No' to be duplicated, but no combination of the two can be duplicated.*
Service	Service_Id, Description, Service_Charge *The primary key in this table is Service_Id, a unique identifier to determine different services provided by P & Co to the client or its business.*
Sales_Service	Invoice_No, Service_Id, Sheet_No *This table is used to create a M:N relationships between the tables 'Sales' and 'Service'. The primary key is a composite, this allows both 'Invoice_No' and 'Service_Id' to be duplicated, but no combination of the two can be duplicated.*

4.3 – Integrity Constraints

The relational model created for the database relies mainly on the use of primary keys and foreign keys. The primary key uniquely identifies each record in the table *[20]*. A foreign key is a field in a relational table that matches the primary key column of another table, it can be used to cross-reference tables *[21]*. To ensure both primary and foreign keys are properly defined, there are two rules to be followed – they are *Entity Integrity* and *Referential Integrity*.

4.3.1 – Entity Integrity

This rule states that, "No part of a primary key can be null, 'no part of' cover cases where the primary key comprises more than one attribute" *[22], p6.2*. It means that for every row in the relation a value must be given for the primary key field. For example, the primary key field for any record within the 'Client', 'Business', 'Bookkeeping', 'VAT', 'PAYE', 'Tax', 'Invoice', 'Payment' and 'Service' table must consist of a value.

4.3.2 – Referential Integrity

This rule states that, "If a base table, R_2, includes a foreign key, F_k, matching the primary key, P_k, of some base table, R_1, then every value of F_k in R_2 must either be:

a. equal to the value of P_k in some row of R_1 or
b. wholly null" *[22], p6.3*

Referential integrity means that, if a *Client_Id* in the *Client* table has a value, then it must match a *Client_Id* of the *Tax* table. Enforcing the referential integrity ensures that records in one table match the records in a related table, which can protect the relationships between tables. This is vital to avoid erroneous differences between records in different tables. All the tables in the proposed database have enforced the referential integrity rules in this development.

4.4 – Normalisation

Normalisation is used to put the relational schema through a series of tests to ensure that it complies with a particular Normal Form *[7], p483*. "Relational database theory tells us that if relations are in normal form, then there will be minimum data redundancy and minimum chance for things to go wrong" *[22], p7.1*. Data redundancies occur when data is repeated more than once in a table; this will result in inconsistent data where updates are incompletely performed and make updates more expensive. Normalisation also minimises the insertion, deletion and update anomalies. Data redundancy and data anomalies can have a considerable influence on the performance of a database and therefore, it is important to get it right in the design process.

Before the process of Normalisation can begin, the notion of *Functional Dependencies* must first be outlined. "A functional dependency is the concept of one attribute in a relation being 'functionally dependent' on another (or on a set of others)" *[22], p7.3*. The functional dependencies of the P & Co database are the following, the attribute in bold is the primary key and italic is the foreign key:

Table	Functional Dependencies
Client	**Client_Id** → { Title, First_Name, Surname, Home_Tel, Mobile, Home_Add_Line1, Home_Add_Line2, Postcode, Email, DOB, National_Ins_No }
Business	**Business_Id** → { Business_Name, Type_Of_Business, Date_Business_Start, Business_Tel, Business_Add_Line1, Business_Add_Line2, Postcode, Status, A/C_YE }
Client_Business	**Client_Id, Business_Id** → Both of the attributes are primary keys.
Bookkeeping	**BKKeeping_Ledger_Id** → { *Business_Id*, Date_Received, Date_Completed, Received_By, Service_Charge }
VAT	**VAT_Ledger_Id** → { *Business_Id*, VAT_Reg_No, VAT_Period, Date_Received, Date_Finished, Received_By, VAT_Service_Charge }
PAYE	**PAYE_Sheet_Id** → { *Business_Id*, PAYE_Ref, PAYE_Office, Date_Finished, PAYE_Service_Charge }
Tax	**Tax_Sheet_Id** → { *Client_Id*, Tax_Ref, Tax_Office, Date_Finished, Tax_Service_Charge }
Sales	**Invoice_No** → { *Business_Id*, Date, Subtotal, VAT, Total }
Payment	**Payment_Id** → { Payment_Date, Received_By, Amount_Owed, Amount_Outstanding, Amount_Paid, Balance }
Sales_Payment	**Payment_Id, Invoice_No** → Both of the attributes are primary keys.
Service	**Service_Id** → { Description, Service_Charge }
Sales_Service	**Invoice_No, Service_Id** → { Sheet_No }

As the functional dependencies of the database have been defined, the processes of normalisation can now commence.

First Normal Form (1NF)

First Normal Form states, "A table is in First normal form (1NF) if and only if all columns contain only atomic values; that is, there are no repeating groups (columns) within a row" *[23], p99.* The database is in 1NF because all the attributes in the database contain only atomic values.

Second Normal Form (2NF)

Second Normal Form states, "A relation schema R is in 2NF if every nonprime attribute A in R is fully dependent on the primary key of R." *[7], p488.* As seen from the functional dependencies

defined above, the database is in 2NF since there are only primary keys determining the nonprime attribute.

Third Normal Form (3NF)

"Third Normal Form is based on the concept of transitive dependency. A functional dependency X → Y in a relation schema R is a transitive dependency if X → Z and Z → Y hold" *[7], p489*. "According to Codd's original definition, a relation schema R is in 3NF if it satisfies 2NF and no nonprime attribute of R is transitively dependent on the primary key" *[7], p490*. Since the database is in 2NF and no transitive dependencies exist in the relation schemas, 3NF holds in the database.

Boyce Codd Normal Form (BCNF)

"A relation schema R is in BCNF if wherever there is a non-trivial functional dependency X → A holds in R, then X is a superkey of R" *[7], p494*. BCNF is slightly stricter than 3NF, but "in practice, most relation schemas that are in 3NF are also in BCNF" *[7], p494.* In the database relation schemas, there are no functional dependencies that do not have a superkey and so, BCNF holds.

4.5 – User Interface Design

HCI stands for human computer interaction, it is the study of the way in which humans operate computers *[24], p5*. In this project, studies into HCI ensures the database interface looks professional and to be as 'user-friendly' as possible. This is particularly important because a system will be useless if people cannot understand how to use it, even if the system does exactly what it is meant to do and is technically brilliant. The goals of HCI can be summarised as "to develop or improve the safety, utility, effectiveness, efficiency, and usability of systems that are interacting with computers" *[25], p14*. Amongst those goals, usability is the key concept which is concerned with making systems easy to learn as well as easy to use.

When using Microsoft Access 2000 to build the database, *'Forms'* are one of the most efficient ways of providing the user with an interface. Therefore, forms have been created to permit users to input the data into the database quickly, effectively and easily.

4.5.1 – User Interface Architecture

Before considering the user data-entry interface design, it is vital to first build the user interface hierarchy, which illustrates the different data-entry screens and how they would connect together. The diagram below (Figure 4.3) is devised from user requirements and Section 4.2, E-R Model. It shows how navigation through the forms is structured. For entity 'Service', it has been integrated with the 'FormInvoice'. The Client Sales Ledger is the combination of tables 'Invoice' and 'Payment'. This architecture should also help the implementation phase that will be outlined in Chapter 5.

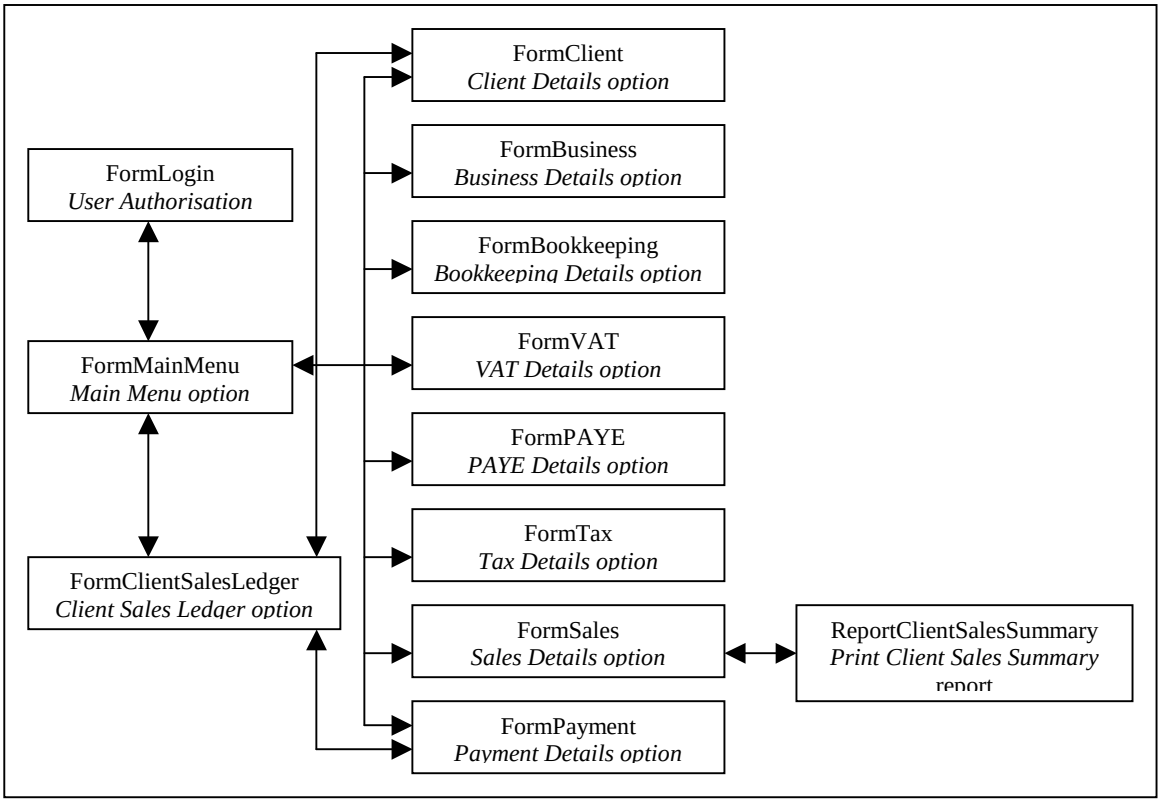


Figure 4.3: User Interface Architecture

4.5.2 – Forms Design

Some useful guidelines regarding HCI issues in creating more usable forms are set out below:

- *Meaningful title and familiar field labels*: avoid computer terminology titles, use terms which users are familiar with.
- *Logical grouping and sequencing of fields*: related fields should be adjacent; sequencing should reflect common patterns.
- *Visually appealing layout of the form*: pay attention to the spacing of entry fields, ensure each alignment is consistent and do not overcrowd the screen.
- *Consistent terminology and abbreviation*: label the text boxes or buttons with the same name if they perform the same function; abbreviation can be used but have to be consistent and brief.
- *Visible space and boundaries for data-entry fields*: use underscores to indicate the number of characters in a field, for example, a small box with '__/__/____' indicates the date format, entry with the similar format, ('25/12/2002') is required. *[25], p494*

While designing the user interface forms, the above guidelines have been followed. A sample of the form is given in *Figure 4.4, Appendix G, Page 59-60*. All the labels have a meaningful title, e.g. the field that needs to input the Surname has the title 'Surname'. In terms of logical grouping and sequencing of fields, related fields are grouped together such as 'Home_Address'. Also, buttons with similar functions are grouped together, for example, 'Add Record', 'Find/Edit Record' and 'Delete Record' belong to the editing function, they are being put together on the middle bottom of the form,

for the searching function button 'Find Sales Ledger'; as it is a different function button, it has been put on the left of the form. Form operations button such as 'exit' has been put on the right hand side of the form. All text boxes and labels are aligned and spaced consistently as seen from the form; text boxes that expect to have a longer input entry are relatively longer than those who have a shorter input entry, i.e. Text box 'First_Name' is longer than 'Title' because there are only four types in 'Title' while many different 'First_Name' are possible – 'Mr, Mrs, Miss and Ms'. Screen size is also one of the factors in this database development since the design of the user interface form varies from size 15' monitor to size 17' monitor. Therefore, all the forms have been fixed to one size so that the design would appear the same among different size monitors.

In order to have a consistent terminology to assist the users to use the database easily, buttons with a common functionality have been labelled with either the same text or icon on every form. Abbreviations have been used in some of the attributes but they are being implemented because the user is familiar with them.

Finally, the data-entry fields, such as 'Home_Tel' and 'Mobile', have an input mask associated with it and so, when the users click on those two fields, field '(_____) _____' will appear. This will enable the users to see what they are expected to type.

Colour and Text

The colour and font style are also issues that can make an interface more usable. The colours of the forms are designed according to the '*Industrial*' style which is one of the functions built inside the Access. One of the user interface form 'Client' (as shown in Appendix H) has been shown to the end user before the implementation of the others. The end user commands that the industrial style looks professional but he would like to have a company logo placed on all forms. Therefore, the company logo 'P & Co Chartered Accountant' has been placed for every form.

"The font used should be what the user uses, this enables the text to be easily read" *[26], p23*. The end user in P & Co confirmed that the most commonly used font in the accountant firm is 'Arial' in size ten, therefore this font and its size has been implemented throughout the database.

Main Menu

In order to assist the new users get to where they want quickly and easily, a 'Main Menu' form has been created. The Main Menu form will automatically appear when the database is being opened. This gives the new user a starting point which they can go back to if they get lost in the database. The form contains nine different pages with obvious function buttons that bring the new users to each input form they want. The sample of the form 'Main Menu' is given in *Chapter 5, Section 5.4,Page 26.*

Chapter 5 - IMPLEMENTATION

5.1 – Introduction

'Implementation' is the third stage according to the 'Waterfall life cycle' methodology. This chapter will first outline the features of Microsoft Access, the description of the implementation of the database tables and relationships based on the design outlined in Chapter 4 followed. The user data-entry interface will then be possible to create, once the tables and relationships have been correctly implemented.

5.2 – Microsoft Access

As shown in Section 3.4.1, selection of software tools, Microsoft Access 2000 was chosen as the software tool used in this database development. Access has a useful set of tools and wizards to help create a database, including *Tables, Queries, Forms, Reports, Macros and Modules [27], p2*. Modules were not used in this database and therefore will not be discussed. The function of each tool is briefly outlined in *Appendix H, Page 61*.

10.3 – Table Implementation

Tables

All the tables listed in Section 4.2.1 in Chapter 4, Design, are implemented in Microsoft Access. The tables were inserted into Microsoft Access in the 'Design View' as opposed to 'Table Wizard'. This allows the attributes to be typed into the table directly, the data types to be selected using drop down menus and the field properties to be defined. For example, when the attribute 'DOB (Date_Of_Birth)' was put in the table 'Client' via 'Design View', data type 'Date' was selected using drop down menus and the field properties were then specified - such as defining an 'input musk' for the date field e.g. '00/00/0000;0;_'. Defining 'input mask' permits users to see the data format once they input a record.

When empty tables have been created, 'field names' are added to every table to state their attributes. The list of attributes can be found in Section 4.2.1. The processes of creating a table are described below and they are in order of priority:

- *Enter the field names* – attributes as shown in Section 4.2.1 were entered in each table. The lists of attributes were agreed by the client that P & Co itself used.
- *Select an appropriate data type* – a suitable data type was selected for each field, for instance, data type 'Currency' was appointed to the field 'Balance'.
- *Provide a brief description* – a brief description was given for each field to show the users the kind of data a field should store.

- *Set field properties* – Properties such as combo box and input mask were used to indicate to the users the data format when inputting a record.
- *Assign primary key* – Primary key was assigned in the relevant tables. The primary key for each table is given in Section 4.2.1. It is simple to assign a primary key in Assess, the field corresponding to the primary key should be highlighted and the 'Primary Key' tool should be selected.

Creating tables is a relatively easy task but the main point is to ensure that each attribute has the correct data type and the primary keys are correctly labelled, because if the keys are not implemented correctly, the relationships and integrity clauses will not be validated and corrected. The table in Figure 5.1 shows the design view of the 'Sales' table. All the other tables were listed in the same way. A sample of the other tables can be seen in *Appendix I, Page 62-65*. The table in Figure 5.1 has Invoice_No as the primary key and Business_Id as the foreign key linking the Business table.

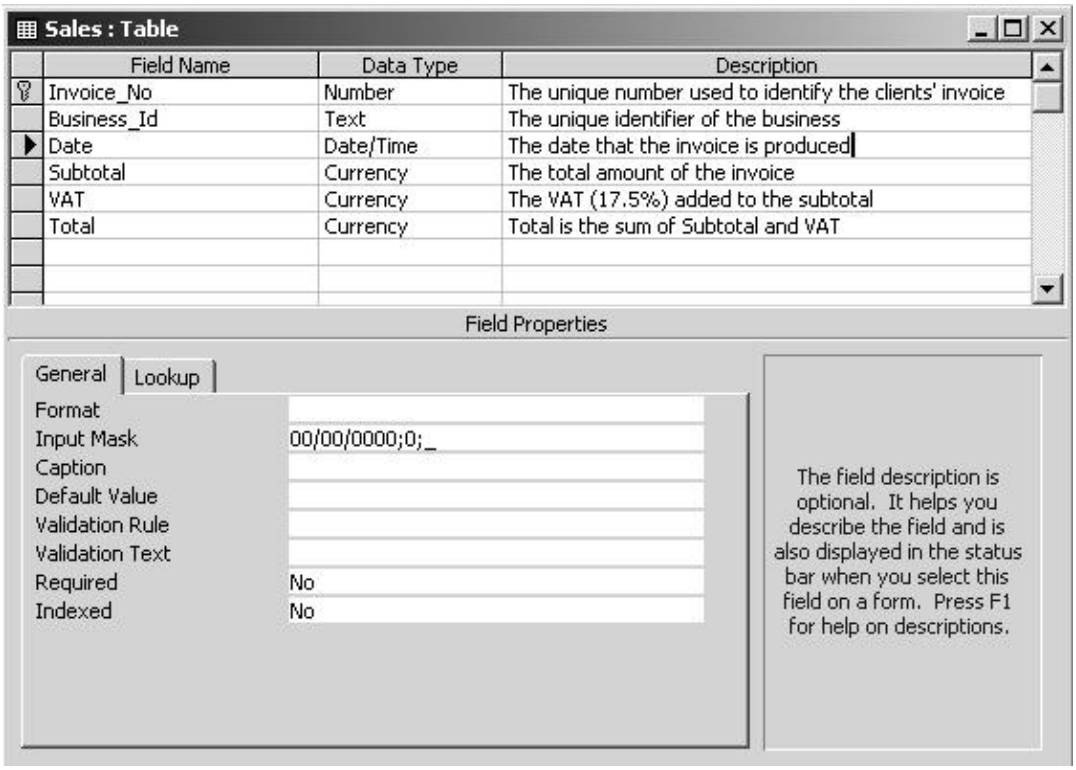


Figure 5.1: Sales Table Design

With the correct implementation of the tables, the next step is to define the relationships and insert the integrity constraints. The tool to link the relationships between tables is already built inside Microsoft Access. All the tables and relationships can be viewed in a simple graphical representation as shown in Figure 5.2. Through selecting the view relationship option in Microsoft Access, all the tables were added, and by dragging the primary key of the source table to the foreign key of the relational table, created the relationships.

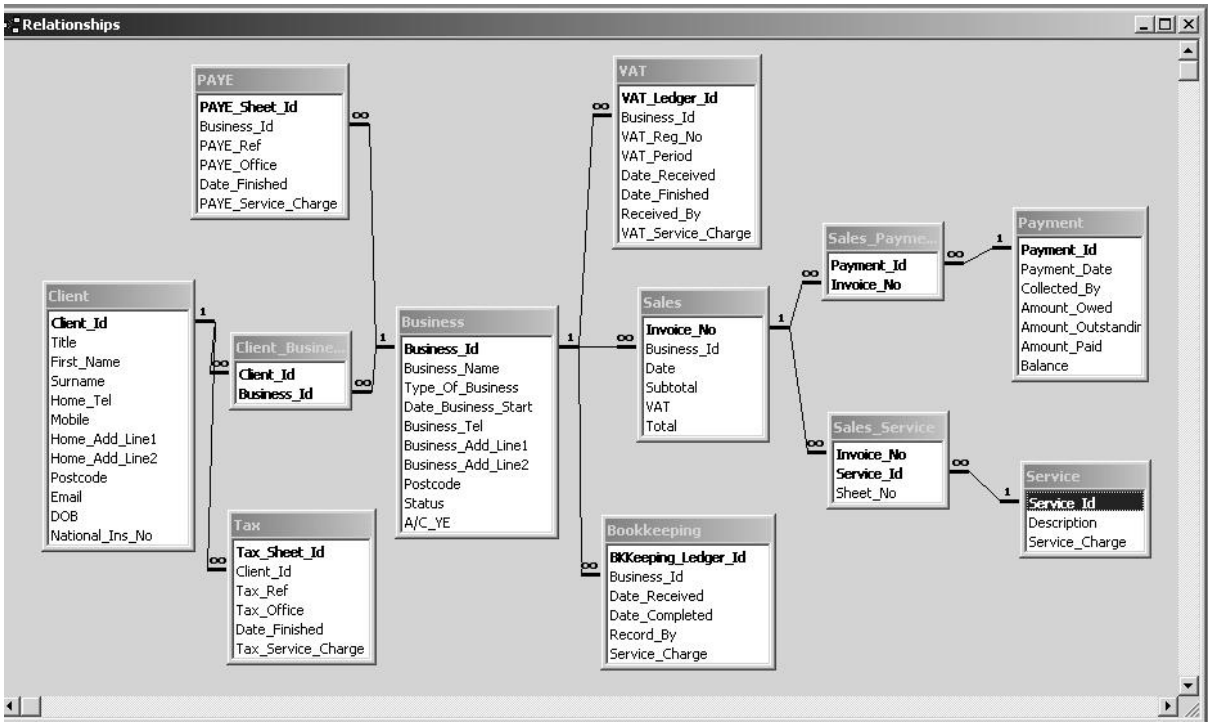


Figure 5.2: Microsoft Access Relationships

To guarantee that the integrity of the database is held, the integrity rules discussed in Section 4.3 must be adhered to in the implementation of the database. *Entity integrity* is achieved by setting the primary key field property *'Required'* to *'Yes'*. Selecting the *'Enforce Referential Integrity'* option in the *'Edit Relationship'* dialogue box enforces Referential Integrity.

5.4 – User Interface Implementation

Creating the user interface as described in Section 4.5.2 (forms design) allows the users to add, edit, delete and find records in a user-friendly manner.

Forms

Forms are created by using 'Table Wizard' mode provided by Microsoft Access. Twelve forms were created as shown in Section 4.5.1 (user interface architecture). There is a 'Main Menu' form, two report forms and nine main forms for data inputs. Implementing user interface using 'Table Wizard' allows you to select the table you want the fields to originate from. The fields you selected will then be placed on the form automatically, but how you want the form to be viewed requires you to move the fields manually to the position that suits it most. The user interface of the forms were designed as simply as possible and the information on the screens designed in such a way that the user could logically navigate his way around the system. The criteria regarding the form design is set out in Section 4.5.2 and the sample of the forms can be found in the Macros Section further on in this chapter.

Queries

The function that queries perform is to retrieve and manipulate specified data. Five queries have been created in this database development to extract specific records. Queries were created by selecting 'Design View'. A 'show table' box with all the database tables then came up, allowing me to select the tables and the fields I want in order to retrieve the specific data. Once all the tables and fields were selected, the queries were saved. Three queries were used within the VAT report option, one query was used in Sales form option and the other was used in Payment form option.

Report

It was only necessary to create one report, the 'Client Sales Summary Report' for this database, but as time allowed, the enhancement for the VAT report was also achieved. The reports were implemented in the Report Design View option and the relevant attributes were selected from the correct database tables. There were three report options involved in the 'VAT Reports form' since P & Co clients varied in the months they needed VAT services. Examples of all the reports were given in *Appendix J, Page 66-67*. The reports have been approved by the P & Co client.

Macros

In order to provide navigation to the user around the system, a tab style Main Menu bar was created. All the forms were linked to the Main Menu using the navigation button such as this  , and the Main Menu form was also linked to all input forms using the navigation buttons (details of the design is given in Section 4.5.2). The Menu screen was loaded automatically once the user opened the database, the user could then access any section of the system by clicking on the tab from the main menu. Figure 5.3 below shows the Client option, by clicking on the 'Find Client Details' button; the user is taken to the client details form that can find a specific record. The navigation buttons and tabs are all standard macro functions built in Access and the wizard was used to create them.

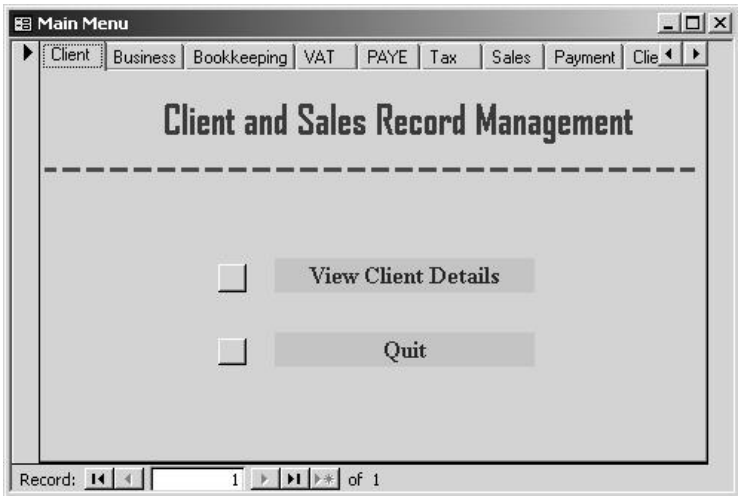


Figure 5.3: Main Menu

In terms of adding, finding/editing and deleting a record, buttons were also created to all input forms using the standard macro functions built in Access. To ensure there is a consistent terminology to assist the users to use the database easily, buttons with a common functionality have been labelled with either the same text or icon on every form. The button 'Find/Edit Record' allows the users to find a specific clients' record. For example, when the users click on the 'Find/Edit Record' button in the form 'Client' (shown in *Appendix G, Page 59-60*), the box below in Figure 5.4 will come up allowing users to type in the details they want to find which matches the whole field in the form. The result of the search will appear once the 'Find What' data is entered and 'Find Next' button is pressed.

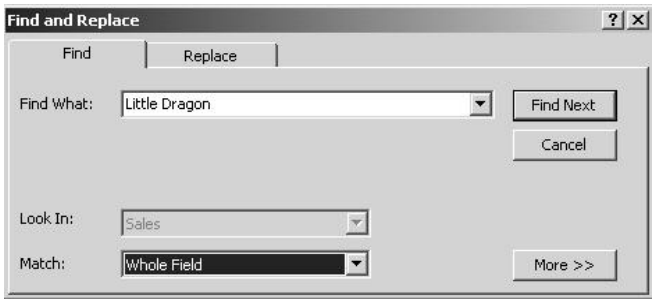


Figure 5.4: Find Box

Visual Basic Access (VBA) code

In the Form 'Sales', as shown in *Figure 5.5*, *Appendix G, Page 59-60*, there is a macro button implemented named 'Save Record' (in the bottom left corner) which will automatically calculate the *Subtotal*, *VAT* and *Total* of the client sales record when the button is clicked. This is achieved by writing a piece of VBA code. The code is then saved as a VBA function, which will be referred directly by the command button name in the properties of the Access icon (see Figure 5.6 and Figure 5.7 below). The VBA code is developed within the Access Design Form by selecting the 'Built Event Option'. Once the user clicks the button, the VBA code performs the function.

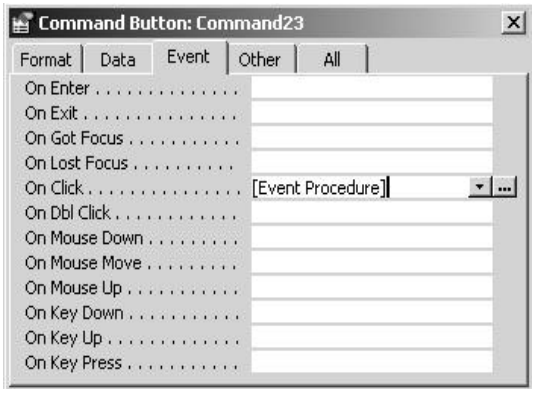


Figure 5.6: Access Icon Properties option

```
**************************************************
'  Function Command23
'  Once the 'Save Record' button is being clicked, it will
'  automatically calculate the following figures

Private Sub Command23_Click()
Subtotal = DSum("Service_Charge", "[Query1]", [Invoice_No]
= [forms]![Sales]![Invoice_No]")
VAT = Subtotal * 0.175
Total = Subtotal + VAT
End Sub
**************************************************
```

Figure 5.7: VBA function 'Command 23'

Visual Basic Access code has also been used in form *'Payment'* (the form is given in Figure 5.7 Appendix G, Page) for the fields *'Amount_Owed' (as shown in Figure 5.8 and 5.9 below)* and *'Balance'(the VBA code is given in Figure 5.10 and 5.11 in Appendix G, Page 59-60)*. The VBA codes were implemented in the same way as the form *'Sales'*. Once the user clicks the field *'Amount_Owed'*, the total amount that the client owed us from the invoices will be summed up. While the user clicks the field *'Balance'*, it will automatically calculate the *'Balance'* by summing up the fields *'Amount_Owed'* and *'Amount_Outstanding'* minus *'Amount_Paid'*.

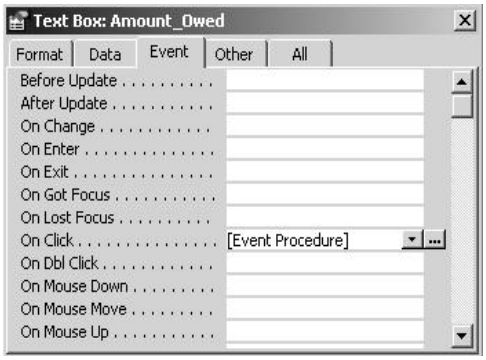


Figure 5.8: Access Icon Properties option

```
**********************************************
'  Function Amount_Owed
'  Once the user clicks the field 'Amount_Owed', the
' total amount from the invoices will be summed up.

Private Sub Amount_Owed_Click()
Amount_Owed = DSum("Total", "[Query2]",
"[Payment_Id] = [forms]![Payment]![Payment_Id]")
End Sub
**********************************************
```

Figure 5.9: VBA function: Amount_Owed

Security

As mentioned in Section 3.3.2, one of the non-functional requirements stated by the client was that the access of the database should be restricted to authorised users only. Therefore, a 'Password Required Login' screen was implemented which is displayed when the database is opened. A correct password must first be entered into the login screen to gain access to the database. This password is held only by the P & Co staff. The password is implemented by opening the P & Co database to exclusive use, then click 'Tools', select 'Security' and 'Set Database Password'. Figures 5.12 and 5.13 below show the procedures of setting the password, while Figure 5.14 shows the login screen.

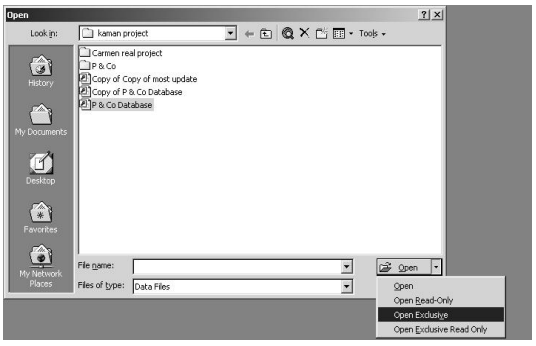


Figure 5.12: Set database to Open Exclusive

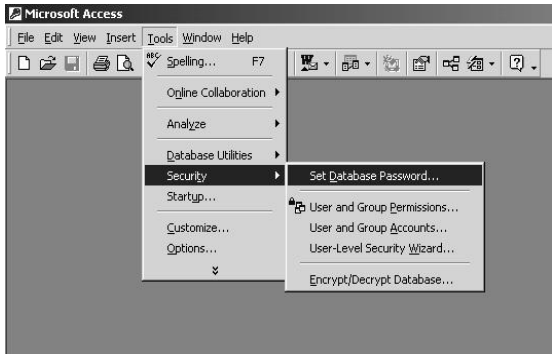


Figure 5.13: Set Password

Figure 5.14: Login Screen

Chapter 6 – TESTING, INSTALLATION AND EVALUATION

6.1 – Introduction

The fourth stage of the Waterfall model is 'Testing'. Testing and evaluation play an important role in any system development. This chapter will therefore feature the procedures used to both test and evaluate the database and discuss some issues involved in installing the database into its operational environment and the maintenance (which is the fifth stage of the Waterfall model).

6.2 – System Testing

Testing is carried out to examine that the database functions correctly and meets the requirements as described in Section 3.3.1. Testing ought to be a continual process rather than a single event carried out once the working database is completed *[28], p13*. Therefore, the system has been subject to a continual testing.

Through testing the database the author implemented, it was showed that the relationship did not link one table to another table properly. Therefore, the author had to work backwards from the testing stage to the designing stage to find out the mistakes. As a result of this, the enhancement period is being reduced from two weeks to one week. Two types of testing had been carried out on each interface form; they are 'Function Testing' and 'Validation Testing', as shown below.

6.2.1 – Function Testing

Function testing confirms that the function performed as required. Each time while a new form was implemented, it was tested to make sure that it was functioning properly. Examples include testing all the buttons on the form that could be 'clicked' to go to other screens were operating, the data that was input or requested was stored in the correct table or retrieved from it.

6.2.2 – Validation Testing

Validation Testing is carried out to ensure that data input is of the correct format and that the data integrity rules as mentioned in Section 4.3 are not violated. The fields on the form with field properties specified were tested to ensure that only valid data is input into the database. For instance, the field 'Home_Tel' should only accept a 'number' field type of format '(_____) _____'. The results of the tests show that an error message would appear if other format of the data type were input into the field.

To ensure the integrity rules are not violated, it is necessary to make sure the records are not duplicated. For example, the Client Details form was tested to ensure that if the 'Client_Id' field is not

being filled, the corresponding record could not be saved. The outcome of the test demonstrates that the database would not permit records to be saved if 'Client_Id' has not been entered. This could avoid data duplication. Each form was being tested by not entering the primary key field and the result was that the corresponding record could not be saved and thus, ensuring the integrity rules is being met.

6.3 – User Acceptance Testing

User acceptance testing involves real users to gather their opinion of the system. This allows the user to try out the system before the system is launched and gain feedback from them. After the function testing and validation testing of the database by the author, the database is still necessary to be tested by the users to ensure it is acceptable to them. It would be ideal for the users to test the system iteratively throughout the development process but due to the time limitations, this was unfeasible. Thus, the user acceptance testing was carried out after the database had been completed.

The testing was conducted based on the database requirements outlined in Section 3.3.1 and the concept of 'usability'. The author chose to emphasise on the level of usability so that the design of this system can be further improved if necessary. Two users had been invited to test the database, details of the test plans, results and comments were gathered in *Appendix K, Page 68-73*. The overall results shows that the staff were very pleased with the functions that the database provided and the system appeared to fulfil its role in providing a user-friendly interface. The database was easy to use; the use of colour and level of navigation were very satisfactory except for some comments on the database design. For example, staff suggested adding a field 'Status' in the 'VAT Report' so that they can see whether a client is active, this was fixed immediately.

6.4 – Installation of the Software and Maintenance

Installation of the software was carried out by the author in the P & Co Leeds office, the other two branches will also install the software soon by the P & Co technician. The process of installation included zipping the database to a floppy disk and then installing it onto the P & Co server. To ensure the database could be reinstalled if one storage medium was corrupted, back-ups of the database were made to a floppy disk and this was kept by the owner in P & Co. Since the author will not be able to perform maintenance of the database, the P & Co technician has agreed to take on this role.

6.5 – Users Training

Training the users is essential in ensuring that the database can and will be used by them. Given that P & Co are very busy with their accountancy role, a one day training had been provided in the Leeds office based on using the database itself rather than given as a lecture. This ensures users are comfortable with using the database quickly and become familiar with it. A detailed user manual, as

shown in *Appendix L, Page 74-85*, had been given to the technician so that he can also train the other staffs in Manchester and London.

6.6 – Evaluation

The criteria that was used to evaluate against the database is based on McCall's studies *[29]*. His guidelines focus on the three important factors in terms of functionality, usability and maintainability of a software product, which is very relevant to any system development like this. Therefore, it had been chosen to be the evaluation criteria. The guidelines of the criteria and the findings from the evaluation are as follows:

1. Correctness – Extent to which the database fulfil the user requirements.

All the requirements defined from the users with additional enhancements such as, the production of VAT report, are being included in the database and all the users were pleased that their requirements are met. Thus, the database fulfils the user requirements and meets the correctness criteria.

2. Reliability – Extent to which no errors are found in terms of the functions of the database.

The data input, find, edit and delete was examined thoroughly for each form by the author and the users. The database operated properly during testing and all data was stored accurately.

3. Efficiency – Extent to which the database improves on current paper-based system.

The database efficiency is determined by the '*time taken to complete a task'*, the two users were timed to record the time it took to complete a task in the paper-based system and using the new database. Two tasks were given to the two users, **1)** input client details and **2)** find a clients' VAT registration number. The users were enquired to complete each task in the paper-based system first and then using the new database. The results are indicated in Figure 6.1.

	Owner in P & Co		Administrator	
Take taken using:	**Task 1**	**Task 2**	**Task 1**	**Task 2**
Paper-based system	6 mins	5 mins	4 mins	4 mins
New Database	4 mins	2.5 mins	2 mins	2 mins

The owner in P & Co spent most of the time meeting clients and dealing with accountancy matters rather than doing the administration work, therefore he took longer to finish these two tasks then the administrator. However, the time taken to complete both tasks using the new database is shorter which shows an improvement on the paper-based system. The administrator who has a basic knowledge in using database and is familiar with the paper-based system completed the two tasks by halving the time. The results show that overall the users perform quicker to input and find records with the new database.

4. **Integrity** – Extent to which only authorised access to the database is allowed.

A login password is required every time the users use the database to avoid unauthorised access. This was tested by the author and the users; the results show that the database will disallow invalid passwords.

5. **Usability** – Extent to which the database is functional and usable.

The usability of the database is determined by three measurements, *'number of times the interface misleads the users'* and *'number of favourable/unfavourable user comments'*. By testing the database, the user had not been mislead by the fields because they were familiar with them. For example, for the field *'National_Ins_No'*, they would know that it means *'National_Insurance_No'*. The number of unfavourable comment was one, which is the form *'Client Sales ledger'* could have been designed in a different way but they said it is still acceptable. However, the numbers of favourable comments made in total by the users far exceed the unfavourable comments which were five, such as *'the navigation is easy to follow'* and *'it is much quicker and easier to find a record using the database'*. Overall, the database was approved by the users that it is functional and usable.

6. **Maintainability** – Extent to which errors or parts require changing can be spotted easily by the maintenance staff.

The P & Co IT technician, who is responsible for maintenance, was asked to confirm that the field names used in the database is understandable later for maintenance purposes. He confirmed that they are understandable.

7. **Performance** – This is the assessment of response time and accuracy of the database.

Ten records were filled in each table in the database by the author in order to test the response time and accuracy of the database. They were then being searched, edited or deleted and the average response time for each task was two seconds with accuracy. The results show that the database performance is in an excellent condition.

The overall results of evaluating the database are favourable. The database fulfils the user requirements and the reliability is 100%. Efficiency and performance are achieved to allow users to operate much quicker in using new database than the paper-based system. Access is restricted to authorised users only. Several favourable comments were received from the users regarding the usability of the database and the staff have already been using the new database to store client and sales records in Leeds office. The other two branches will also start using the database soon. In terms of maintenance issue, the technician found it easy to maintain because he has already got knowledge in using Access and the field names are made clear to him. It can therefore be summed up that the database meets the evaluation criteria and has been deemed a success by both the author and the user.

Chapter 7 – CONCLUSION

7.1 – Introduction

The preceding chapters featured the process of meeting the project's main aim of developing a database for P & Co Chartered Accountant. This chapter vitally assesses the project aligned with the minimum requirements outlined in Section 1.2. It also includes possibilities of the further enhancements that could be implemented to the database and a overall conclusion of this project.

7.2 – Project Evaluation

The implemented system successfully meets the initial overall project aim which is, *'To implement a client and sales record management system for P & Co Chartered Accountant'*. The minimum requirements are also being fulfilled and the extent to which they were met are described in the following:

- **To replace the paper-based system in P & Co Chartered Accountant to a computerised system** – This is achieved by building a new database to replace the current paper-based system in P & Co Chartered Accountant.

- **To gather user requirements from the owner in P & Co; in short, to obtain a clear view of the problem on the existing paper-based system –** To obtain a clear view of the problem existed in P & Co, interviews with the P & Co owner and research were conducted on the background and current paper-based system procedures in P & Co Chartered Accountant. While problems were defined, they are then being analysed using suitable analysis tool to aid in devising the user requirements.

- **To meet the user requirements and to recommend a new system by introducing database to store client and sales records –** The user requirements were met by building a database since the development will involve a large amount of related data. Some enhancements had also been met which is to produce the VAT report. In order to meet the user requirements efficiently and properly, research was conducted into various software development methodologies to guide me through this development. Research on the selection of different database software tools was also being carried out to choose the most appropriate tool to use for this development. The choice was influenced by the software that P & Co had available to them.

- **To design and implement a database holding information about P & Co customers and their accounts details –** To design the database correctly involved investigating and applying database

design theory, the resources was found either from previous database modules notes, textbooks or online. HCI issues were also investigated and applied in order to design an easy-to-use interface. Implementation was aided by reading Access textbooks, such as, *Access 2002 VBA Handbook [27]* and asking friends who have knowledge in building a database in Access.

- **To evaluate the system –** To ensure the database is usable, functional and of good quality, evaluation criteria were examined to decide on the most suitable criteria to employ in this development. The evaluation criteria were successfully applied to evaluate the database quality and so, this requirement was achieved.

7.3 – Further Enhancements

Although the system produced achieved the users requirements and the users were very pleased with the database overall, some changes and enhancements could also be made to the database to improve its functionality and usability. Some of the enhancements have already been implemented while the rest will need to be made in the future due to the time limitations. The enhancements are outlined below:

- An up-to-date printout of monthly VAT report was implemented so that the staff will no longer need to prepare the VAT taking sheet every month.

- The database can be shared across three branches by installing a database server in two of the branches. Then the database should be installed in the database server to allow sharing inside the branch. Active Server Page codes can then be written to post the database online. An Active Server Page (ASP) is an HTML page that include small embedded programs that are processed on a Microsoft Web server before the page is sent to the user *[30]*. It allows database to be retrieved online with login password to restrict public access.

  Backup of the database will have to be made every night in two of the branches and the backups should be kept privately by the seniors. This is done in order to avoid sudden accidents such as fire, when one of the backup is burnt, there is still another that can be reinstalled and used. The follow up of the networking of the database will be carried out by the P & Co technicians since he has strong knowledge in networking.

- A function button ‘Produce Invoice’ have been added in the interface forms ‘Bookkeeping’, ‘VAT’, ‘PAYE’ and ‘Tax’ so that when the service is completed, the staff can just click the ‘Produce Invoice’ button to create sales invoice for the client.

- A function button 'Find Balance' has been added in the interface forms 'Client' so that when client request for the account balance, the staff can just click on the button to retrieve information from the 'Client Sales Ledger' form.

- The print button on the 'Sales' and 'Payment' forms print **all** invoices, not just the invoice for the selected record, sample of the invoice and receipt can be found in *Appendix G, Page 59-60*, which is the same as the interface forms. Therefore, this needs to be altered to ensure only the selected record is printed and there is a proper layout.

- Again, the print button on the 'Client Sales Ledger' form and 'Client Sales Summary Report' prints **all** the clients account details, not just the account details for the selected client, therefore these needs to be altered to ensure only the selected client is printed.

- The design of the 'Client Sales Ledger' form can be redesigned but since it is still acceptable by the client, the author has not spent any more time on this.

7.4 – Project Conclusion

To conclude, the new system met all the minimum requirements identified in the project as well as the project objectives. By following the *'Waterfall model'* methodology produced comprehensive and high quality reports throughout this project development. Client involvements was also being ensured at all times which enabled any amendments to be made immediately to suit the client's performance.

The project was deemed to be a success for all parties concerned. The client was very pleased with the database and gained a working system for his company that met his requirements while saving him the financial cost of contracting a software developer to produce the system. The client also verified that he had gained an interesting insight into the skills, requirements and technique to generate a Information System.

Reference:

[1] Softwarecasa: ScanSoft PaperPort Deluxe 8.0
URL: http://www.softwarecasa.com/products/p11260.htm?source=goo [27th January 2003]

[2] Best: Sage Software
URL: http://www.sota.com/configurator/index.cfm?action=s1 [27th January 2003]

[3] Avison D.E., and Fitzgerald G, (1995), **Information Systems Development: Mehtodologies, Techniques and Tools**, McGraw-Hill.

[4] Cotterell, M and Hughes, B, (1999), **Software Project Management, 2nd Edition**, McGraw-Hill.

[5] Martyn Clark, (2000), **Introduction to Information Systems: Module Summary Notes**, University of Leeds, School of Computing.

[6] Owen Johnson (2001), **Object Orientated Analysis and Design: Module Notes**, University of Leeds, School of Computing.

[7] Elmasri, R and Navathe, S (2000), **Fundamentals of Database Systems, 3rd Edition**, Addison-Wesley.

[8] A Comparison of SQL Server 2000 with DB2 v8.1
URL: http://www.databasejournal.com/features/mssql/article.php/10894_2177441_3 [10th February 2003]

[9] The Comparison of SQL Server 2000 with Oracle9i
URL: http://www.databasejournal.com/features/mssql/article.php/2170201 [10th February 2003]

[10] Access97/2000 and SQL Server 7/2000 Compared
URL: http://databasejournal.com/features/mssql/article.php/1442781#part_2 [10th February 2003]

[11] Microsoft Access Database Design
URL: http://www.clearform.com/microsoft_access.htm [10th February 2003]

[12] Oracle licenses at discount prices
URL: http://www.pro-dba.com/pricing.html [12th February 2003]

[13] Neil Pike (2000), **SQL Server: Common Problems, Tested Solutions**, Apress.

[14] Compare Prices for MICROSOFT SQL SERVER 7 in Programming Tools , Productivity Applications or Internet and Communication
URL: http://www.dealtime.com/xPP-Programming_Tools~KW-MICROSOFT_SQL_SERVER_7~CR-1
[12th February, 2003]

[15] Microsoft Access 1997, Sanjaya Hettihewa
URL: http://www.exam-ta.ac.uk/access97199b.htm [12th February 2003]

[16] Hawryszkiewycz, I.T. (1994), **Introduction to systems analysis and design, 3rd Edition**, Prentice Hall Australia.

[17] Barry Eaglestone, (1991), **Relational Databases**, Stanley Thornes (Publishers) Ltd.

[18] Chapter 3 – Entity Relationship Model, Susan Gauch, with some slides by David L. Spooner, Rensselaer Polytechnic Institute
URL: http://www.ittc.ku.edu/~sgauch/647/s00/notes/Ch3/sld003.htm [22nd February 2003]

[19] Mott P, and Roberts S (1999-2000), **DB11: Introduction to Databases, Module Notes and Coursework**, School of Computing, University of Leeds.

[20] Databases Glossary
URL: http://databases.about.com/library/glossary/bldef-primarykey.htm [3rd March 2003]

[21] Databases Glossary
URL: http://databases.about.com/library/glossary/bldef-foreignkey.htm [3rd March 2003]

[22] Roberts, SR, (2001), **Database Principles and Practice: Module Course Notes**, University of Leeds.

[23] Teorey, TJ, (1999), **Database Modelling and Design**, Morgan Kaufmann Publishers.

[24] Kirakowski Jurek, (1988), **Human Computer Interaction: from voltage to knowledge**, Chartwell-Bratt.

[25] Jenny P, Yvonne R, Helen S, David B, Simon H and Tom C (1994), **Human-Computer Interaction**, Addison Wesley.

[26] Faulkner Christine, (1998), **The Essence of Human-Computer Interaction**, Prentice Hall.

[27] Susann Novalis and Dana Jones, (2001), **Access 2002 VBA Handbook**, Sybex.

[28] Arthur M.Langer, (1997), **Analysis and Design of Information Systems, 2nd Edition**, Springer.

[29] McCall J., Richards P., and Walters G., (Nov 1997), **Factors in Software Quality, Vol. I**, AD-A049014/015/055.

[30] Active Server Page - a searchWin2000 definition
URL: http://searchwin2000.techtarget.com/sDefinition/0,,sid1_gci213787,00.html [20th April 2003]

Appendix A – Personal Reflection

By reviewing the duration of this project, this has proved to be a challenging but rewarding experience. The personal objectives identified for this project (as outlined in Summary) have been fulfilled, much of the work completed during my studies at University were reviewed, skills were gained in project development and in database design and implementation, resulting in a product that the P & Co staff were pleased with and useful to the organisation.

The two greatest challenges I found in this project were the database design and implementation due to the lack of experience. As a result, more time had been spent in these two phases. However, more than the minimum functions were implemented at the end and this has been the part of the project I have found most valuable due to the practical skills gained. Also, I found it very effective and beneficial by following the waterfall model since it provides a framework for developing the project.

Given the many other coursework commitments, it is very difficult to set specific targets to keep the project progress on course. However, through planning a realistic project schedule was of great help, ensuring the tasks were completed on time to prevent a build up of work. This has assisted me to realize the significance of the project management. Contacting the owner in P & Co was another problem encountered during the project since he was always busy at work, which led to the delay of confirmation of some requirements. Arranging an earlier appointment with him would have been helpful to solve this.

If I have another opportunity to complete this project again, I would ensure the database is designed correctly before the implementation since much time was wasted on the correction of the database design and implementation. I would also start this project earlier than I did.

The following is the advice I would give to students embarking on a similar project:

1. Start early and do not underestimate the time it takes to design and implement the database as well as its interface, especially for those who need to learn the database software tool as I did.
2. Devising a realistic project schedule is vital to guide you through the project; taking into consideration that things will most likely need more time than you expect.
3. Avoid getting frustrated with the amount of work to be completed, keep on working and maintain the productivity.

In general, I am very pleased that I did this project because of the practical skills I gained in database design, implementation and project management that I previously did not have.

Appendix B – Project Schedule (Outdated)

Semester 1

Week: Ending	04/10	11/10	18/10	25/10	01/11	08/11	15/11	22/11	29/11	06/12	13/12	20/12	27/12	03/01	10/01	17/01	24/01
1st Semester	1	2	3	4	5	6	7	8	9	10	11						
Phase 1																	
Initial Research																	
Requirements Gathering																	
Requirements Analysis																	
Write Up Report																	
Mid-Project Report																	
Phase 2																	
Functional Dependency Analysis																	
Christmas Vocation																	
Exam Preperation																	
Semester 1 Exam Period																	

Semester 2

Week: Ending	31/01	07/02	14/02	21/02	28/02	07/03	14/03	21/03	28/03	04/04	11/04	18/04	25/04	02/05	09/05
2nd Semester	1	2	3	4	5	6	7	8	9					10	11
Phase 2															
Design Database															
Implement Database															
Test Database															
Phase 3															
Design Data Entry Interface															
Implement Data Entry Interface															
Test Data Entry Interface															
Progress Meeting															
Phase 4															
Networking Design															
Easter Vacation															
Phase 5															
Evaluate System															
Phase 6															
Write Up Report															

Key

Term Time	Project Work	Vacation	Revision	Exam Period

Appendix C – Project Schedule (Revised)

Semester 1

Week Ending	04/10	11/10	18/10	25/10	01/11	08/11	15/11	22/11	29/11	06/12	13/12	20/12	27/12	03/01	10/01	17/01	24/01
1st Semester	1	2	3	4	5	6	7	8	9	10	11						
Phase 1																	
Initial Research																	
Requirements Gathering																	
Requirements Analysis																	
Write Up Report																	
Mid-Project Report																	
Phase 2																	
Functional Dependency Analysis																	
Christmas Vocation																	
Exam Preperation																	
Semester 1 Exam Period																	

Semester 2

Week Ending	31/01	07/02	14/02	21/02	28/02	07/03	14/03	21/03	28/03	04/04	11/04	18/04	25/04	02/05	09/05
2nd Semester	1	2	3	4	5	6	7	8	9					10	11
Phase 2															
Normalisation															
Design Database															
Implement Database															
Client Sales Summary Report															
Phase 3															
Design Data Entry Interface															
Implement Data Entry Interface															
Test Data Entry Interface															
Progress Meeting															
Phase 4															
Enhancement if time available															
Easter Vacation															
Phase 5															
Evaluate System															
Phase 6															
Write Up Report															

Key

Term Time	Project Work	Vacation	Revision	Exam Period

Appendix D – Correspondence with the User

Minutes of Meeting with Jonathan Pitayanukul

Date	4th November 2002
Present	Ka Man Chan (KMC) Jonathan Pitayanukul(JP)
Location	P & Co Chartered Accountant

KMC asked JP if he could describe the current paper-based procedures and problems in P & Co Chartered Accountant.

JP said that there is seven different client details need to be stored in the client folders and the details of the procedures and problems are summarised as below:

Client and Business Details

- A new paper file is created to record the new client and business details. Additional correspondence is also filed into it, for instance, their bookkeeping details and tax details etc. The file is then place in one of the filing cabinets according to the business identification number.
- If the staffs wish to find specific client or business details, he/she will search through the specific cabinet which contains the file. While the information is requested from other branches' staff, the staff will need to ring or fax the required information back to reply. Occasionally, client files may be lost and therefore cannot find the required information.
- Updates and deletions of clients or business details are always being written on paper first until the staffs have time to find the specific client files to make the correction. However, the staffs sometimes forget to or lose the paper which cannot make the amendment.

Bookkeeping Details

- Clients who make less than £55,000 profit a year in their business need a bookkeeping service in every three months. The staff who collects the thirteen weeks' bookkeeping taking sheets (the sheet which records each weeks' profits and expenses) and purchase invoices from a client will record the date the bookkeeping is brought in and completed as well as who received it on a piece of paper as a record. The staffs may occasionally forget to or lose the paper that cannot keep track of the record. In the worst case, the bookkeeping may be left uncompleted until the client ring up and check the process.

VAT Details

- Clients who make more than or equal to £55,000 a year in their business need a VAT service in every three months. There is a paper prepared every month by the staff to record which client needs to complete the VAT and when the client has brought the VAT taking sheets (but sometimes the paper prepared may be out-of-date) in. The staff must make sure all the clients have brought the VAT in by fifteenth of every month for the accountant to work on since there will be a penalty from HMCE if the payment is not posted in by the end of the month. Again, the staff may sometimes forget to or lose the paper that cause problems (e.g: penalty).
- As mention in Section 1.4 and 1.5, works are now transferred across three branches, while client rings up to check the process of the VAT workings, the staff will have to ring the responsible branch for the process and then reply back to the client. This process is very time-consuming, which will delay the job that the staff is working on.

PAYE Details

- PAYE means 'pay as you earn', the employees and employers have to pay national insurance contributions (NIC) and tax on what they earn. Manchester office is now responsible for all three branches' PAYE services, the accountant who deals with it always need to request other branches' staff to look for information in order to complete the work. Details of PAYE information is placed in another block of filing cabinet which the staff will need to search through the whole filing cabinet. This is again very time-consuming.

Tax Details

- Each person who is self-employed needs to review his/her tax situation every year. London office is now responsible in completing self-assessment tax return as the specialist accountant is situated in London. Thus, Manchester and Leeds offices' staff always need to find the required information for the London accountant to complete the work.

Client Sales Ledger Details (same as Clients' Sales and Payment Record Details)

- Only London and Manchester offices have the system to record clients' sales and payments while Leeds clients' accounting system is being kept in Manchester office. Therefore, once a service is provided to the client, or the client has made a payment, the staff in Leeds office will need to inform Manchester either by phone or by fax for updating the clients' sales and payment records.
- If a client rings up to Leeds office checking how much he/she has owed P & Co, the Leeds' branch staff will then ring up the Manchester branch to get the required information which may cause low workload while there are many requests from clients.

- There are no standard sales invoices and payment invoices in the current system means that the staff will need to produce invoices every time they charge client or receive payment from client.

JP stated that a summary report of the clients' sales record would be wanted in order to calculate the average amount P & Co earns from each client.

KMC then explained to JP that she would need to analyse the above details and problems in order to produce the user requirements (what should be included in the system) for him to approve and sign before the design of database.

JP agreed and KMC has arranged another meeting with JP for the approval of user requirements.

Appendix D – Correspondence with the User

Minutes of Meeting with Jonathan Pitayanukul

Date	**4th November 2002**
Present	**Ka Man Chan (KMC)** **Jonathan Pitayanukul(JP)**
Location	**P & Co Chartered Accountant**

KMC brought the user requirements for JP approval before she builds the client and sales record management system.

KMC explained to JP that since the time given for this project is very limited, it would not be able to implement the networking of the database.

JP agreed on the user requirements and said that if KMC does not has enough time to finish, she can leave the networking part of the database because there will be a computer technician coming in from Taiwan in March 2003 and he will be able to follow up for the networking of the database.

After the agreement on the above user requirements, JP added that the database should be restricted to authorised users only. Also, he would like to have a simple and ease of use user interface that can operate in real time to enable the administrator or accountant to enter details whilst on the phone.

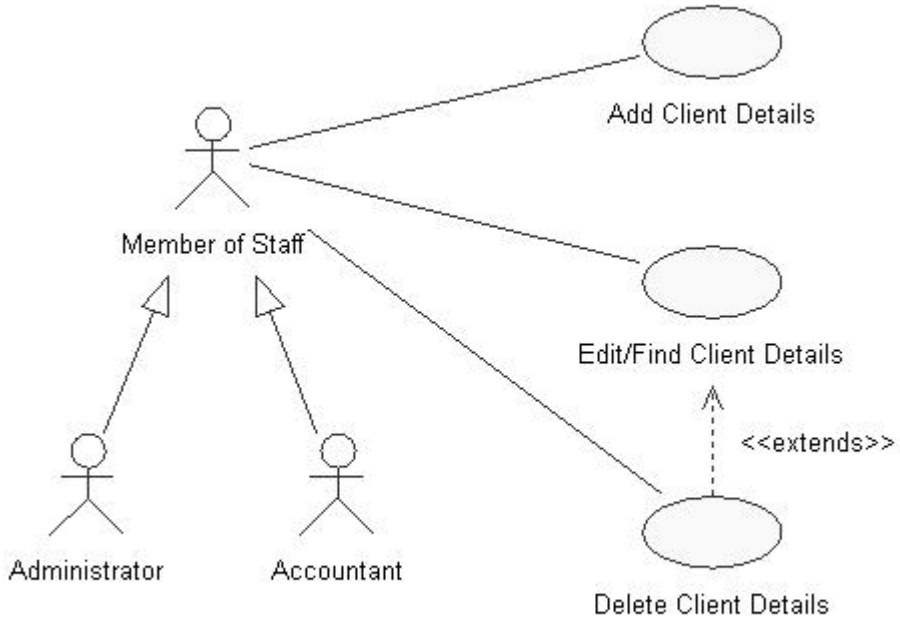
KMC asked JP how much he would be willing to spend in this project, how many people in P & Co would be using the database at one time and is there a lot of data entry in one day?

JP replied since he would be investing money in updating the companies PCs, monitors and purchasing the ScanSoft PaperPort, his budget for this development would be £200. He also replied that there is not a lot data entry in one day, the maximum is twenty transactions per day and the maximum concurrent access is eight people (as some employees are part-time).

KMC thanked JP for the time for the interviews and has agreed by JP that he would test the system once the system is completed.

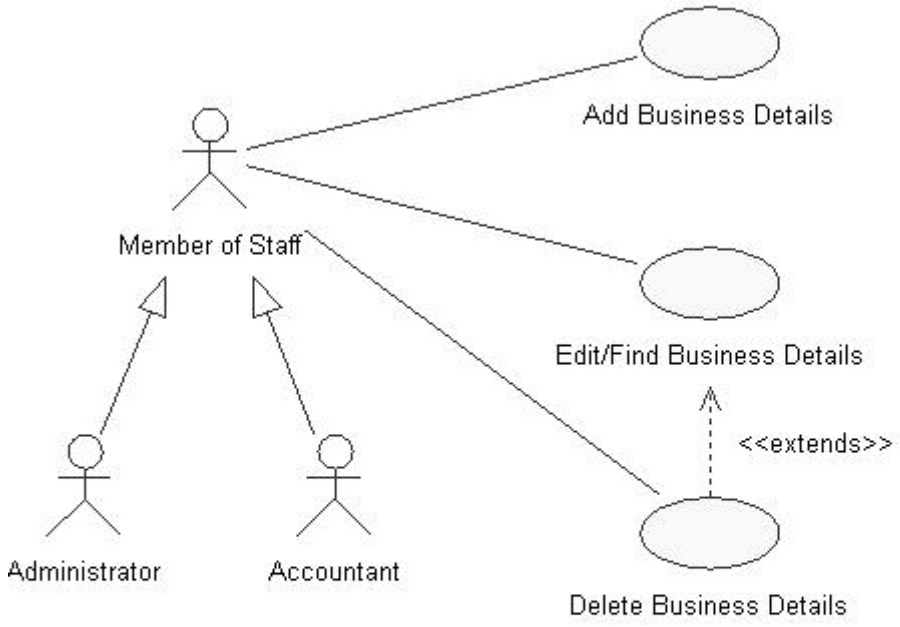
Appendix E – Use Case Diagram

Client Part of the Database



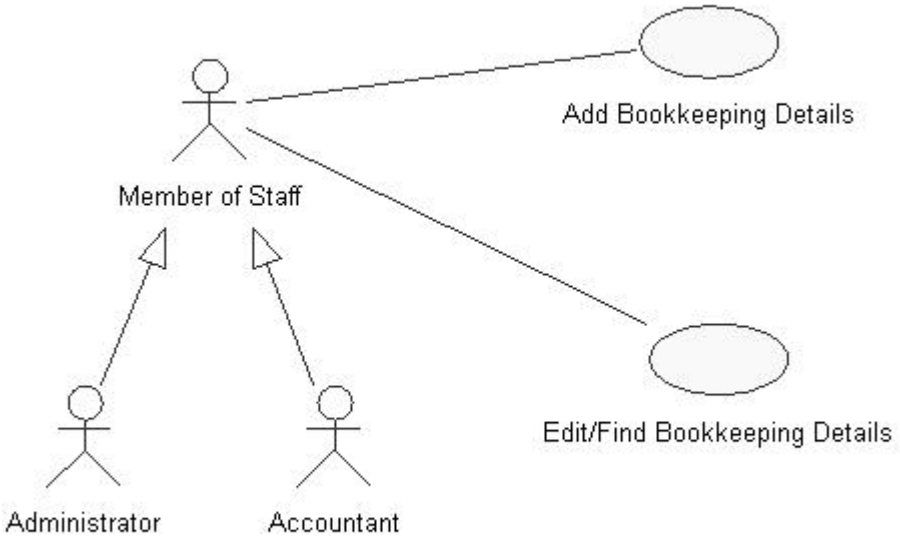
Appendix E – Use Case Diagram

Business Part of the Database



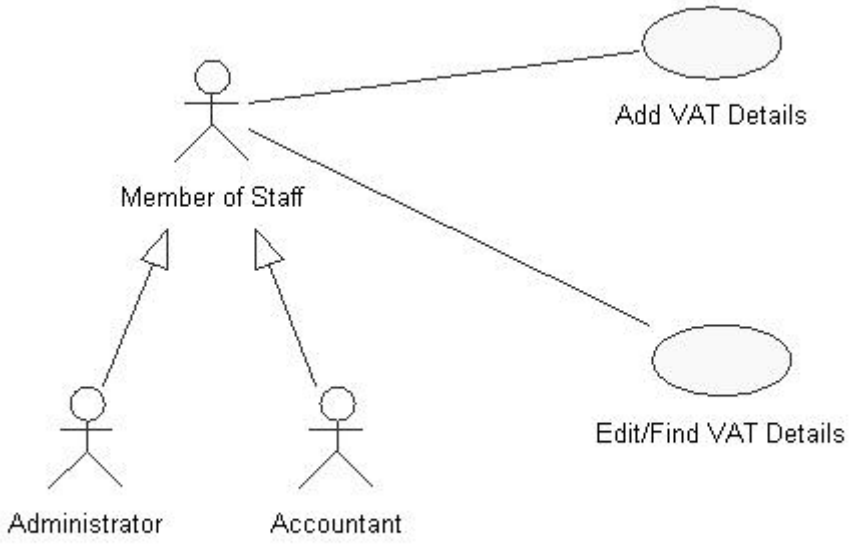
Appendix E – Use Case Diagram

Bookkeeping Part of the Database



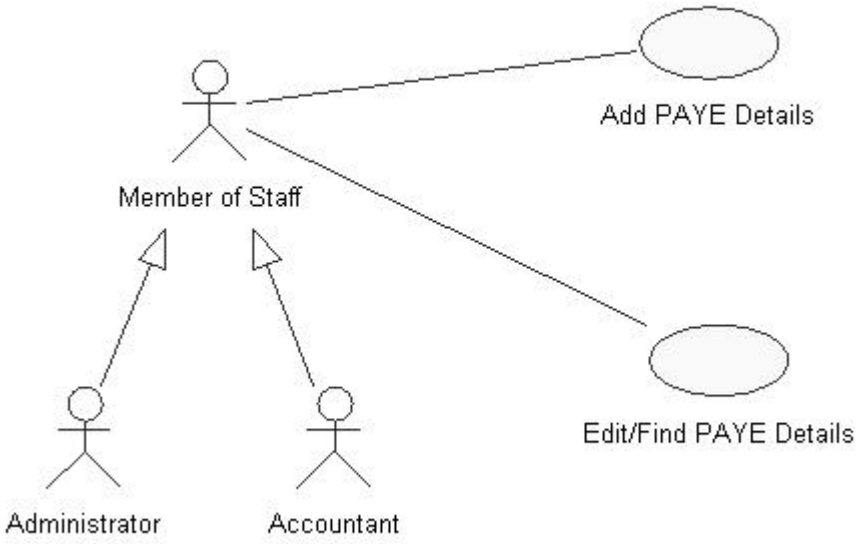
Appendix E – Use Case Diagram

VAT Part of the Database



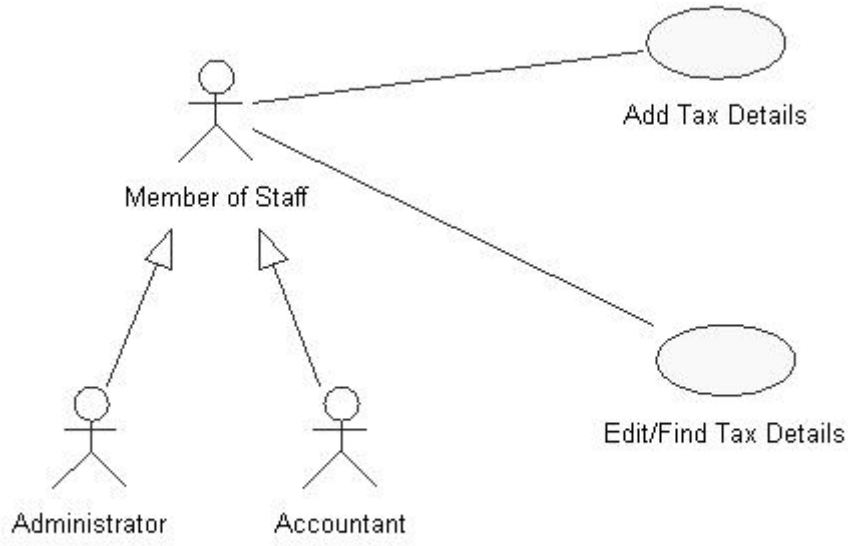
Appendix E – Use Case Diagram

PAYE Part of the Database



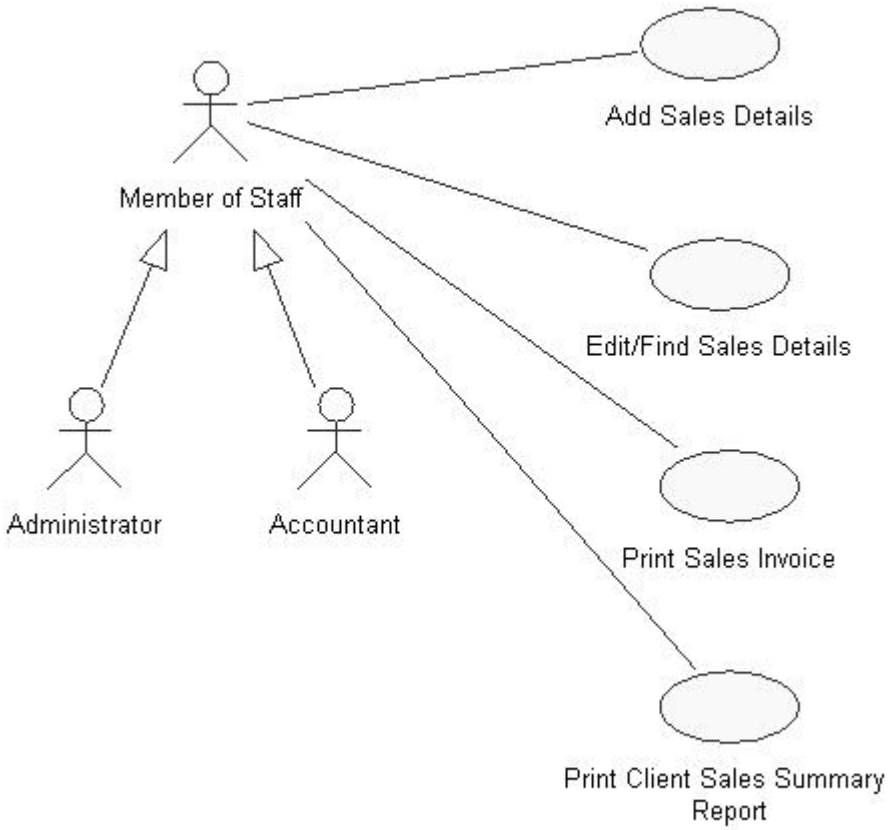
Appendix E – Use Case Diagram

Self Assessment Tax Return Part of the Database



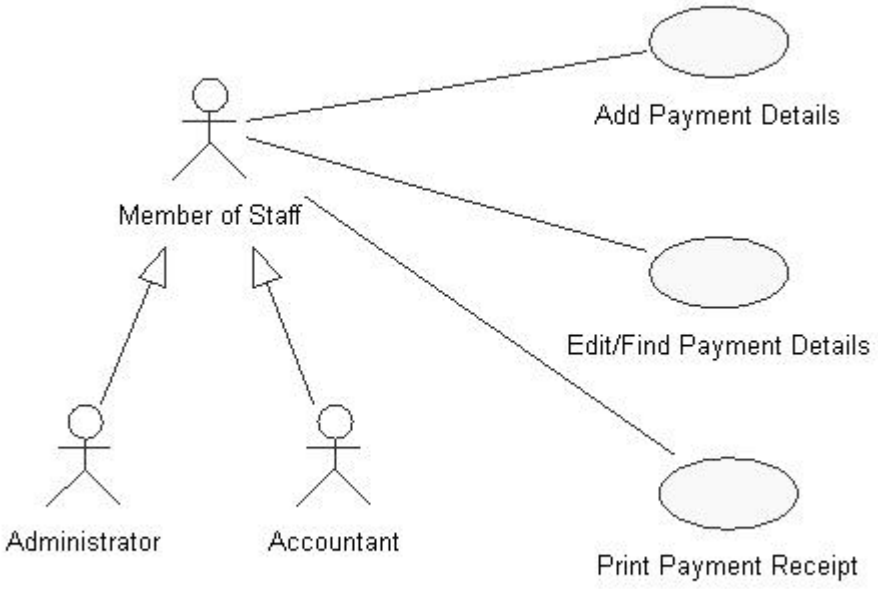
Appendix E – Use Case Diagram

Sales Record Part of the Database (Sales Invoice)



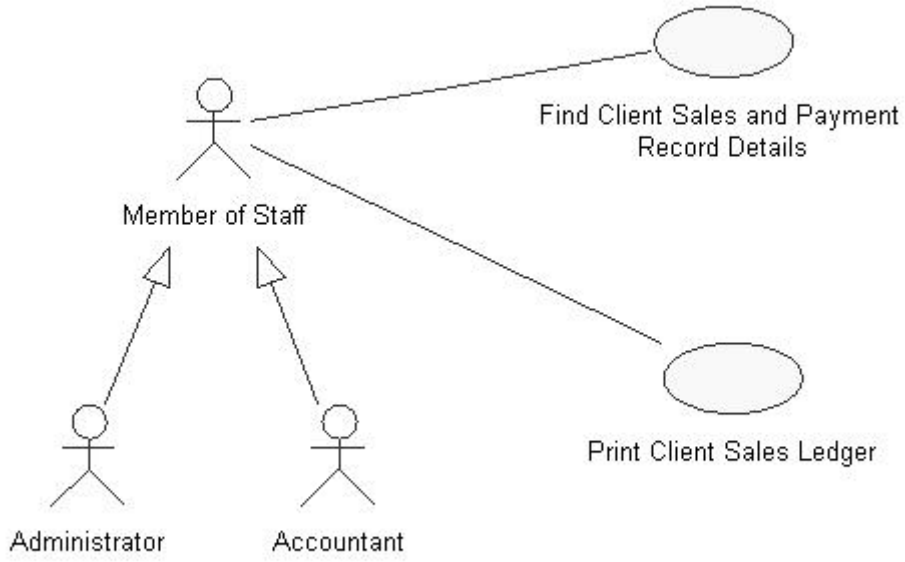
Appendix E – Use Case Diagram

Payment Record Part of the Database (Payment Receipt)



Appendix E – Use Case Diagram

Client Sales Ledger Part of the Database



Appendix E – Use Case Description

Use Case Specification: Add Client Details

Revision History

Date	Description	Author
03/Dec/02	Use Case Description was completed after the Use Case Diagram were drawn and agreed by the owner in P & Co	Ka Man Chan

Use Case Name:	Add Client Details
Summary:	The Member of Staff in P & Co receives an information sheet from a potential client, checks the details and inputs the client into the database.
Primary Actor:	Member of Staff
Other Actor:	None
Goal:	To add the client details into the database.
Basic Course of Events:	Use Case begins when the Member of Staff receive a client information sheet. The staff then checks if all the details are being filled in and then inputs into the database.
Alternative Paths:	1) Some of the details are being missed out that the client will have to bring it back and fill it in again. 2) Client details are already in the database, so checks are made to determine if the client has been already P & Cos' client.
Exception Paths:	None.
Trigger:	An information sheet is received from the Member of Staff.
Assumptions:	An information sheet is completed.
Pre-Conditions:	The client had identified the service(s) they require from P & Co.
Post-Conditions:	The client details are input into the database as a new client record.
Project:	"Client and Sales Management System for P & Co Chartered Accountant."
Author:	Ka Man Chan
Date:	03/Dec/02

Appendix E – Use Case Description

Use Case Specification: Find/Edit Client Details

Revision History

Date	Description	Author
03/Dec/02	Use Case Description was completed after the Use Case Diagram were drawn and agreed by the owner in P & Co	Ka Man Chan

Use Case Name:	Find/Edit Client Details
Summary:	The Member of Staff receives additional/different information about a client, the Member of Staff will search the database to retrieve information and make necessary changes to the client record as needed.
Primary Actor:	Member of Staff
Other Actor:	None
Goal:	To Find/Edit the client details into the database.
Basic Course of Events:	Use Case begins when the Member of Staff receives additional/different information about a client. The Member of Staff uses a range of key fields to search for the client in the database, finds him/her and makes the necessary changes.
Alternative Paths:	1) The changes may have already been made and thus no editing is required. 2) No changes are required, the Member of Staff is only retrieving client information. 3) Member of Staff receive a request for information about a client, the Member of Staff finds the client details from the database and reply back.
Exception Paths:	The client may not have been added in the database.
Trigger:	1) Request for client details to be changed is received by the staff member. 2) Request for retrieving client details from other branches' staff is received by the Member of Staff.
Assumptions:	The client is the member of P & Co Chartered Accountant.
Pre-Conditions:	1) The client or staff has identified the changes that need to be made and told the Member of Staff.
Post-Conditions:	Searching the client record successfully or changing the client record successfully.
Project:	"Client and Sales Management System for P & Co Chartered Accountant."
Author:	Ka Man Chan
Date:	03/Dec/02

Appendix E – Use Case Description

Use Case Specification: Delete Client Details

Revision History

Date	Description	Author
03/Dec/02	Use Case Description was completed after the Use Case Diagram were drawn and agreed by the owner in P & Co	Ka Man Chan

Use Case Name:	Delete Client Details
Summary:	This use case extends 'Find/Edit' when a client record is no longer needed, the Member of Staff then deletes it from the database.
Primary Actor:	Member of Staff
Other Actor:	None
Goal:	To delete the client details into the database.
Basic Course of Events:	Use Case begins after a Member of Staff found a client record. If there are no transaction records for the customer, the customer record maybe deleted.
Alternative Paths:	None.
Exception Paths:	Cannot be deleted if transactions records exist.
Trigger:	The Member of Staff wishes to clean up the inactive records in the database.
Assumptions:	Client record is no longer required.
Pre-Conditions:	The record has already been found.
Post-Conditions:	The client record has been successfully deleted from the database.
Project:	"Client and Sales Management System for P & Co Chartered Accountant."
Author:	Ka Man Chan
Date:	03/Dec/02

Appendix F – Use Requirements

1. Develop a database, which is able to store all details for Client, Clients' Business, Bookkeeping, VAT, PAYE, Tax, Sales Invoice and Payment.
2. Adding, finding, editing and deleting for a Client details must be possible.
3. Adding, finding, editing and deleting for a Clients' Business details must be possible.
4. Adding, finding and editing for the Bookkeeping details of the client business must be possible.
5. Adding, finding and editing for the VAT details of the client business must be possible.
6. Adding, finding and editing for the PAYE details of the client business must be possible.
7. Adding, finding and editing for the Tax details of the client business must be possible.
8. Adding, finding and editing for the Sales record details of the client or its business must be possible.
9. Adding, finding and editing for the Payment record details of the client or its business must be possible.
10. Finding a Clients' Sales Ledger details must be possible.
11. Able to print the Sales Invoice and the Clients' Sales Summary Report.
12. Able to print the Payment Invoice.
13. Produce an up-to-date printout of monthly VAT report, provided there is sufficient time available.
14. Design the architecture of the networking of the database provided there is sufficient time available so that all three branches can access the same database.

The above user requirements have been agreed by the users.

Appendix G – Sample of the User Interface Form

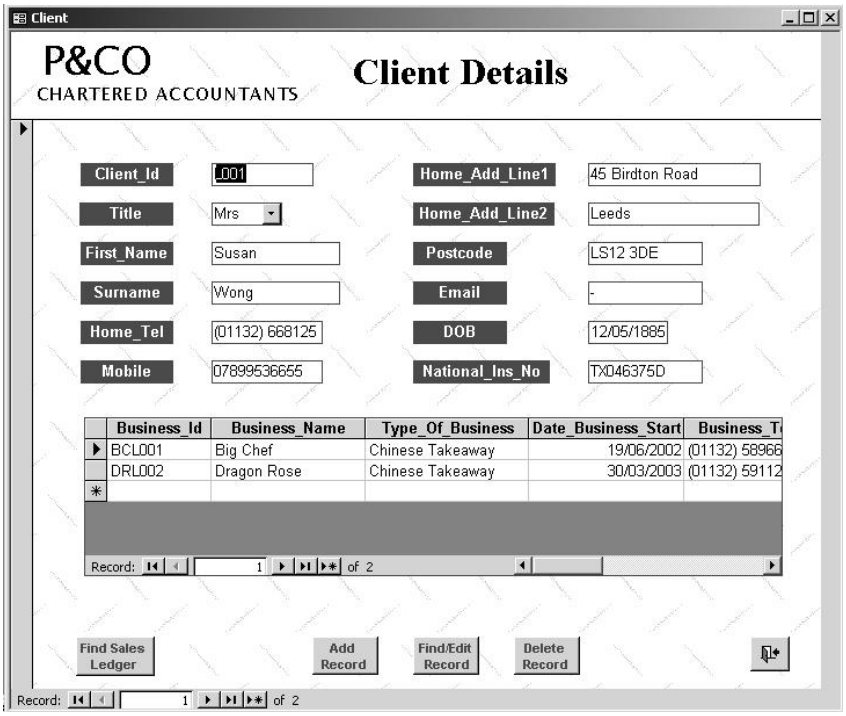


Figure 4.4: Client Details Interface Form

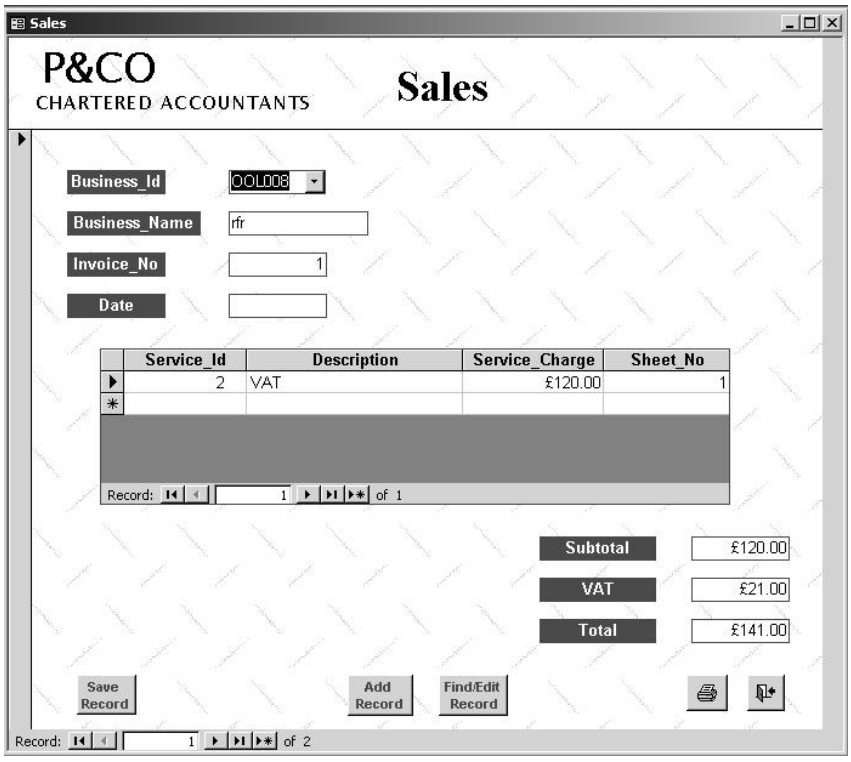


Figure 5.5: Sales Record Interface Form

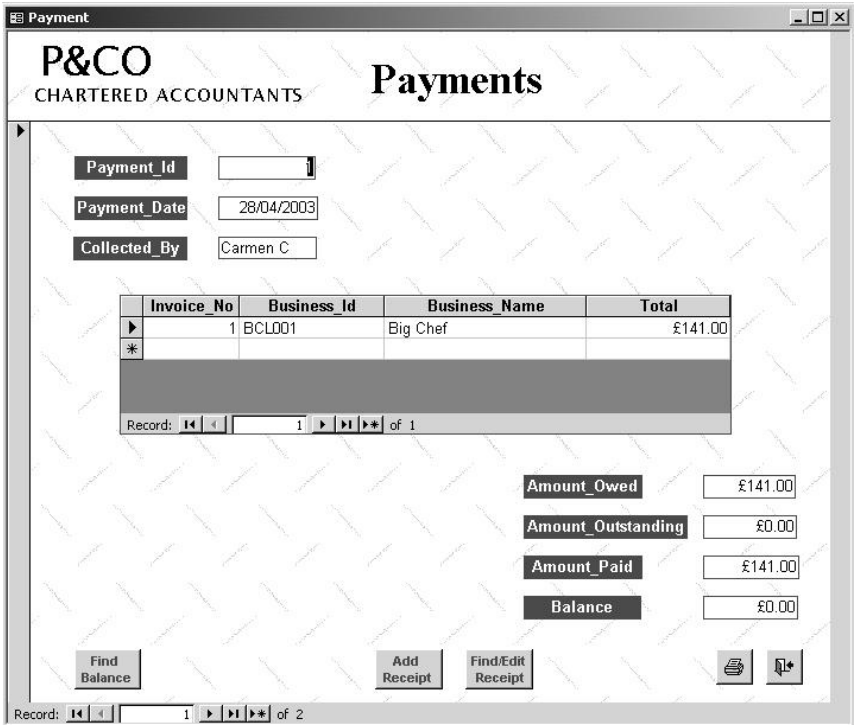


Figure 5.7: Payment Record Interface Form

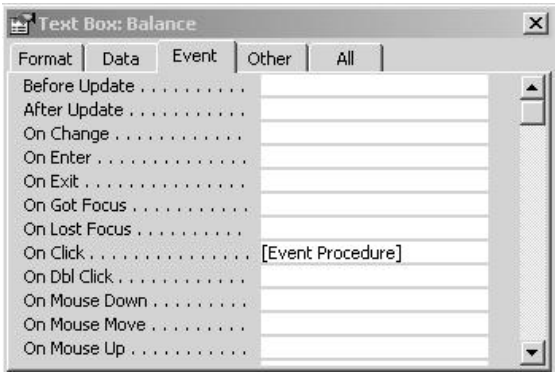


Figure 5.10: Access Icon Properties option

```
**********************************************
'  Function Balance

Private Sub Balance_Click()
Balance = Amount_Owed + Amount_Outstanding -
Amount_Paid
End Sub
**********************************************
```

Figure 5.11: VBA function: Balance

Appendix H - Features in Microsoft Access 2000

The functions of each tool in Microsoft Access 2000:

Tables

Tables are the most important tools in Microsoft Access. Its function is to store all the data associated with tables. Entities identified in the E-R diagram (as shown in Figure 4.1) are transformed into tables.

Queries

The function that queries perform is to retrieve and manipulate specified data. Five queries have been created in this database development to extract specific records.

Forms

Forms play a key role in most Access applications. All data entry and data review take place in forms *[27], p148.* It also allows the developer to generate a more professional and easier approach of data input and appearance. Forms have been used throughout the database development as the user data entry interfaces.

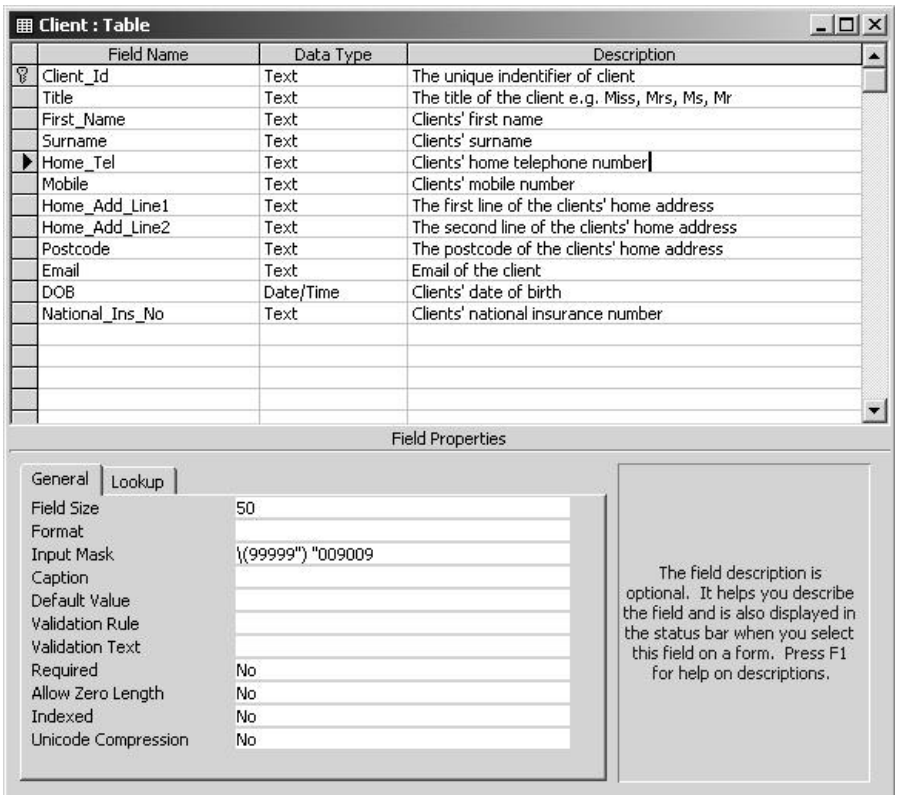
Reports

The report function is used to print information *[27], p2.*

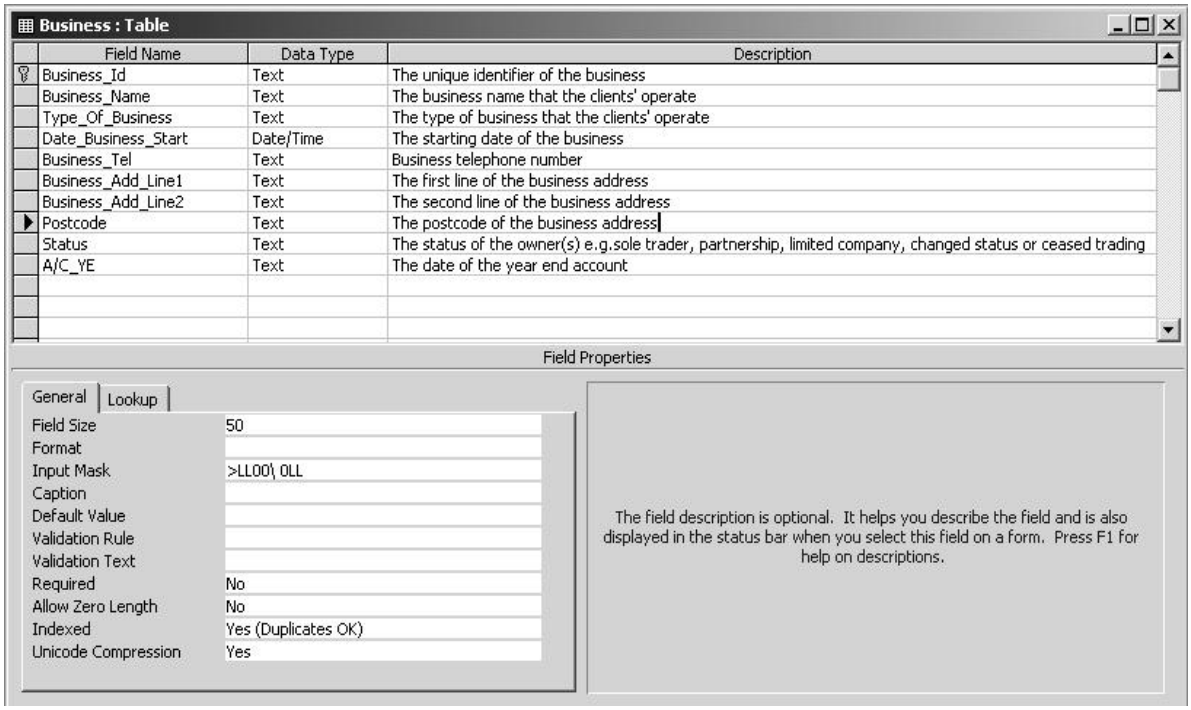
Macros

Using macros allows the developer to transform the interactive database into an automated database application.

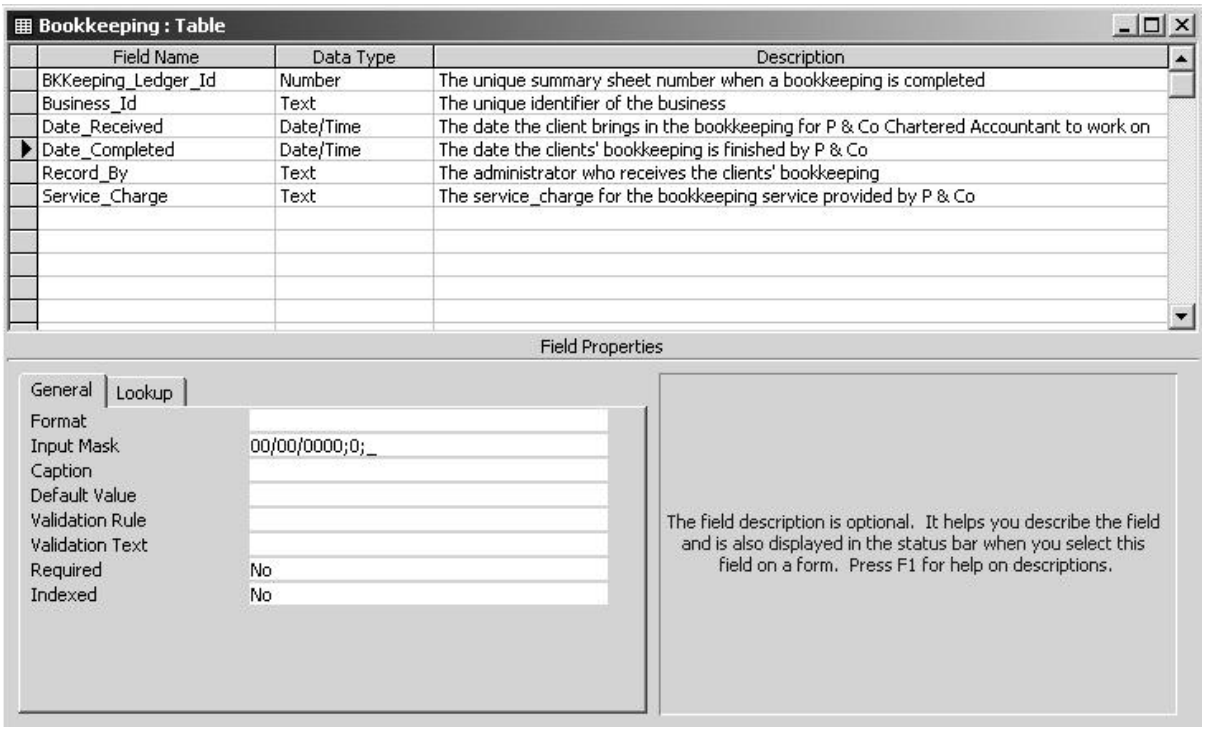
Appendix I – Design View of the Database System



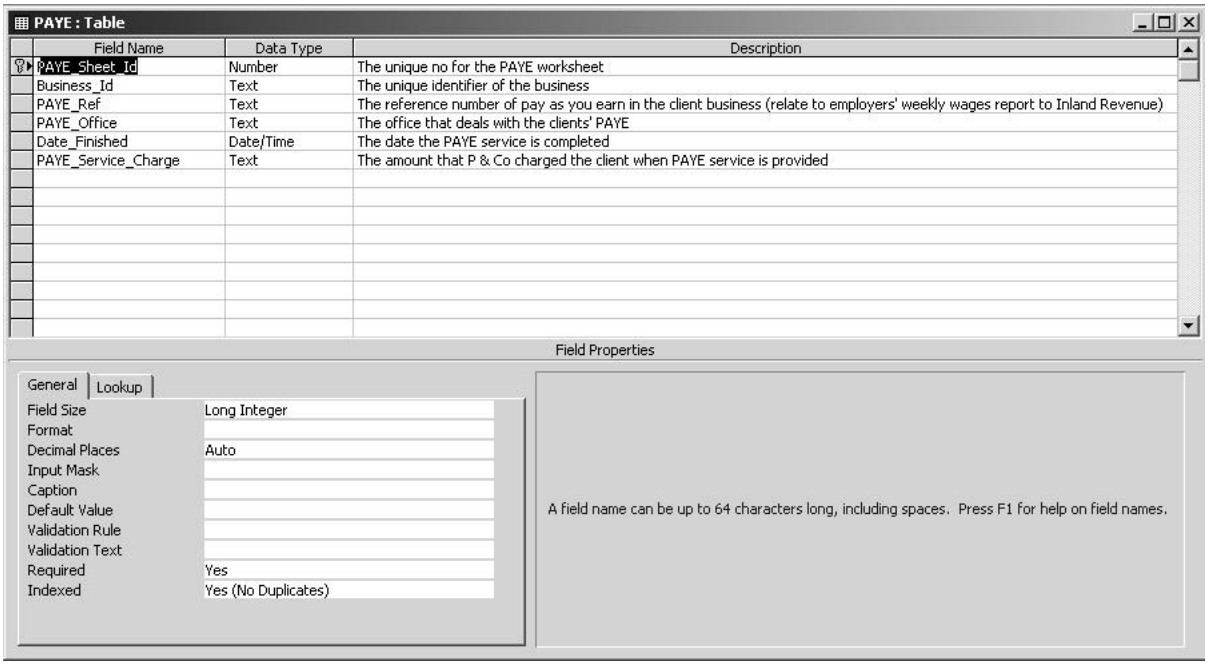
Client Table Design



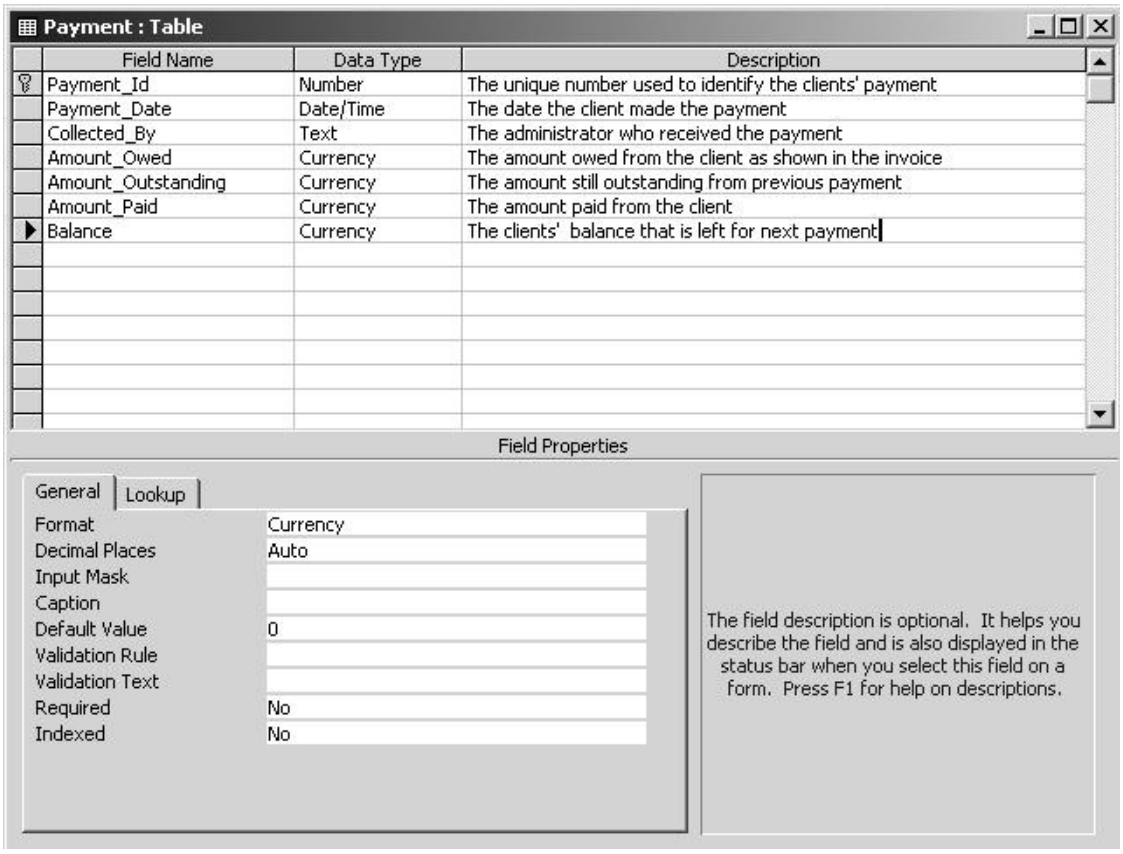
Business Table Design



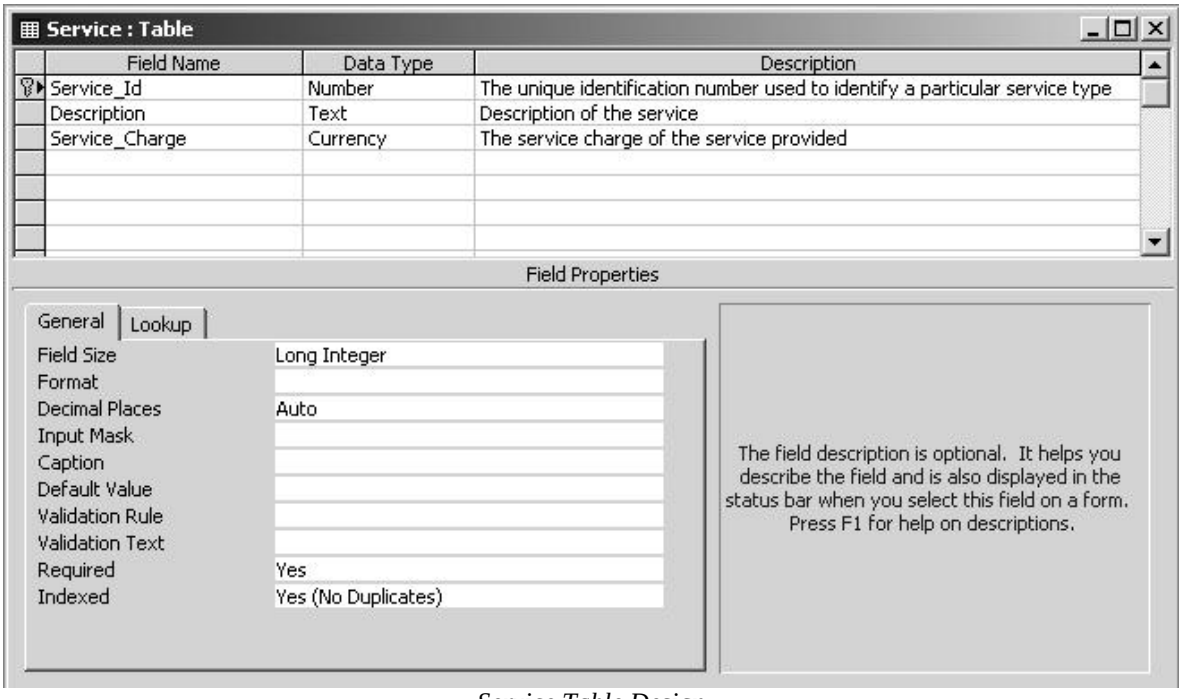
Bookkeeping Table Design



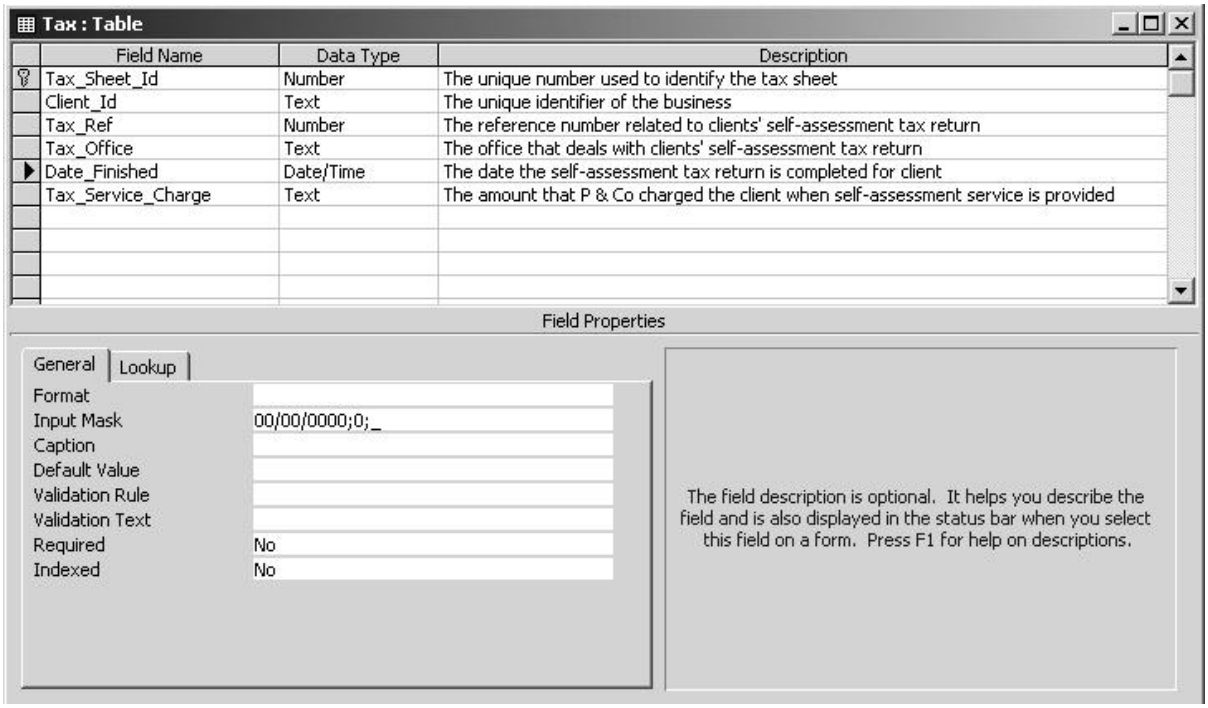
PAYE Table Design



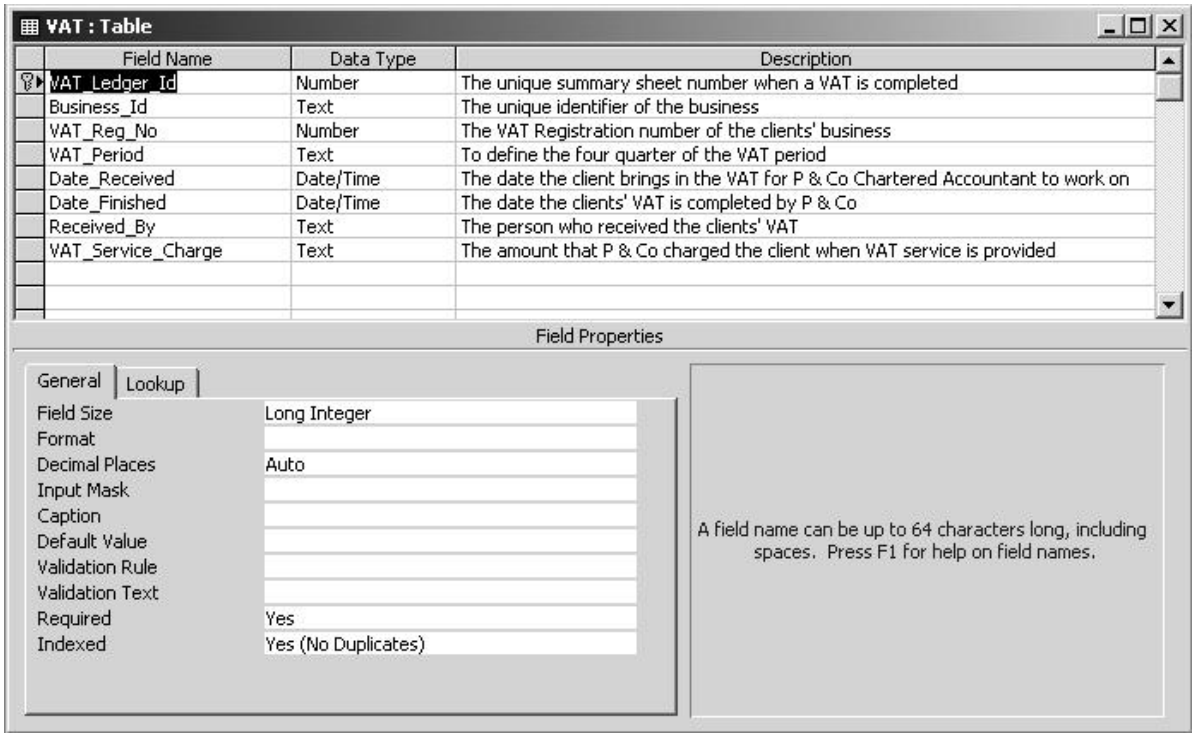
Payment Table Design



Service Table Design



Tax Table Design

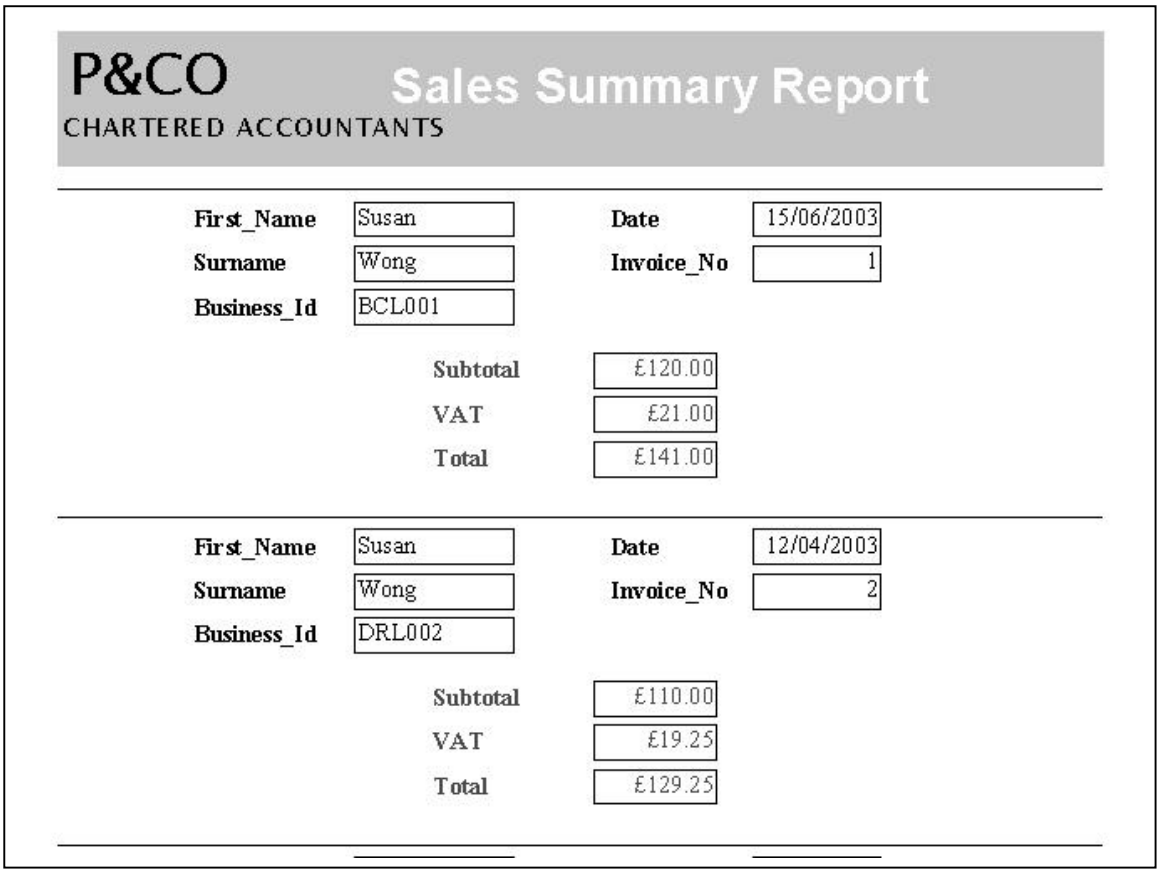


VAT Table Design

Note: The above associated with the Sales Table Design included in the Report are the nine main table designs in the P & Co Database.

Appendix J – Database Report

Client Sales Summary Report



P&CO
CHARTERED ACCOUNTANTS

Sales Summary Report

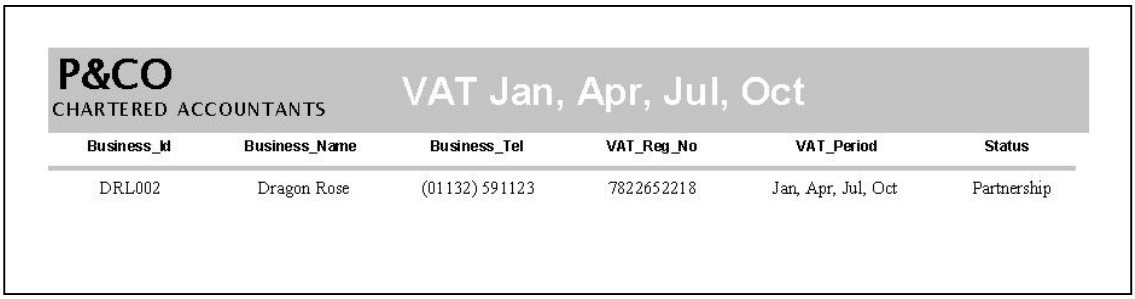
First_Name	Susan	Date	15/06/2003
Surname	Wong	Invoice_No	1
Business_Id	BCL001		

Subtotal	£120.00
VAT	£21.00
Total	£141.00

First_Name	Susan	Date	12/04/2003
Surname	Wong	Invoice_No	2
Business_Id	DRL002		

Subtotal	£110.00
VAT	£19.25
Total	£129.25

VAT Report Jan, Apr, Jul, Oct

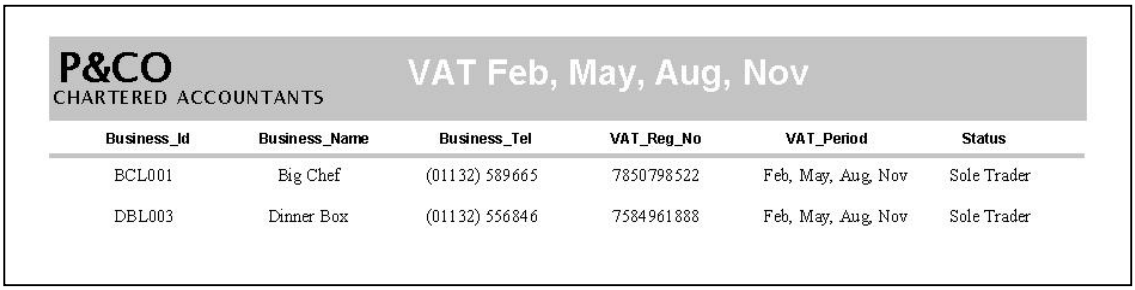


P&CO
CHARTERED ACCOUNTANTS

VAT Jan, Apr, Jul, Oct

Business_Id	Business_Name	Business_Tel	VAT_Reg_No	VAT_Period	Status
DRL002	Dragon Rose	(01132) 591123	7822652218	Jan, Apr, Jul, Oct	Partnership

VAT Report Feb, May, Aug, Nov

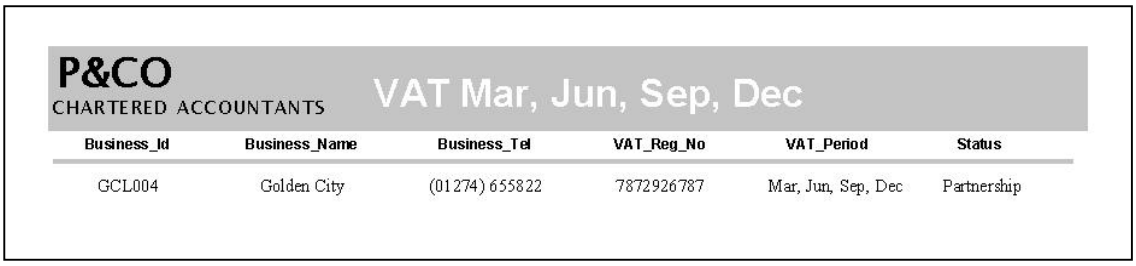


P&CO
CHARTERED ACCOUNTANTS

VAT Feb, May, Aug, Nov

Business_Id	Business_Name	Business_Tel	VAT_Reg_No	VAT_Period	Status
BCL001	Big Chef	(01132) 589665	7850798522	Feb, May, Aug, Nov	Sole Trader
DBL003	Dinner Box	(01132) 556846	7584961888	Feb, May, Aug, Nov	Sole Trader

VAT Report Mar, Jun, Sep, Dec



P&CO
CHARTERED ACCOUNTANTS

VAT Mar, Jun, Sep, Dec

Business_Id	Business_Name	Business_Tel	VAT_Reg_No	VAT_Period	Status
GCL004	Golden City	(01274) 655822	7872926787	Mar, Jun, Sep, Dec	Partnership

Appendix K – Testing of the Database System (Filled in by users)

Form Tested	Function Tested	Expected Result	Actual Result
Client Form	Add Client	Form permits a Client Details to be added to the 'Client' and 'Client_Business' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Client_Id' field.	As expected
	Find/Edit Client	Find a Client record correctly and permits amendments on that record.	As expected
	Delete Client	Warning message turns up to avoid unintended deletion of records and the record should be deleted from the related table(s) while the client has not required any P & Co service.	As expected
	Find Sales Ledger	This button is implemented in order to convenience the staff in case the client wants to check their account details. Once the button 'Find Sales Ledger' is pressed, it will bring the staff to the Form Client Sales Ledger and the staff will be able to look for the client accounts' details.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Business Form	Add Business	Form permits a Business Details to be added to the 'Business' and 'Client_Business' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Business_Id' field.	As expected
	Find/Edit Business	Find a Business record correctly and permits amendments on that record.	As expected
	Delete Business	Warning message turns up to avoid unintended deletion of records and the record should be deleted from the related table(s) while the client has not required any P & Co service.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Bookkeeping Form	Add Bookkeeping	Form permits a Bookkeeping Details to be added to the 'Bookkeeping' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Bkkeeping_Ledger_Id' field.	As expected
	Find/Edit Bookkeeping	Find a Bookkeeping record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
VAT Form	Add VAT	Form permits a VAT Details to be added to the 'VAT' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'VAT_Ledger_Id' field.	As expected
	Find/Edit VAT	Find a VAT record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
PAYE Form	Add PAYE	Form permits a PAYE Details to be added to the 'PAYE' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'PAYE_Sheet_Id' field.	As expected
	Find/Edit PAYE	Find a PAYE record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected

Form Tested	Function Tested	Expected Result	Actual Result
Tax Form	Add Tax	Form permits a Tax Details to be added to the 'Tax' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Tax_Sheet_Id' field.	As expected
	Find/Edit Tax	Find a Tax record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Sales Form	Add Sales	Form permits a Sales Details to be added to the 'Sales' and 'Sales_Service' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Invoice_No' field.	As expected
	Find/Edit Sales	Find a Sales record correctly and permits amendments on that record.	As expected
	Save Sales Record	Once 'Save Record' button is pressed, the figures for the fields 'Subtotal', 'VAT' and 'Total' are automatically calculated correctly.	As expected
	Preview and Print a Sales Invoice	The user interface of the form 'Sales' is also the standard sales invoice that can be printed out to clients. Preview of the sales invoice in print layout is possible and allow the invoice to be printed.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Payment Form	Add Payment	Form permits a Payment Details to be added to the 'Payment' and 'Sales_Payment' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Payment_Id' field.	As expected
	Find/Edit Payment	Find a Payment record correctly and permits amendments on that record.	As expected
	Automatic calculation for fields 'Amount_ Owed' and 'Balance'	Once the fields 'Amount_Owed' and 'Balance' are clicked, the VBA calculation functions are performed automatically.	As expected
	Preview and Print a Payment Invoice	The user interface of the form 'Payment' is also the standard payment invoice that can be printed out to clients. Preview of the payment invoice in print layout is possible and allow the invoice to be printed.	As expected
	Find Balance	Once the 'Find Balance' button is pressed, it is linked to the form Client Sales Ledger to allow the user to look for the clients' outstanding balance.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Client Sales Ledger Form	Find Record	This form is made up by the combination of 'Sales' and 'Payment' tables to show the P & Co clients' account details, so 'update', 'edit' and 'delete' of the clients' record should be either done in forms 'Sales' or 'Payment'. Once the 'Find Record' is pressed, a Client accounts' record is find correctly.	As expected
Client Sales Ledger Form	Preview and Print a Clients' Sales Ledger	The user interface of the form 'Client Sales Ledger' can be printed out to clients. Preview of the client sales ledger in print layout is possible and allow the client sales ledger to be printed.	As expected

Form Tested	Function Tested	Expected Result	Actual Result
Client Sales Summary Report	Preview and Print a Report	Preview all reports in print layout and allow the report to be printed	As expected
VAT Report	Preview and Print a Report	Preview all reports in print layout and allow the report to be printed	As expected
Main Menu Form	Open each form	When the 'View' button is clicked the relevant form should open.	As expected
	Close Main Menu	While the 'Quit' button is clicked the main menu should close and bring the user back to the Windows option.	As expected
	View VAT Report Sub-Menu	Once the 'View VAT Report Menu' button is pressed in Section 'VAT', the relevant VAT sub-menu should open.	As expected
	View Client Sales Summary Report Sub-Menu	Once the 'View Client Sales Summary Report Menu' button is pressed in Section 'Sales', the relevant Client Sales Summary sub-menu should open.	As expected

Comments:

- The navigation is easy to follow with the Main Menu to guide me through the database.
- Less time is needed to find a client record than before which means we can concentrate on our work, such as finishing the clients' bookkeeping and VAT earlier to provide a better customer service.
- We no longer need to worry about clients' missing files since their details are now stored in the server with backup every night.
- The Sales invoice and Payment receipt could have improved but this is still acceptable.
- The interface design looks professional and I really like the design of the P & Co logo on each forms and reports.

From:

Jonathan Pitayanukul

Owner of P & Co

Appendix K – Testing of the Database System (Filled in by users)

Form Tested	Function Tested	Expected Result	Actual Result
Client Form	Add Client	Form permits a Client Details to be added to the 'Client' and 'Client_Business' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Client_Id' field.	As expected
	Find/Edit Client	Find a Client record correctly and permits amendments on that record.	As expected
	Delete Client	Warning message turns up to avoid unintended deletion of records and the record should be deleted from the related table(s) while the client has not required any P & Co service.	As expected
	Find Sales Ledger	This button is implemented in order to convenience the staff in case the client wants to check their account details. Once the button 'Find Sales Ledger' is pressed, it will bring the staff to the Form Client Sales Ledger and the staff will be able to look for the client accounts' details.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Business Form	Add Business	Form permits a Business Details to be added to the 'Business' and 'Client_Business' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Business_Id' field.	As expected
	Find/Edit Business	Find a Business record correctly and permits amendments on that record.	As expected
	Delete Business	Warning message turns up to avoid unintended deletion of records and the record should be deleted from the related table(s) while the client has not required any P & Co service.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Bookkeeping Form	Add Bookkeeping	Form permits a Bookkeeping Details to be added to the 'Bookkeeping' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Bkkeeping_Ledger_Id' field.	As expected
	Find/Edit Bookkeeping	Find a Bookkeeping record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
VAT Form	Add VAT	Form permits a VAT Details to be added to the 'VAT' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'VAT_Ledger_Id' field.	As expected
	Find/Edit VAT	Find a VAT record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
PAYE Form	Add PAYE	Form permits a PAYE Details to be added to the 'PAYE' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'PAYE_Sheet_Id' field.	As expected
	Find/Edit PAYE	Find a PAYE record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected

Form Tested	Function Tested	Expected Result	Actual Result
Tax Form	Add Tax	Form permits a Tax Details to be added to the 'Tax' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Tax_Sheet_Id' field.	As expected
	Find/Edit Tax	Find a Tax record correctly and permits amendments on that record.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Sales Form	Add Sales	Form permits a Sales Details to be added to the 'Sales' and 'Sales_Service' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Invoice_No' field.	As expected
	Find/Edit Sales	Find a Sales record correctly and permits amendments on that record.	As expected
	Save Sales Record	Once 'Save Record' button is pressed, the figures for the fields 'Subtotal', 'VAT' and 'Total' are automatically calculated correctly.	As expected
	Preview and Print a Sales Invoice	The user interface of the form 'Sales' is also the standard sales invoice that can be printed out to clients. Preview of the sales invoice in print layout is possible and allow the invoice to be printed.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Payment Form	Add Payment	Form permits a Payment Details to be added to the 'Payment' and 'Sales_Payment' table. Input into every field in the form is bound properly to the table. Ensures that a number is entered into the unique 'Payment_Id' field.	As expected
	Find/Edit Payment	Find a Payment record correctly and permits amendments on that record.	As expected
	Automatic calculation for fields 'Amount_ Owed' and 'Balance'	Once the fields 'Amount_Owed' and 'Balance' are clicked, the VBA calculation functions are performed automatically.	As expected
	Preview and Print a Payment Invoice	The user interface of the form 'Payment' is also the standard payment invoice that can be printed out to clients. Preview of the payment invoice in print layout is possible and allow the invoice to be printed.	As expected
	Find Balance	Once the 'Find Balance' button is pressed, it is linked to the form Client Sales Ledger to allow the user to look for the clients' outstanding balance.	As expected
	Quit	Once the quit button is pressed, it should bring the staff back to the main menu.	As expected
Client Sales Ledger Form	Find Record	This form is made up by the combination of 'Sales' and 'Payment' tables to show the P & Co clients' account details, so 'update', 'edit' and 'delete' of the clients' record should be either done in forms 'Sales' or 'Payment'. Once the 'Find Record' is pressed, a Client accounts' record is find correctly.	As expected
Client Sales Ledger Form	Preview and Print a Clients' Sales Ledger	The user interface of the form 'Client Sales Ledger' can be printed out to clients. Preview of the client sales ledger in print layout is possible and allow the client sales ledger to be printed.	As expected

Form Tested	Function Tested	Expected Result	Actual Result
Client Sales Summary Report	Preview and Print a Report	Preview all reports in print layout and allow the report to be printed	As expected
VAT Report	Preview and Print a Report	Preview all reports in print layout and allow the report to be printed	As expected
Main Menu Form	Open each form	When the 'View' button is clicked the relevant form should open.	As expected
	Close Main Menu	While the 'Quit' button is clicked the main menu should close and bring the user back to the Window option.	As expected
	View VAT Report Sub-Menu	Once the 'View VAT Report Menu' button is pressed in Section 'VAT', the relevant VAT sub-menu should open.	As expected
	View Client Sales Summary Report Sub-Menu	Once the 'View Client Sales Summary Report Menu' button is pressed in Section 'Sales', the relevant Client Sales Summary sub-menu should open.	As expected

Comments:

- It is much quicker and easier to find a record using the database.
- We can now update the client record in real time whilst talking to the clients. This avoid the lose of clients' details being written on paper.
- There are some forms with subform where we need to scroll through to enter the details which is a bit inconvenient.
- The database provides a comprehensive functions that allow us to do the job quickly.

From:

Kitty Wan

Member of P & Co Staff

P & Co Chartered Accountant

Company Database

User Manual

Written and Produced by:
Ka Man Chan
Version 1.0
29th April 2003

1. INTRODUCTION

This menu was designed and written by the system developer, Ka Man Chan. It is written and presented in an easy to understand and brief style as demand by the client. The menu will provide a brief overview of the Database for P & Co Chartered Accountant and should be used in combination with the user training and thereafter as a reference manual.

The Database covers nine main sections: **Client, Business, Bookkeeping, VAT, PAYE, Tax, Sales, Payment and Client Sales Ledger**. A brief explanation of the processes involved in each section will be included in this section, using screen shot of the system to assist in understanding.

Due to the time constraints and similarity of the form such as *'Client and Business'*, *'Bookkeeping, VAT, PAYE and Tax'*, this version will only contain detailed information on the 'Client', 'VAT', 'Sales' and 'Payment' processes. A more detailed version will be completed and given to the user.

2. OPENING THE DATABASE

This system is a Microsoft Access database package. To open the database:

- Click on the **Start Menu**
- Choose **Programs**, then Choose **Microsoft Access**

Figure 2.1. Starting Microsoft Access

Microsoft Access is then loaded and the file 'P & Co Database.mdb' needs to be opened. The display screen will turn up when Access commences and the database can then be chosen. See Figure 2.2 below.

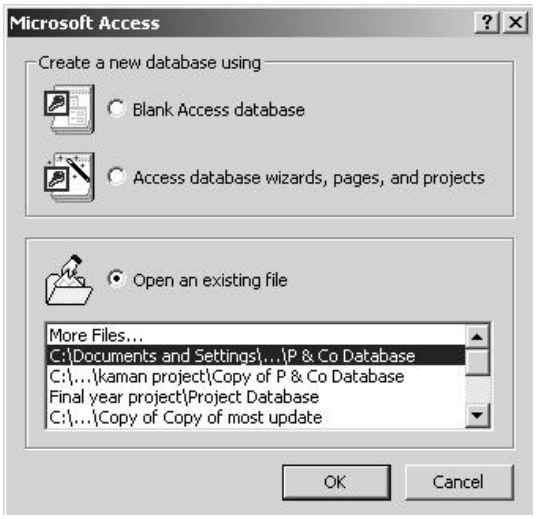


Figure 2.2. Selecting the Database

By **clicking** on the name **'P & Co Database.mdb'** and **select OK,** the database will automatically load up and the user will be prompted to enter a login password, as shown in Figure 2.3. This is implemented in order to restrict the database access to authorised users only. The password will be supplied at the appropriate time.

Figure 2.3. Enter a Password

After the **password is entered** and **OK is clicked**, the database is now running, the user will be able to see the Main Menu Screen, as shown in Figure 2.4. The user can then choose any of the nine options by clicking on the tab button with the relevant text on it.

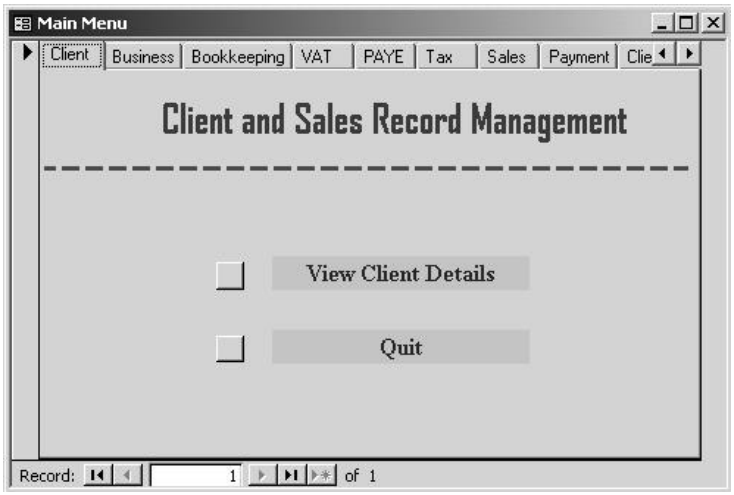


Figure 2.4. Main Menu

By selecting the relevant tab button on the main menu, the database will navigate the user to the relevant screens. The Quit option will close the Database and return the user to the Windows menu.

3. Client

When the Client option is chosen from the Main Menu, the user is directed to the screen shown in Figure 3.1 below.

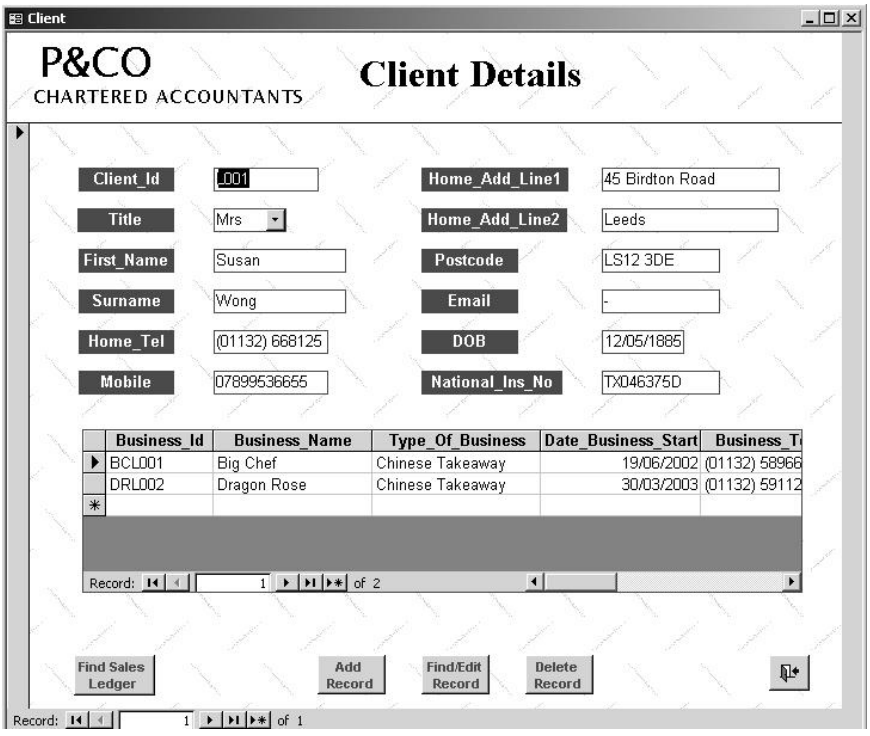


Figure 3.1. Client Details Screen

3.1 – Add Client Record

New Client Record can be added by clicking this button 

in the Client Details form, a new screen as indicated in Figure 3.2 will turn up. The user can then input all the client details in the form.

Figure 3.2. New Client Details Screen

3.2 – Find/Edit Client

Client record can be found and edited once the details are entered in the database. Clients' details can be found and edited by the 

button. When this button is clicked, the Find/Replace box will turn up, see Figure 3.3 and the user can enter any key fields in the 'Find What' box and then select one of the field 'Look in'. If 'Client' is selected in the 'Look in' box, it means that the database will be able to search through the database to retrieve the right Client record that matches the 'Find What' field. Afterthat, 'Find Next' button need to be clicked to operate the search function. Once the client record is found, the user can view client details and make necessary changes if necessary.

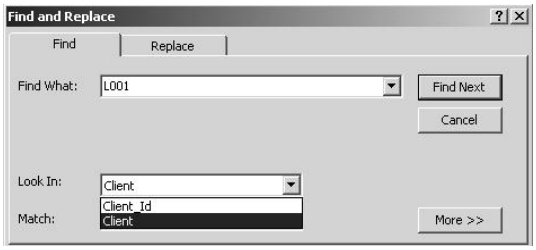


Figure 3.3 Find Box

3.3 – Delete Client

Client record may only be deleted when there is no transactions records exist. This means that the client has not required any service from P & Co Chartered Account and thus, the record do not need to be kept. When a member of staff wishes to delete the client record, the Find/Edit Record button will first need to be pressed to go to the corresponding client record. Once the client record is found, the Delete Record button can be clicked to delete the client record.

3.4 – Find Sales Ledger

Client Sales Ledger record can be found by clicking this Find Sales Ledger button to bring the user to the 'Client Sales Ledger' form, see Figure 3.4. The user can then find the corresponding client sales ledger record to get the account details. This button is implemented as an enhancement to convenient the user.

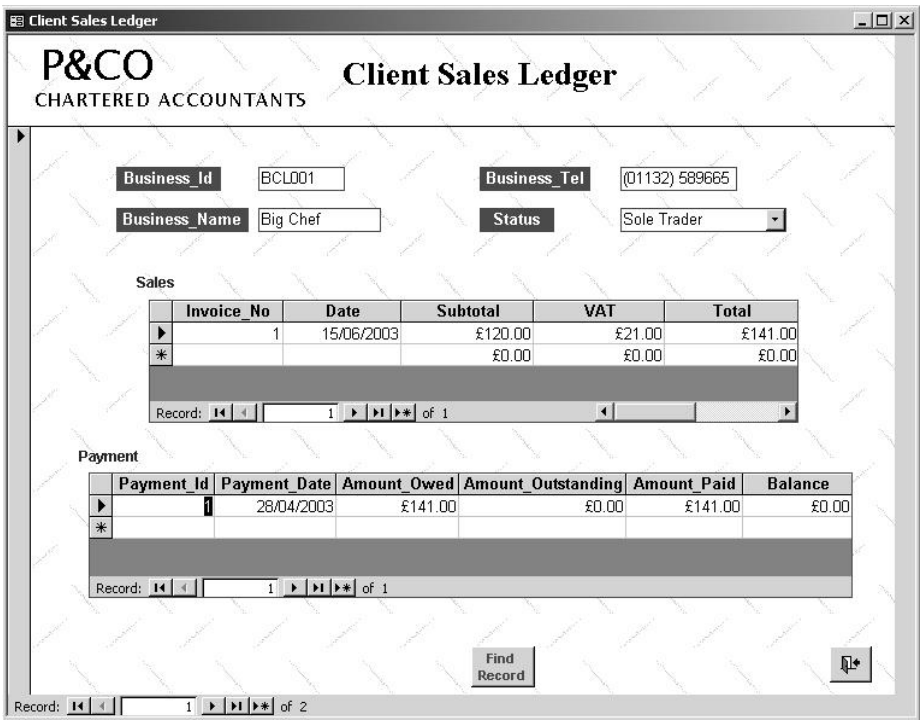


Figure 3.4. Client Sales Ledger Form

3.5 – Quit and Back to Main Menu

The user can go back to the main menu by clicking this button .

4. VAT

When the VAT option is chosen from the Main Menu, the user is directed to the screen shown in Figure 4.1 below.

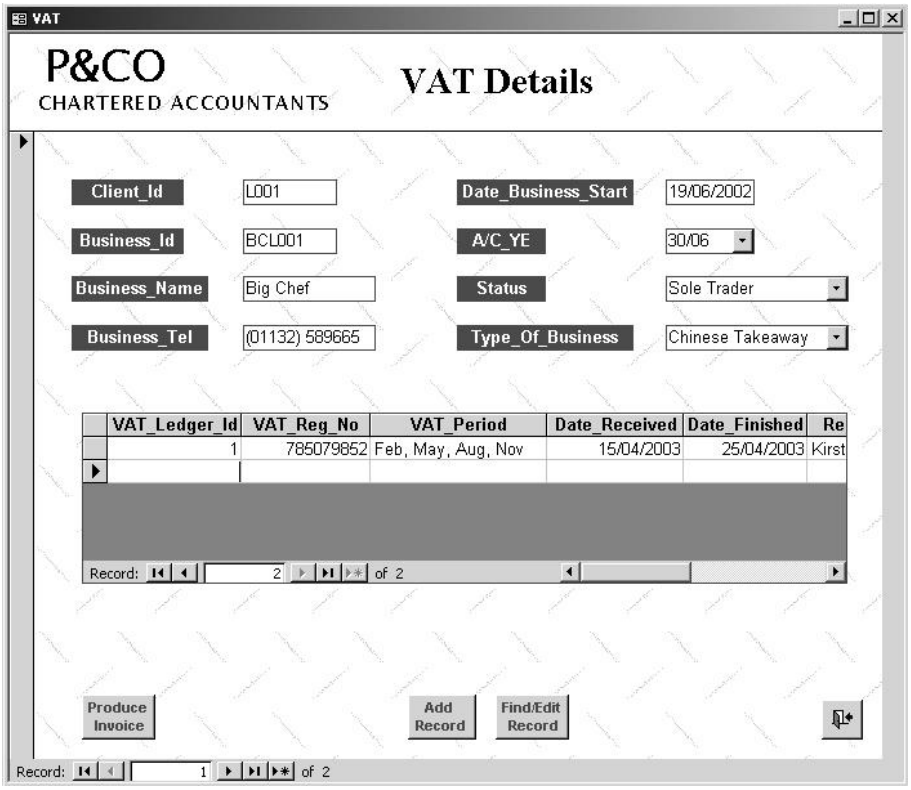


Figure 4.1. VAT Details Screen

4.1 – Add VAT Record

New VAT Record can be added by clicking this button 

in the VAT Details form, the function button operates as the same as the Client Details Form. However, when details are input, it is only the field 'Client_Id', 'Business_Id' and the VAT subform, as shown in Figure 4.2 that needs to be entered, the rest of the business details will come up automatically since they are linked to the business table.

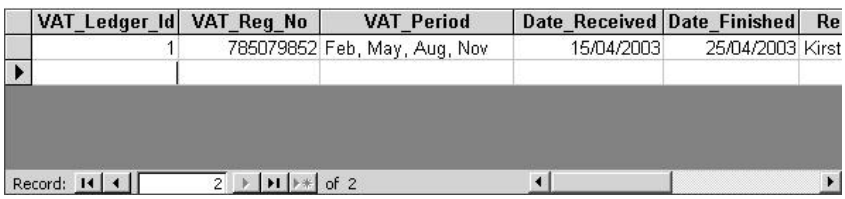


Figure 4.2. VAT subform

4.2 – Find/Edit VAT Record

Find/Edit for VAT record of the client business operates the same as the Client Form as mentioned in Section 3.2. VAT details can be found and edited by the Find/Edit Record button. When this button is clicked, the Find/Replace box will turn up, see Figure 3.3 and the user can enter any key fields in the 'Find What' box and then select one of the field 'Look in'. If 'VAT' is selected in the 'Look in' box, it means that the database will be able to search through the database to retrieve the right VAT record that matches the 'Find What' field. Afterthat, 'Find Next' button need to be clicked to operate the search function. Once the VAT record is found, the user can view VAT details and make necessary changes if necessary.

4.3 – Produce Invoice

Sales Invoice can be produced by clicking this Produce Invoice button to bring the user to the 'Sales' form which is the sales invoice to the client, see Figure 4.3. The user can then create new invoice by clicking the 'Add Invoice' button. This button is implemented as an enhancement to convenient the user.

Figure 4.3. Sales Invoice Form

4.4 – Quit and Back to Main Menu

The user can go back to the main menu by clicking this button .

5. Sales

When the Sales option is chosen from the Main Menu, the user is directed to the screen shown in Figure 4.3 above.

5.1 – Add Sales Record

New Sales Record can be added by clicking this button 

in the Sales Details form, a new screen will turn up. The user can then input all the client sales details in the form.

5.2 – Find/Edit Sales Record

Find/Edit for Sales record of the client business operates the same as the Client Form and VAT Form as mentioned in Section 3.2 and Section 4.2. Sales details can be found and edited by the 

button.

5.3 – Save Record

Once this button 

is clicked, the fields 'Subtotal', 'VAT' and 'Total' will be calculated automatically provided that the user has entered the top half of the form, the sample can be found in Figure 5.1.

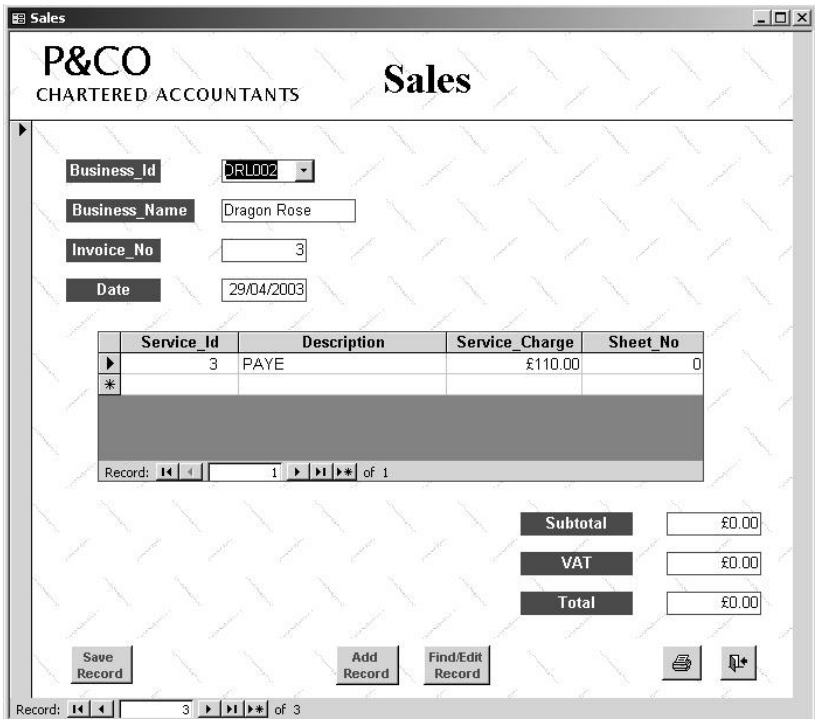


Figure 5.1. Sales Invoice Form

5.4 – Print Sales Invoice

Once the  button is clicked, the sales invoice form can be printed, which is the same as the sales interface form.

5.5 – Quit and Back to Main Menu

The user can go back to the main menu by clicking this button  .

6. Payment

When the Payment option is chosen from the Main Menu, the user is directed to the screen shown in Figure 6.1 below.

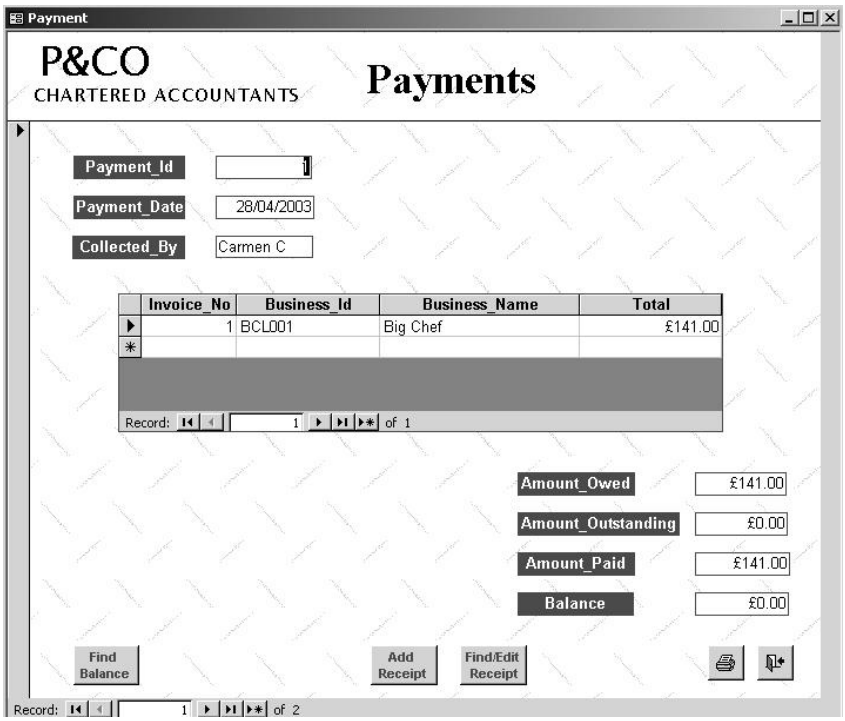


Figure 6.1. Payment Receipt Form

6.1 – Add Payment Record

New Payment Record can be added by clicking this button 

in the Payment Details form; a new screen will turn up. The user can then input all the client payment details in the form. When the user has filled in the top half of the form (after the payment subform), the user can just click on the field 'Amount_Owed', the total amount the client owed will be calculated automatically. For the field

'Amount_Outstanding', the user will need to make use of the Find Balance button to find the client outstanding balance. This Find Balance button will bring the user to the Client Sales Ledger form which is shown in Figure 3.4, the user will need to look up the field 'Balance' and then enter that in the 'Amount_Outstanding' field. Also, the field 'Amount_Paid' will also need to be entered by the user since the client may not pay the entire bill at once. However, the field 'Balance' will then be automatically calculated when the above fields are entered.

6.2 – Find/Edit Payment Record

Find/Edit for Payment record of the client operates the same as the Client Form and VAT Form as mentioned in Section 3.2 and Section 4.2. Payment details can be found and edited by the Find/Edit Receipt button.

6.3 – Print Payment Invoice

Once the button is clicked, the payment invoice form can be printed, which is the same as the payment interface form.

6.4 – Quit and Back to Main Menu

The user can go back to the main menu by clicking this button .

7. Business

This form is operate very similar to the Client Form apart from the 'Find Balance' do not appear in this form, the user can add, find, edit and delete the business record which is the same as the Client Form. Also, the user can quit and go back to the main menu. Therefore, for details of how to operate this form can be found in **Section 3**.

8. Bookkeeping, PAYE, Tax

These form are operated the same as the VAT Form, the user just need to input the 'Client_Id', 'Business_Id' and the subforms, then the rest of the client or business details will turn up automatically because they are either linked to the client table or business table. For the functionality such as add, find, edit and produce invoice are operated the same as VAT. Thus, for details of how to operate these forms can be founded in Section 4.

9. Client Sales Ledger

The sample interface form of this can be found in Figure 3.4, the user can only find client sales ledger by clicking the ‘Find Record’ button since amendments will need to be made either from ‘Sales’ or ‘Payment’ form. When the changes are made either to Sales or Payment, the Client Sales Ledger form will be updated automatically. Also, there is a print button for the user to print out the Client Sales Ledger.